SKYCORE FRAMEWORK (DRAFT)

SKYLINK PLATFORM

OCTOBER 1, 2023

SEASON7
sznsupport@szn7.com

What's the SKYLINK ?

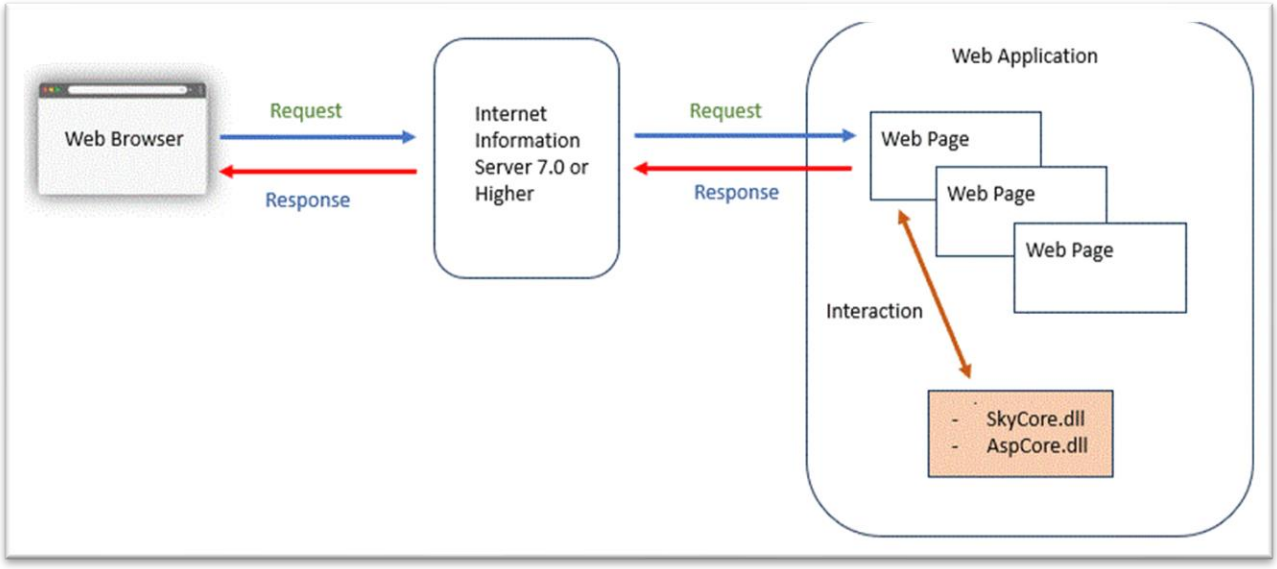
SKYLINK is web application development platform which enable programmers to build web sites easily. This platform is built on Asp.Net Tech knowledge and allows developers to use useful built-in methods which can be easily implemented.

SKYLINK is comprised of SkyCore and AspCore development frameworks. A developer uses SkyCore to develop modern web application in multi-area and uses AspCore for traditional Asp.Net programming.

Basic Concepts

Developing web applications, we spend countless number of hours over HTML coding and each elements' styles.
For that reason, (in order to increase development efficiency and reduce development time) it would be good practice to use dynamic http handler and premade web components. In that sense, SKYLINK products claim to be a good way of web development.

Fig1. Conceptual Scheme



Contact

If you have any question, please feel free to reach out to

sznsupport@szn7.com

Development Platform

SKYLINK offers dynamical web programming and Asp.Net programming environment in Visual Studio 2013 or higher.

- The SKYLINK products support C# or VB.net languages.
- The SKYLINK works on Windows Server/PC and Internet Information Service(ver 7.0 or higher).

1. Dynamic Development Environment (SkyCore Framework)

In order to develop web applications dynamically, it must include **"skycore.dll"** file in the application project(bin folder).
- Developers can enhance their programming experience through this dynamic development environment.
- The **"skycore.dll"** is one of the SKYLINK products and the file should be located in "bin" folder in the web project.
- Also it should define a handler at **"system.webServer"** section in the "web.config file".

Fig1. web.config settings

```
<system.webserver>
        <handlers>
                <add name="skycore" verb="*" path="*" type="skycore.IHandler" />
        </handlers>
        ..............
```

2. Asp.Net Development Environment (AspCore Framework)

In order to develop Asp.net applications, it must include **"aspcore.dll"** file in the application project(bin folder).
- This environment is for those who want develop web applications in the traditional Asp.Net project.
- The **"aspcore.dll"** is one of the SKYLINK products and the file should be located in "bin" folder in the web project.
- Also it should define a handler at **"system.webServer"** section in the "web.config file".

Fig2. web.config settings

```
<system.webserver>
        <handlers>
                <add name="aspcore" verb="POST" path=".api" type="aspcore.IHandler" />
        </handlers>
        ..............
```

Download Templates

The templates of "SkyCore Framework" and "Asp.Net Framework" respectively include **"SkyCore.dll" and "AspCore.dll"** in the "bin" folder.

1. SkyCore templates (vb & c#)
2. AspCore templates (vb & c#)

These project templates are ready to develop any web application immediately.(see. "Prerequisite & example projects" chapter)

(1) Download template project and extract files from zip file.
(2) Copy & paste to the project folder.
(3) Change the forder name that you want to name it.
(4) Mount(publish) on Internet Information Service.
(5) Open a web project in Visual Studio.

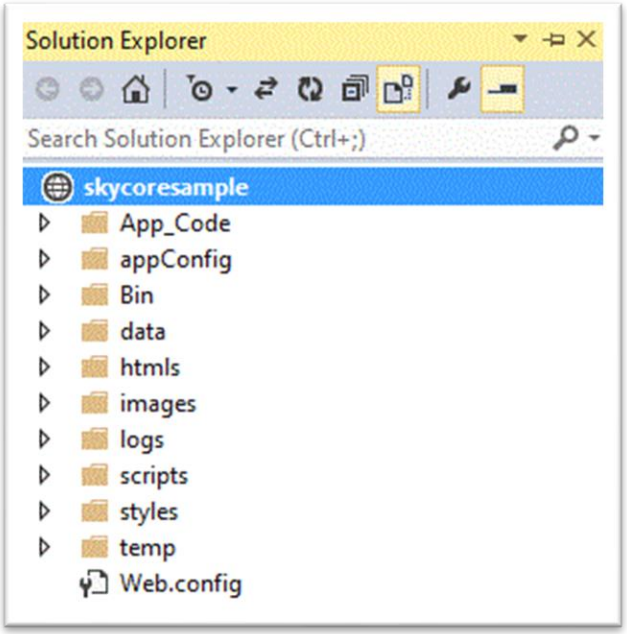
1. SkyCore Framework (Dynamic)

Explanation for system folders in this project template

- **App_Code**: Program Classes, initially empty
- **appConfig**: Web Project Environmental Information (application.cfg - text file)
- **bin**: Project reference files like **skycore.dll** or others
- **data**: reserved for developer, initially empty
- **htmls**: html files, initially empty
- **images**: system image files, includes some image files
- **logs**: system logs and reserved for developer, initially empty
- **scripts**: java script files, initially empty
- **styles**: css files, initially empty
- **temp**: reserved for developer, initially empty

Download SkyCore & example projects (vb & c#): skycoresample.zip

Fig1. Project Template



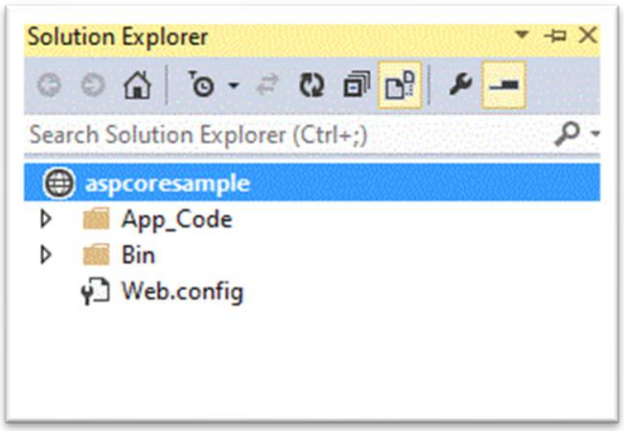
2. AspCore Framework (Traditional Asp.Net)

Explanation for system folders in a project

■ **App_Code**: Program Classes, initially empty
■ **bin**: Project reference files like **aspcore.dll** or others

Download AspCore & example projects (vb & c#): [aspcoresample.zip]

Fig2. Project Template



SkyCore Framework

SkyCore is one of the SKYLINK Products which enable programmers to build web sites easily & dynamically.
The "skycore.dll" is dynamic link library file which include built-in classes and javascript functions.
That file is the only vital element for each web project and should be located in "bin" folder before starting programming.

Main Built-in Classes

■ **WebPage Class**
In order to use SkyCore platform, webpages must inherit this class. There are all the methods what you need for building webpages and it enables making communication with client's web browser directly

■ **WebCore Class**
This class provides important httprequest information & many useful functions.

■ **Mail Class**: Calling this class, a developer can quickly implement mailing program.

■ **Encrypt Class**
Enabling a developer to encrypt/decrypt data.

■ **OleData Class**
Data input/output for excel, csv, and so on

■ **SQLData Class**
Data input/output for Microsoft SQL Database

■ **OraData Class**
Data input/output for Oracle Database. A developer need to register & put "OraOledb12.dll" file into "bin" folder.

■ **ToolKit Class**
Various Webpage UI(User Interface) Controls

Built-in Javascript Functions

$ApiRequest()
Web browser sends data through "$ApiRequest" function. Web Server replys messages responding to the request.

$WaitOn()
Display a built-in waiting icon

$WaitOff()
Disable a built-in waiting icon

$ScrollTop()
Move scroll to top

$fadeoutobj(o)
Fading out html element. o:element object

$fadeinobj(o)
Fading in html element. o:element object

$PopOn(htmltext)
Popup window to display html text

$PopOff()
Disable Popup window (fading out)

$ClsPop(event)
Disable Popup window

$CenterElement(elementId)
Centering fixed html element

$callFunctionByName(functionName, context, arguments)
Call a javascript function by name

$CoreAction(args)
Call a server-side function directly by name

$stripXmltags(xml, tagname)
Returns contents in XML by specific tagname

$tableAddRow(cellClickedInTable)
Add a row in table element after selected row

$tableDelRow(cellClickedInTable)

Delete a selected row in table element

$tableCopyRow(cellClickedInTable)
Copy a row in table element after selected row

$tableToggleRow(cellClickedInTable, cellId)
Add/Delete a row in table element

$ElmDisplayToggle(element)
Display/Hide a html element

$ClickElement(element)
Invoke click event on specific html element

$StartTimer(elementId)
Display timer in html element

Environmental Settings

"application.cfg" file in appConfig folder

1.System Configuration: These values can be replaced but Do Not Delete
2.Using Oracle Database, OraOledb must be registerd in system (regsvr32.exe OraOledb12.dll)
3. mail.securityprotocoltype : 0.default, 48.ssl3, 192.tls, 768.tls11, 3072.tls12, 12288.tls13
4. app.app.runningmode : 0.Demo, 1.Double Authentication, 2.Traditional Authentication

- **app.app.name**: Web Application Name
- **app.app.version**: Web Application Version
- **app.app.releasedate**: Web Application Release Date
- **app.app.runmode**: Web Application Running Mode
- **app.app.name**: Web Application Name
- **app.app.home**: Web Application Main Page
- **app.app.license**: Product License Key
- **app.Settings.encryptkey**: Data Encryption Key
- **app.Settings.isocode**: Web Application Default Language Code
- **app.Settings.font**: Web Application Default Font Name
- **app.Settings.pagetimeout**: Page Timeout
- **app.Settings.waitimage**: Wait-Icon url
- **app.folders.code**: Server-side program code file folder (Web Application Default)
- **app.folders.html**: Html file folder (Web Application Default)
- **app.folders.script**: Javascript file folder (Web Application Default)
- **app.folders.style**: Css file folder (Web Application Default)
- **app.folders.image**: Application image file folder (Web Application Default)

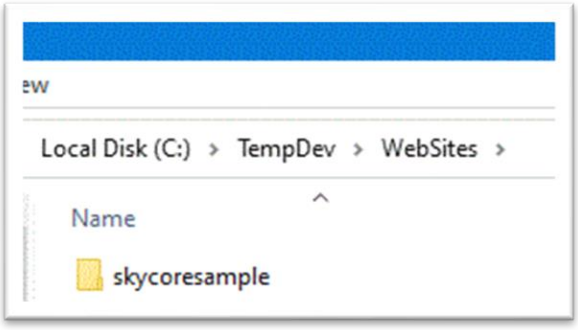
■ **app.folders.temp**: Temporay file folder (Web Application Default)
■ **app.folders.data**: Application data file folder (Web Application Default)
■ **app.folders.log**: Application log file folder (Web Application Default)
■ **app.folders.bin**: Reference file folder (Web Application Default)
■ **app.mail.server**: Mailserver Address
■ **app.mail.port**: Mailserver Port
■ **app.mail.addr**: Sender Email Address
■ **app.mail.id**: Mailserver Id
■ **app.mail.password**: Mailserver Password
■ **app.mail.title**: Notice Mail Title
■ **app.mail.credentials**: Use/Not Use Credentials
■ **app.mail.securityprotocoltype**: Mailserver Security Type
■ **app.sqldb.source**: MS-SQL Server data source
■ **app.sqldb.catalog**: MS-SQL Server database name
■ **app.sqldb.id**: MS-SQL Server login id
■ **app.sqldb.password**: MS-SQL Server login password
■ **app.sqldb.timeout**: MS-SQL Server connection timeout
■ **app.oradb.host**: Oracle database host
■ **app.oradb.service**: Oracle database service name
■ **app.oradb.port**: Oracle database connection port
■ **app.oradb.id**: Oracle database login id
■ **app.oradb.password**: Oracle database login password
■ **app.oradb.timeout**: Oracle database connection timeout

Prerequisite

Publish to IIS(Internet Information Service) & Open Project in Visual Studio

1. Copy & paste this project folder

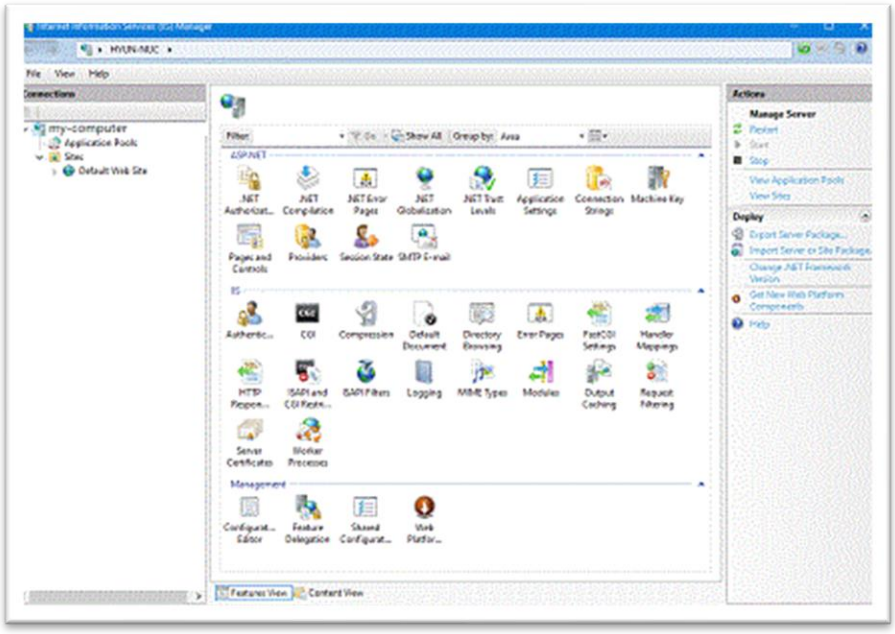
■ Copy the example project to any foler after extraction.



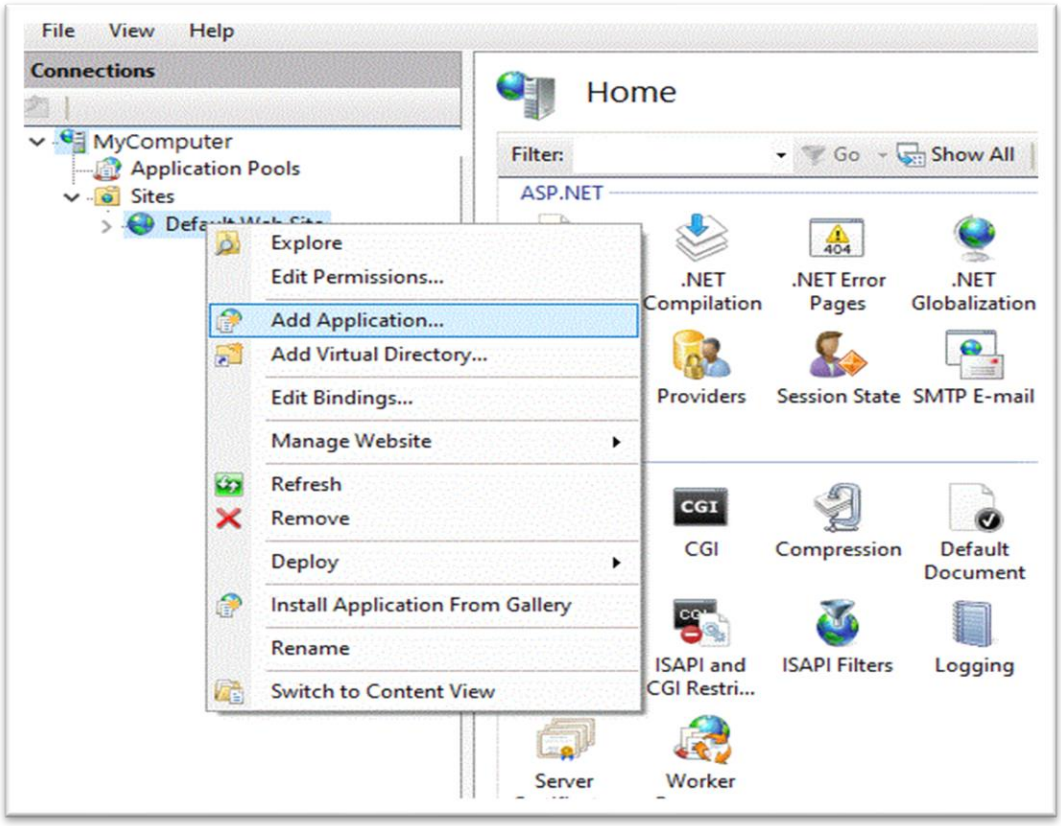
2. You can change the forder name as your own project name

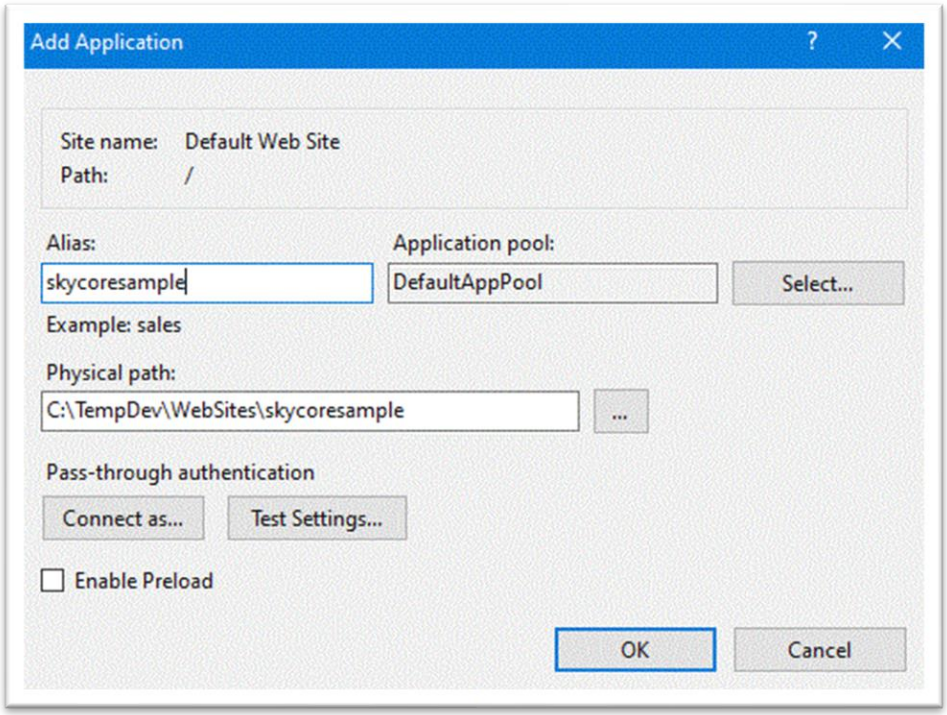
3. Mount(publish) the project on IIS(Internet Information Service) on local/server computer

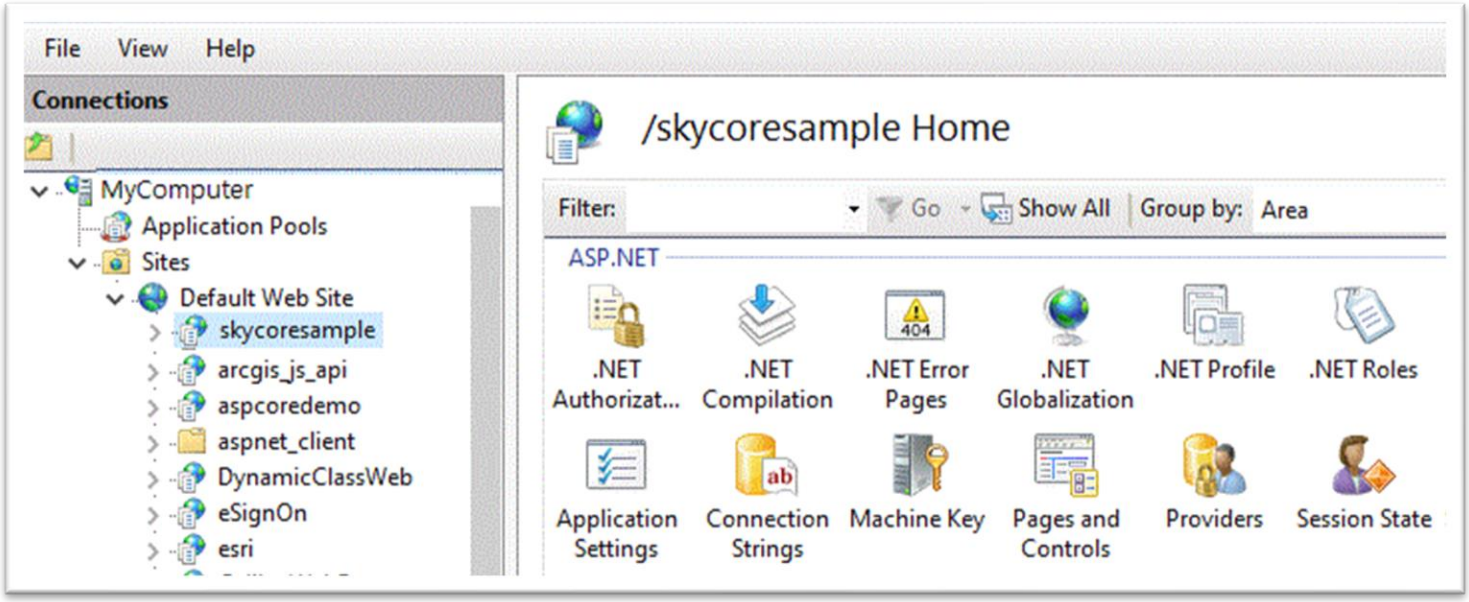
■ Open Internet Information Service



■ Add Application (right mouse click on a website to mount)

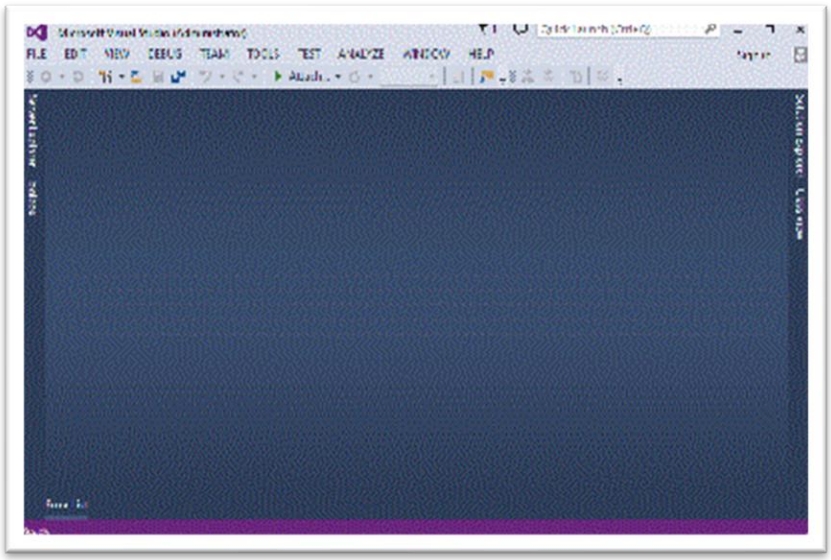




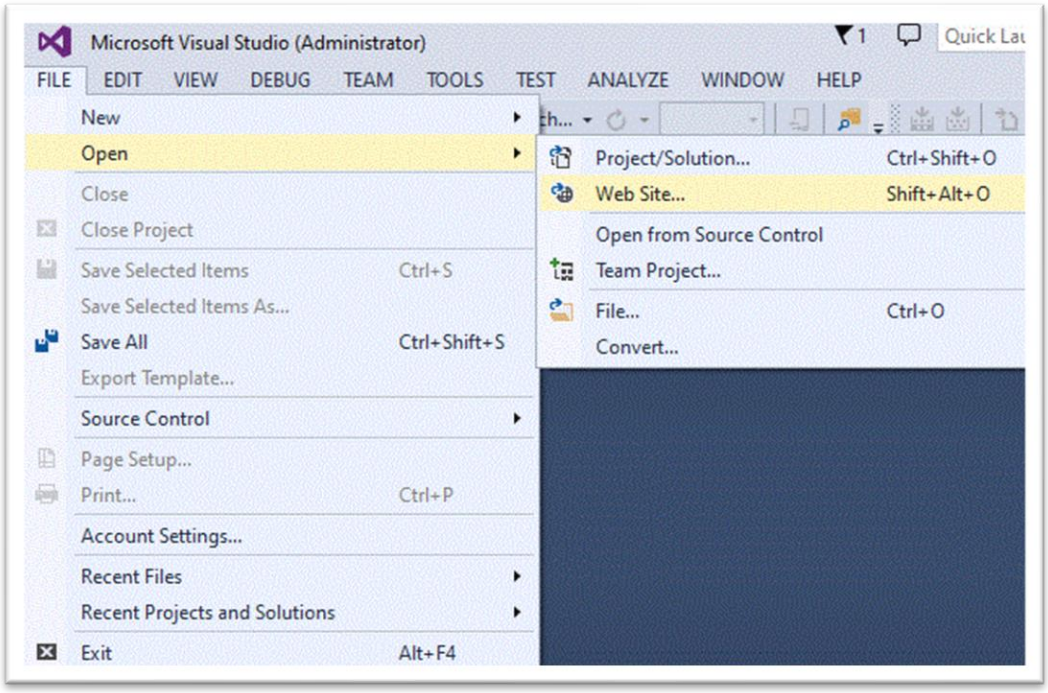


4. Open a web project in Visual Studio (2013 or higher)

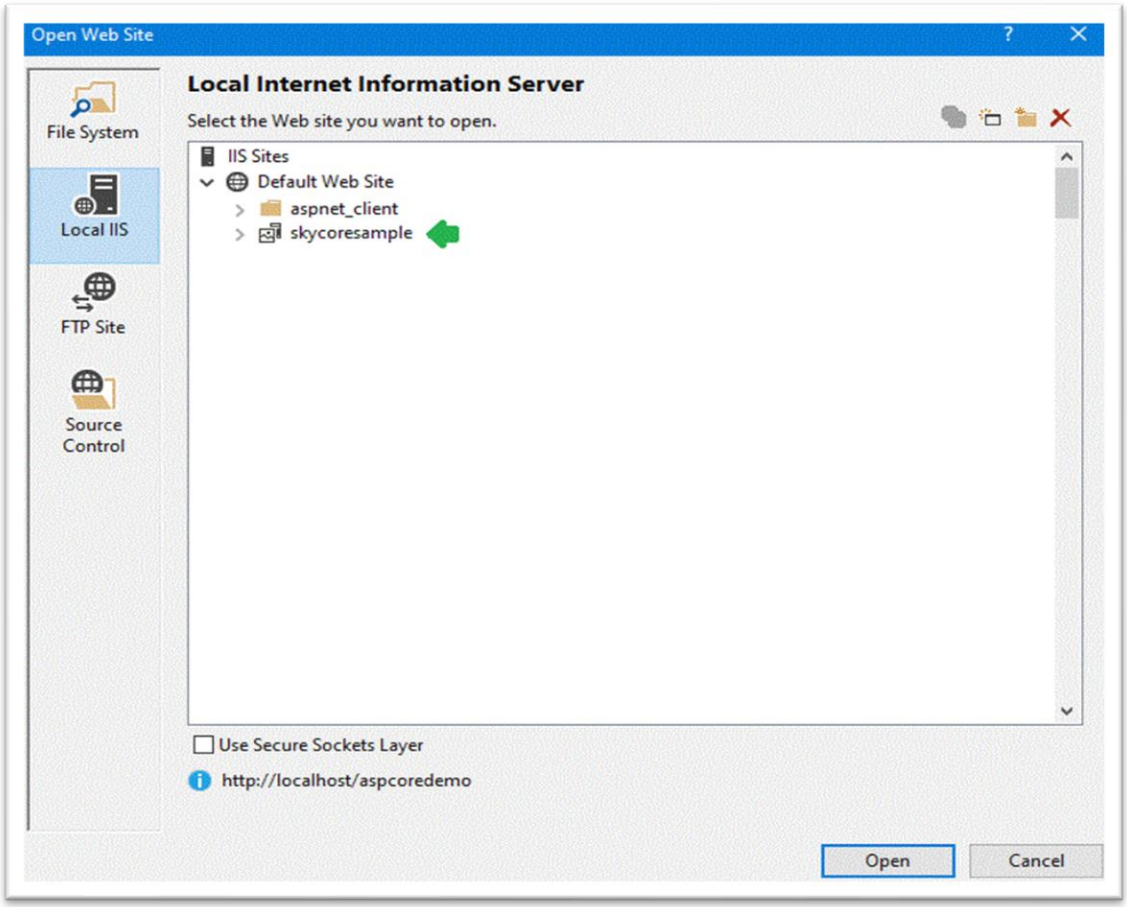
■ Open Visual Studio as administrator.



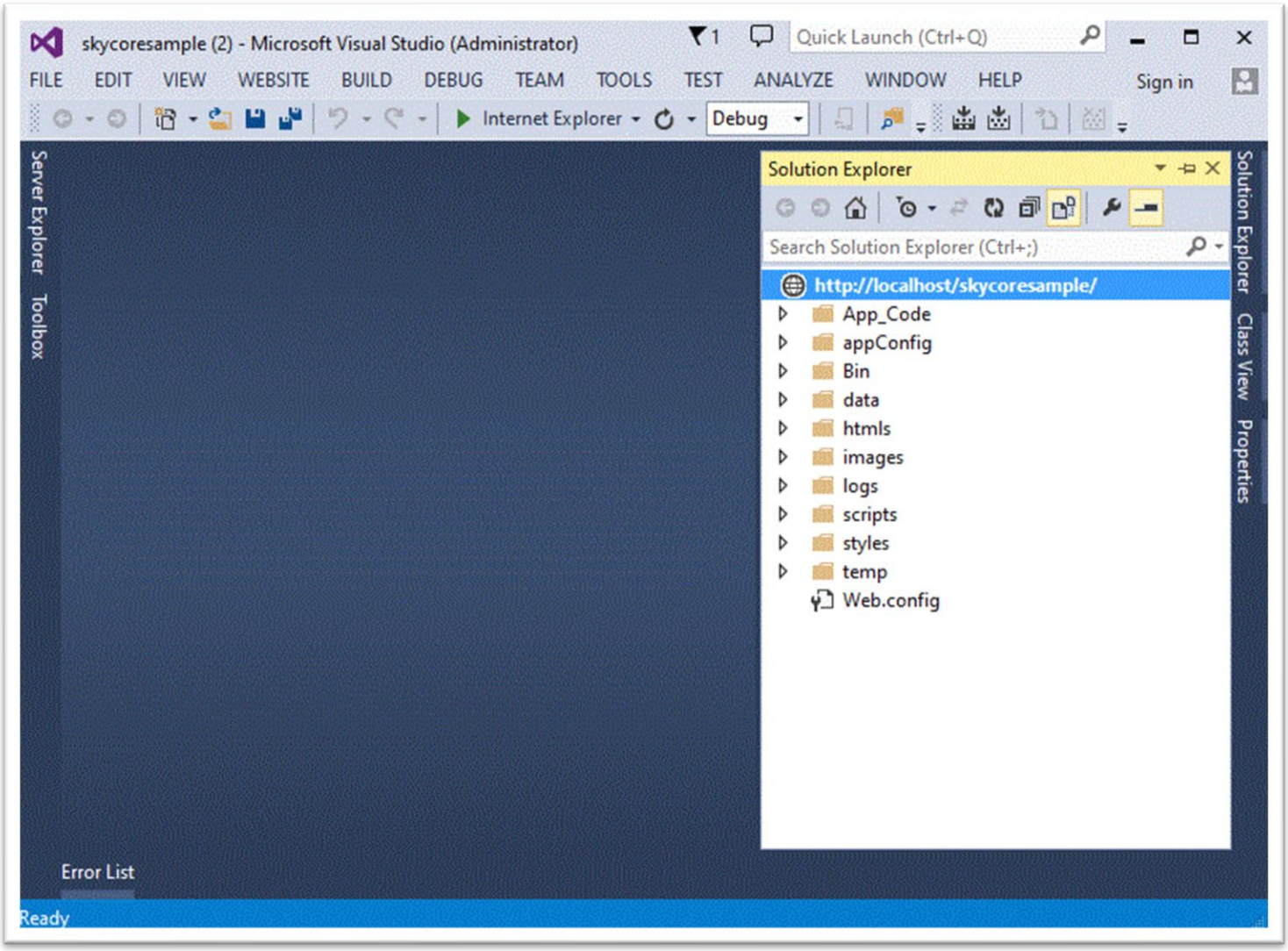
■ Click "File" menu and "Open - Web Site"



■ Select a web project in Local IIS



■ Let's start programming



Hello World - The First Web Application

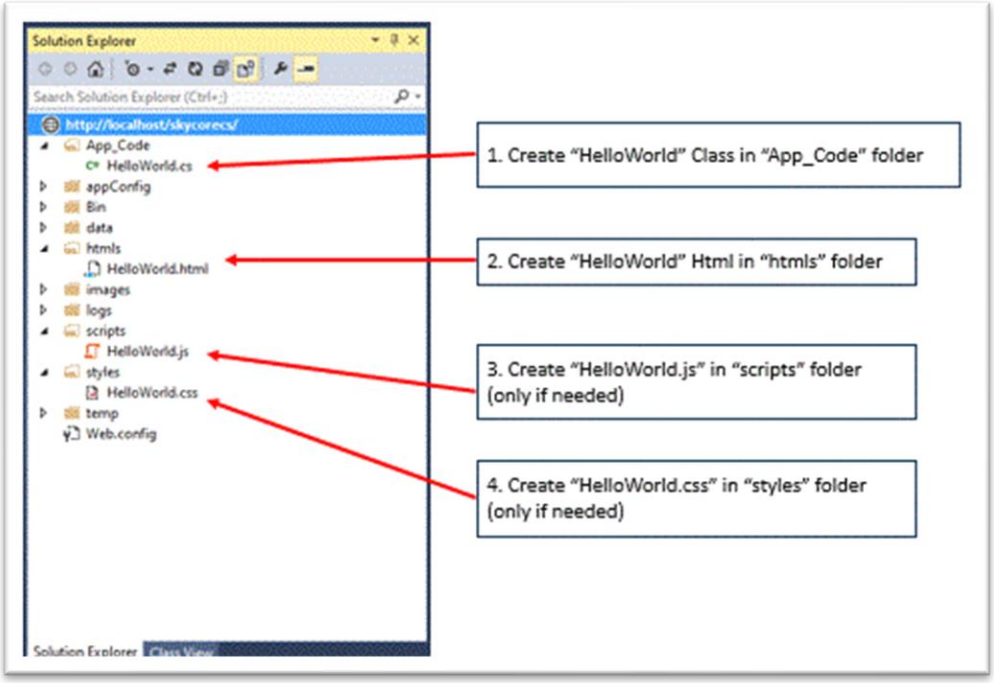
There are 3 ways to create a webpage like below

1. Using "Webpage" class with a html template
2. Using "Webpage" class without a html template
3. Traditional Html file

Now, Let's create the first webpage "HelloWorld".

1. Using "Webpage" class (Built-In) with a html template

If the webpage name is "Helloworld", it is supposed to have the same file names as
- "cs/vb" extention in "App_Code" folder
- "html" extention in "htmls" folder
- "js" extention in "script" folder (if it needs javascript code for "Helloworld" webpage)
- "css" extention in "styles" folder (if it needs styles for "Helloworld" webpage)



A. "cs/vb" extention in "App_Code" folder

Fig1. Helloworld.cs

```
using System;
using skycore;

public class HelloWorld:WebPage
{
   public HelloWorld()
            {
            }
}
```

Fig2. Helloworld.vb

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class HelloWorld
   Inherits WebPage

End Class
```

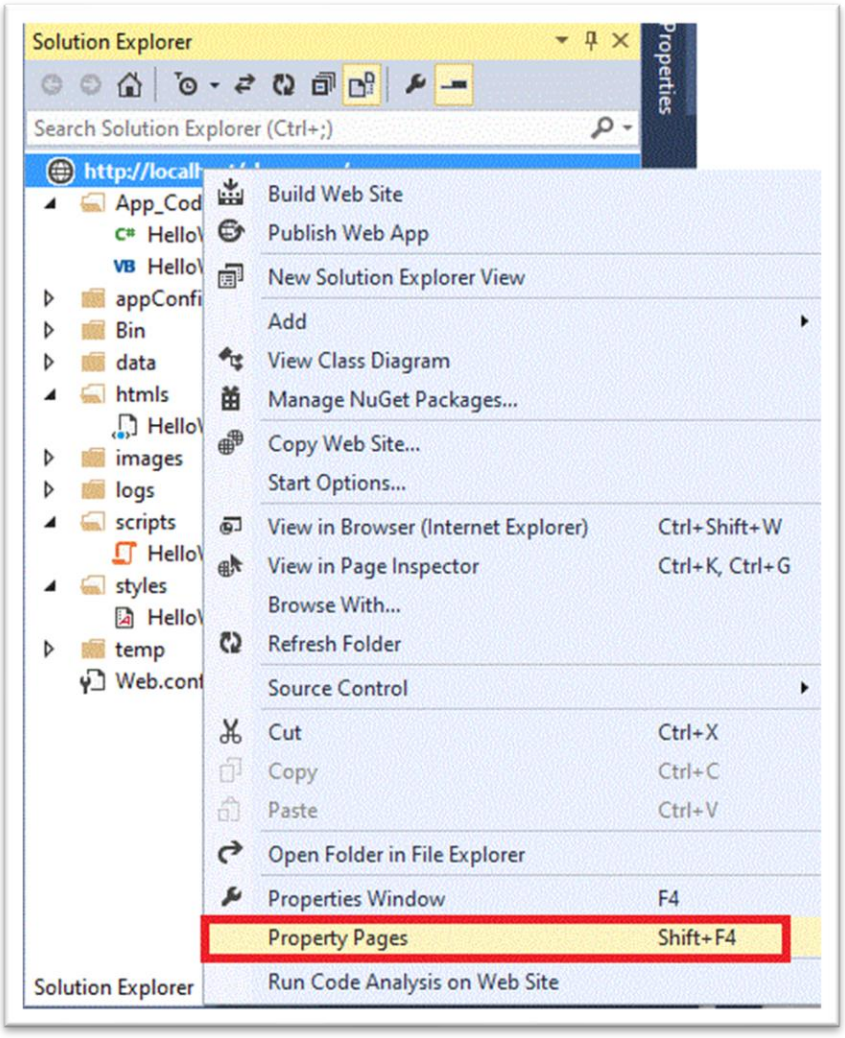
B. "html" extention in "htmls" folder

Fig3. Helloworld.html

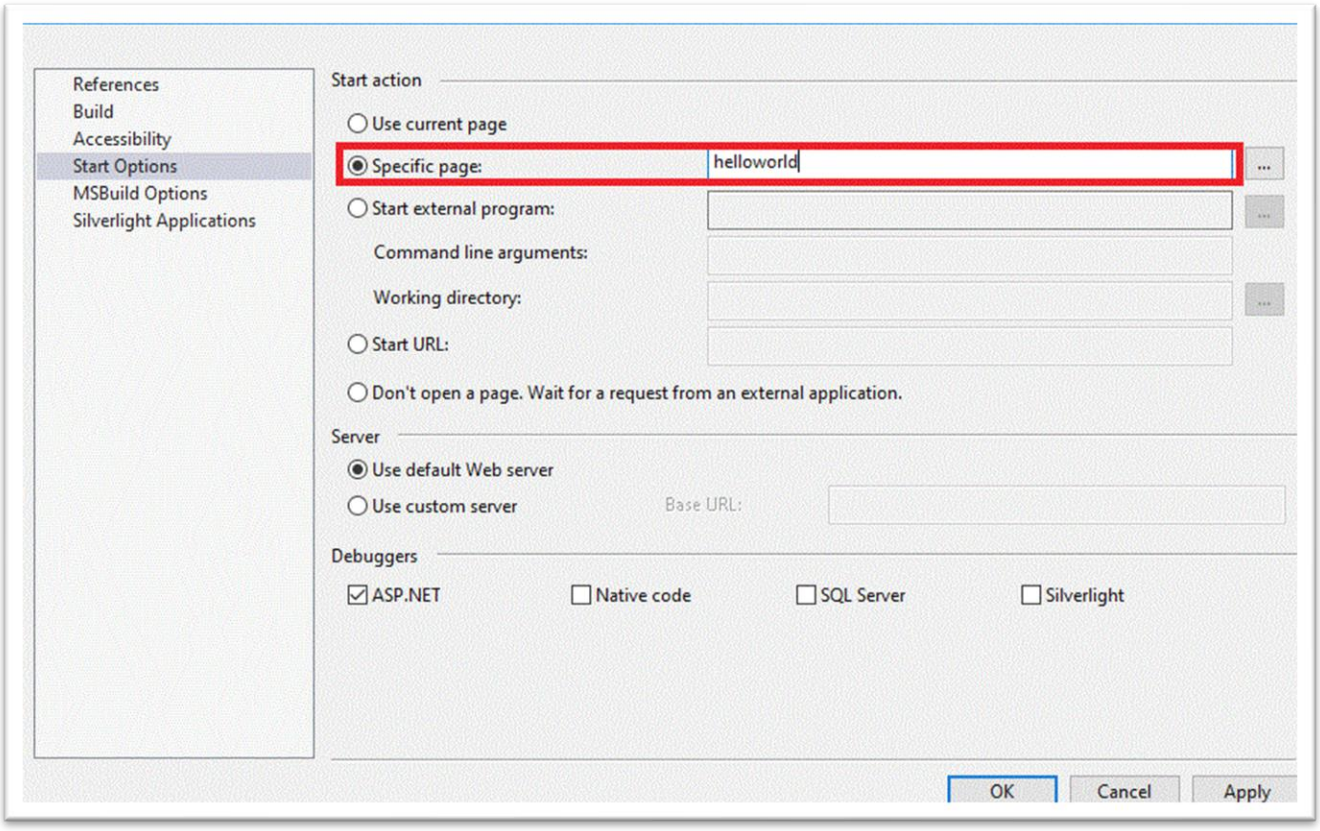
```
<h2>HelloWorld</h2>
<br>
```

C. Set starting page in web application property

- In solution explore view, click right mouse button on web project
- Click "Property Pages" in the menu



- Select "Specific page", enter "HelloWorld". And click "OK" button

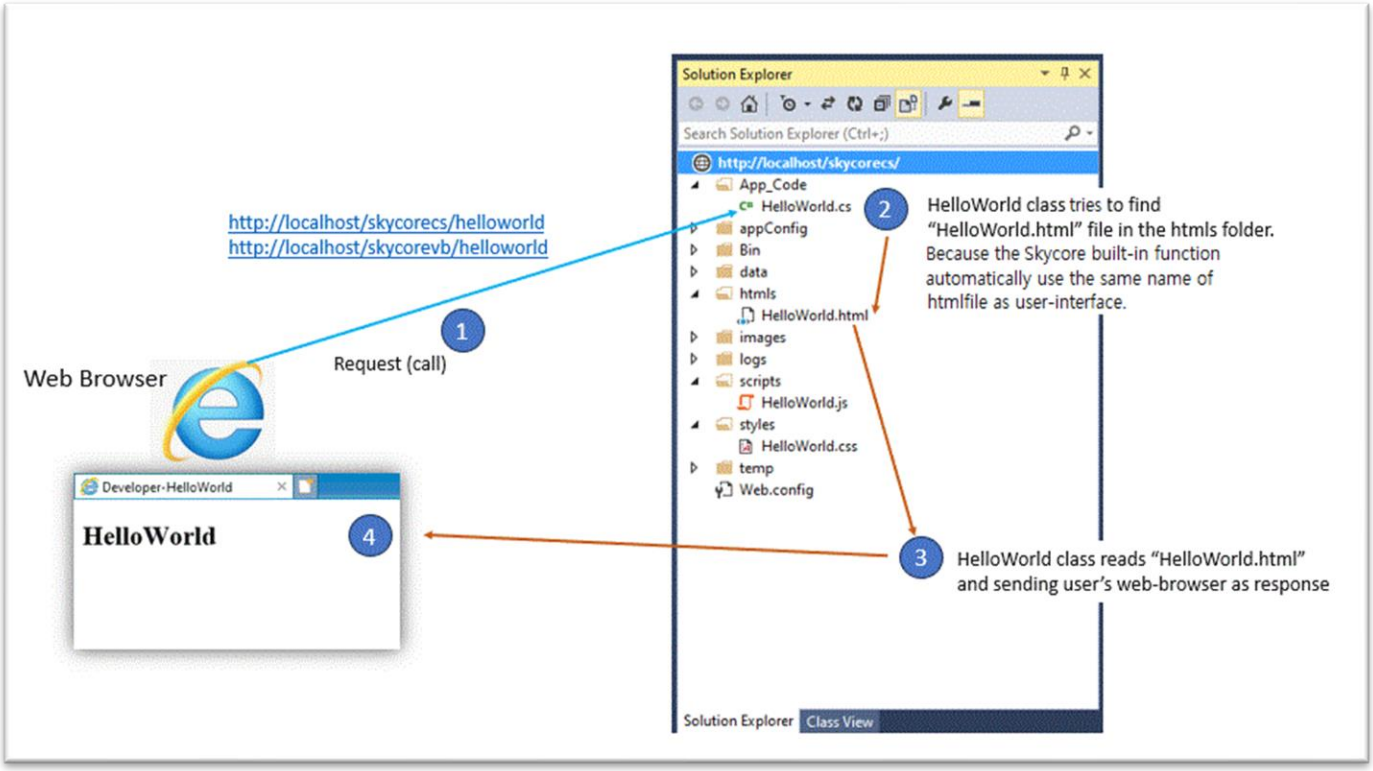


D. Run web application

- It shows "Hello World" in web browser

E. How it works ?

- User's web browser calls directly "helloworld" class in "App_Code" folder (by built-in skycore handler)
- The "helloworld" class finds the html file as same name (helloword.html) in "html" folder
- If the file exists, it reads "helloworld.html" and send it to user's browser and the browser displays the html
- If there is the same name of js file(helloworld.js) in script folder, it will be added in the head of "helloworld.html" before sending it to user's browser
- If there is the same name of css file(helloworld.css) in style folder, it will be added in the head of "helloworld.html" before sending it to user's browser



- If it doesn't work, please check if handler configuration is correct in web.config file

```
<system.webserver>
        <handlers>
                <add name="skycore" verb="*" path="*" type="skycore.IHandler" />
        </handlers>
        ..............
```

F. Implementing a simple Postback method

- The following example shows how to implement **Postback method** easily when you post to the server.
- When user's web browser calls a webpage in SkyCore platform, it automatically adds the same name of js file as webpage name.
- it will be declared in the head of "helloworld.html" as javascript link.

(i) Add a javascript function in Helloworld.html

Fig4. add "ClickMe" event in Helloworld.html

```
<h2>HelloWorld</h2>
<br>
<div onclick="ClickMe()" style="cursor:pointer; text-decoration:underline;">Click Me</div>
<br>
<div id="pholder" style="border:1px solid #0094ff; width:300px; height:100px;"> </div>
```

In this example, HelloWorld.html has been added a div element with "pholder" id.
This Postback method will show "Hi HelloWorld" in the "pholder" element as a server response.

Fig5. add "ClickMe" function in Helloworld.js at "scripts" folder

```
function ClickMe() {
   $WaitOn();
   $ApiRequest($fn(arguments));
}
```

In this example, it shows "Wait Cursor" and invokes the "ClickMe" postback function.
It is wating for a response from a web server. "Wait Cursor" will be turned off automatically when postback function calls.

- $WaitOn() : Built-In function, it shows "Wait Cursor".

- $WaitOff() : Built-In function, it turns off "Wait Cursor".
- $ApiRequest(n); : Built-In function, it invokes a Postback function.
- $fn(arguments); : Built-In function, it returns a current-function-name.
- $fn(arguments) = "ClickMe" in this example. (n): function name

(ii) Add a Postback function in Helloworld.cs or Helloworld.vb

Fig6. add "ClickMe" function in Helloworld.cs

```
using System;
using skycore;
public class HelloWorld:WebPage
{
   public HelloWorld()
           {
           }

   public ApiResponse ClickMe() {
      ApiResponse _ApiResponse = new ApiResponse();
      _ApiResponse.SetElementContents("pholder", "Hi HelloWorld");
      return _ApiResponse;
   }
}
```

Fig7. add "ClickMe" function in Helloworld.vb

```
Imports Microsoft.VisualBasic
Imports skycore
Public Class HelloWorld
   Inherits WebPage
   Public Function ClickMe() As ApiResponse
      Dim _ApiResponse As New ApiResponse
      _ApiResponse.SetElementContents("pholder", "Hi HelloWorld")
      Return _ApiResponse
```

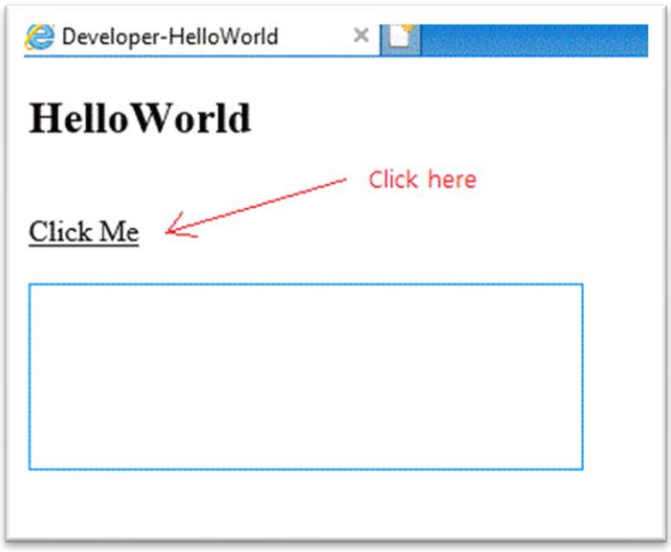
```
    End Function

End Class
```

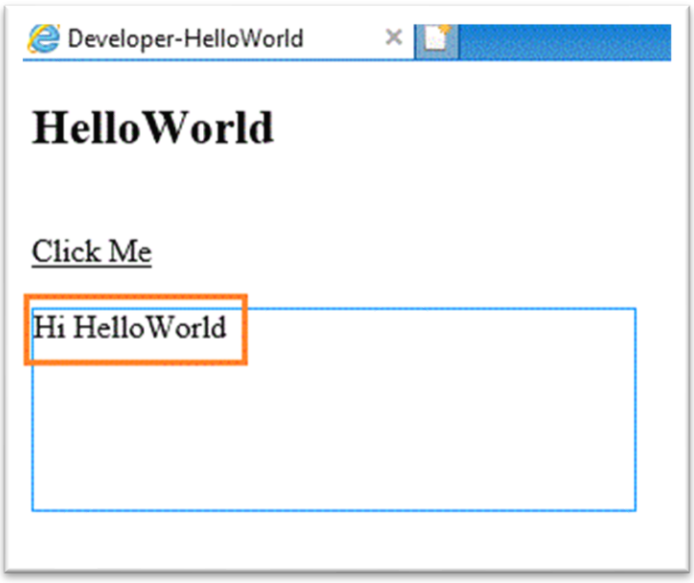
Once "$ApiRequest" in javascript calls "ClickMe" postback method, the "ClickMe" method must be declared in the HelloWorld class. Using built-in "ApiResponse" class, we can directly respond back to user's web browser(client). There are several built-in functions to respond to client. It will explain those in the another chapter.

- ApiResponse.SetElementContents("pholder", "Hi HelloWorld") : Using the "SetElementContents" method, it sets the element value (id="pholder") to "Hi HelloWorld".

G. Run Web Application

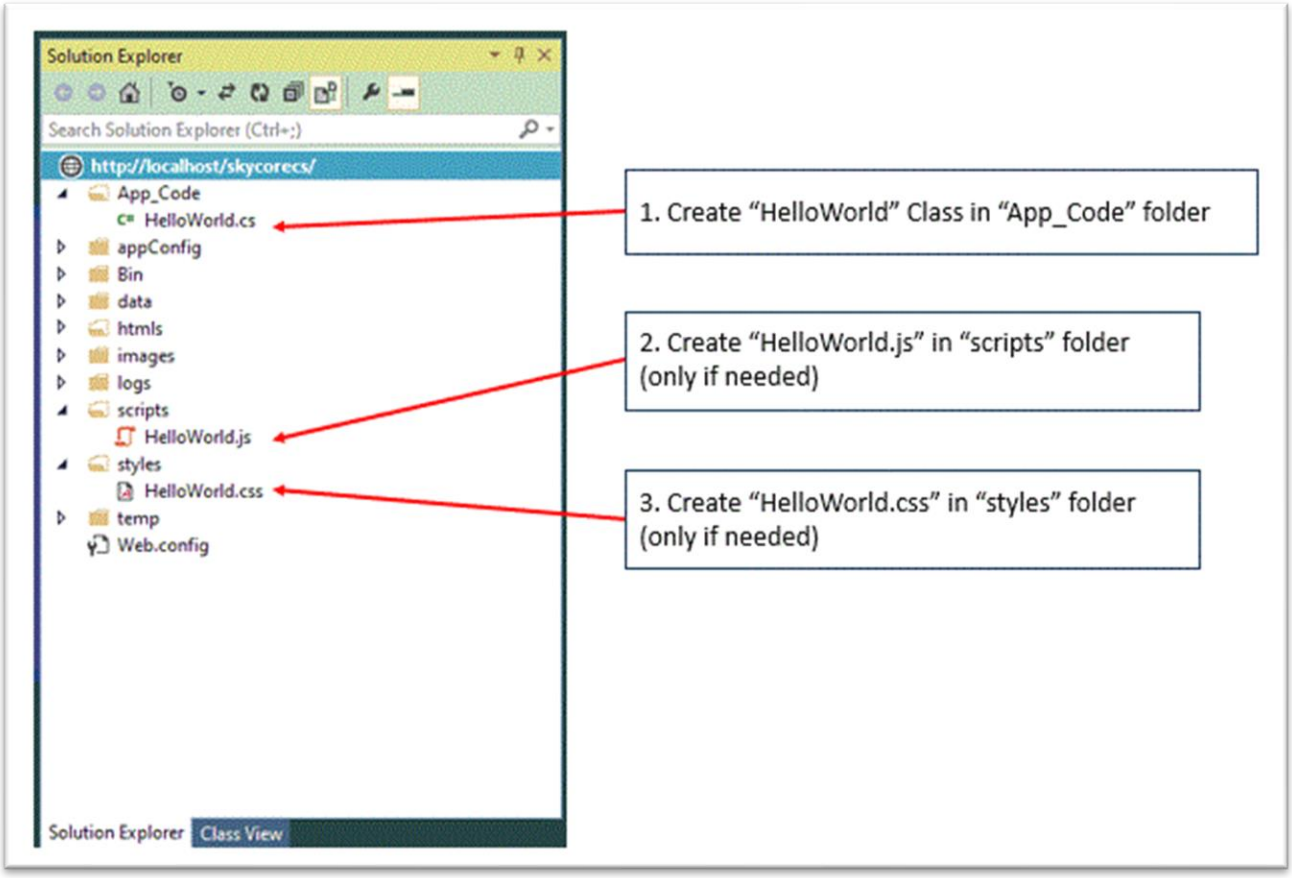
When the "Click Me" is clicked, the "ClickMe" method send user's web browser "Hi HelloWorld" (txt/html) and it will show the text in the screen.





2. Using "Webpage" class (Built-In) without a html template

If the webpage name is "Helloworld", it is supposed to have the same file names as
- "cs/vb" extention in "App_Code" folder
- "js" extention in "script" folder (if it needs javascript code for "Helloworld" webpage)
- "css" extention in "styles" folder (if it needs styles for "Helloworld" webpage)



A. "cs/vb" extention in "App_Code" folder

Fig1. Helloworld.cs

```
using System;
using skycore;
public class HelloWorld:WebPage
{
   public HelloWorld()
          {
          }
```

```
  public override void OnInitialized()
  {
     HtmlDoc.HtmlBodyText = @"<h2>HelloWorld</h2> " +
                    @"<br> " +
                    @"<div onclick=""ClickMe()"" style=""cursor:pointer; text-decoration:underline;"">Click Me</div> " +
                    @"<br>" +
                    @"<div id=""pholder"" style=""border:1px solid #0094ff; width:300px; height:100px;""> </div> ";

  }

  public ApiResponse ClickMe() {
     ApiResponse _ApiResponse = new ApiResponse();
     _ApiResponse.SetElementContents("pholder", "Hi HelloWorld");
     return _ApiResponse;
  }
}
```

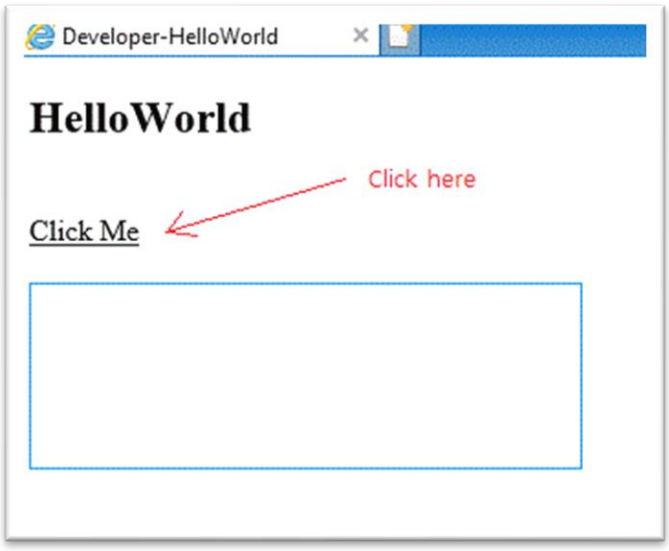
Fig2. Helloworld.vb

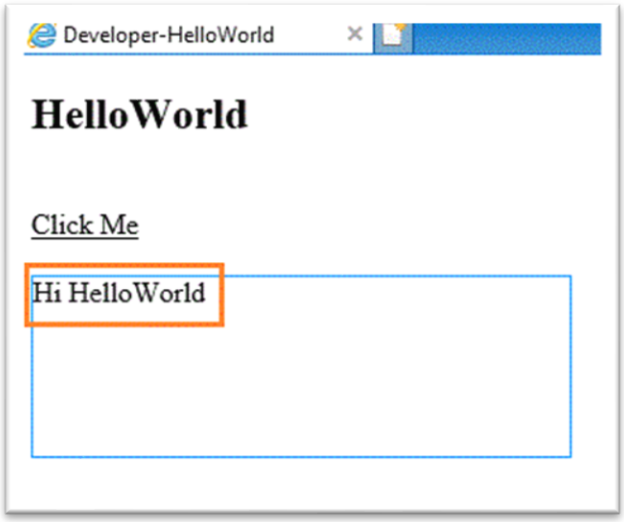
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class HelloWorld
  Inherits WebPage

  Public Overrides Sub OnInitialized()
     HtmlDoc.HtmlBodyText = _
          "<h2>HelloWorld</h2> " +
          "<br> " +
          "<div onclick=""ClickMe()"" style=""cursor:pointer; text-decoration:underline;"">Click Me</div> " +
```

```
            "<br> " +
            "<div id=""pholder"" style=""border:1px solid #0094ff; width:300px; height:100px;""> </div> "

    End Sub

    Public Function ClickMe() As ApiResponse
        Dim _ApiResponse As New ApiResponse
        _ApiResponse.SetElementContents("pholder", "Hi HelloWorld")
        Return _ApiResponse
    End Function

End Class
```

B. Run web application

When a web page is initilized, we can set page Html at HtmlDoc.HtmlBodyText property.





LogIn Webpage

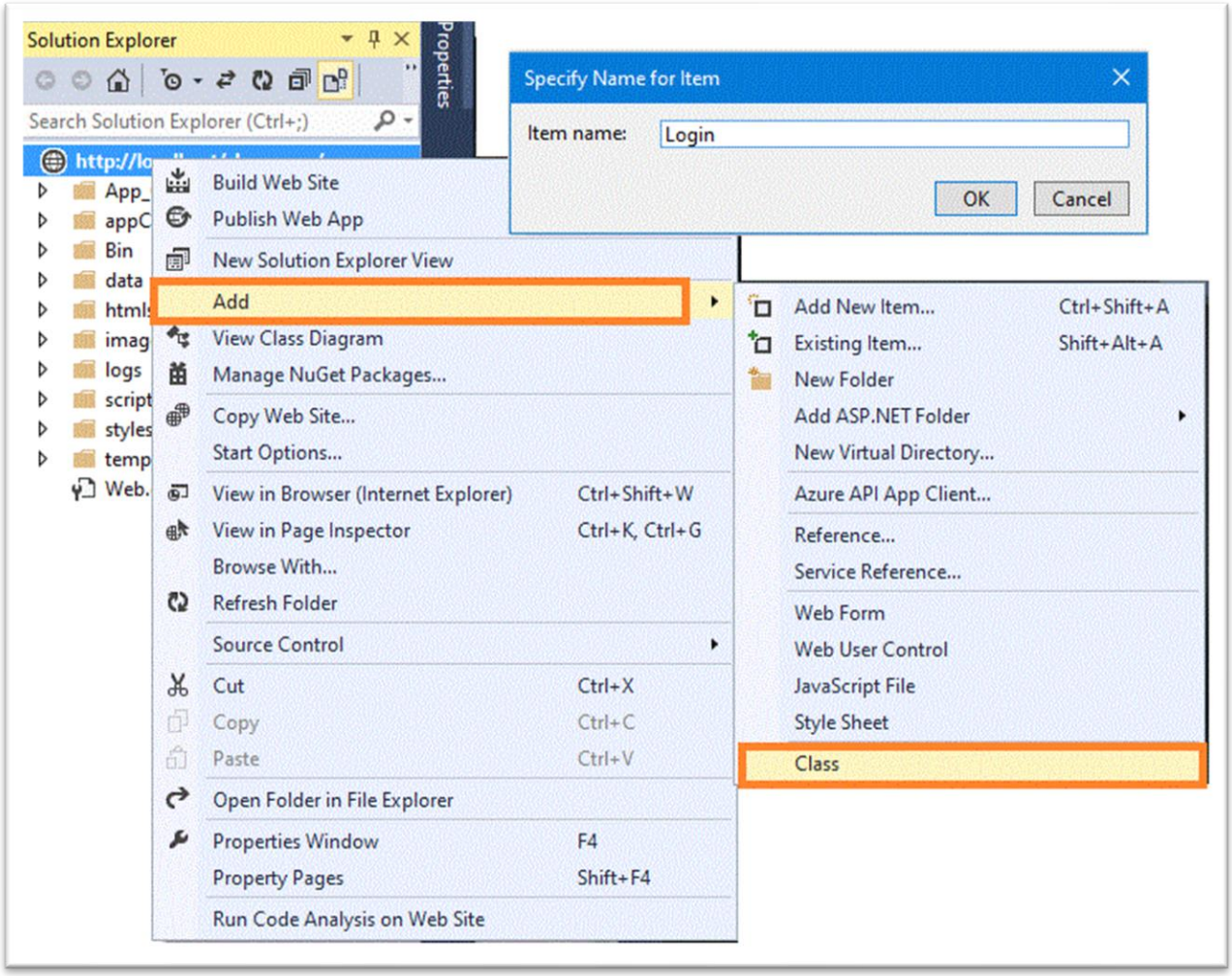
In order to create a webpage, You have to

1. Create "Login" class file in the ***App_Code*** system folder
2. Using/Imports "skycore" namespace on top of the code
3. Inherits built-in "WebPage" class
4. Create "Login.html" file in the ***htmls*** system folder
5. Create "Login.js" file in the ***scripts*** system folder
6. Create "Login.css" file in the ***styles*** system folder

1. Create "Login" class

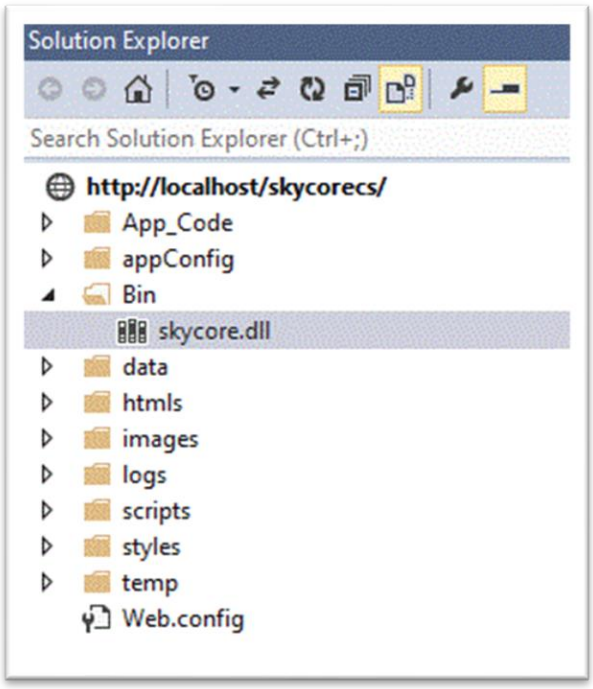
Right mouse click on web project name, click "add" menu and "class". Enter "Login" on popup window.

Then, "Login.cs" or "Login.vb" will be created in App_Code folder.



2. Using/Imports "skycore" namespace on top of the code

Check if there is "skycore.cll" in the bin folder



Import/Using "skycore" statements must be placed in a class before references to any identifiers

Fig1. Login.cs

```
using System;
using skycore;
 public class Login
{
        public Login()
        {
        }
}
```

Fig2. Login.vb

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class Login

End Class
```

3. Inherits built-in "WebPage"

Add inherits statement in the class

Fig3. Login.cs

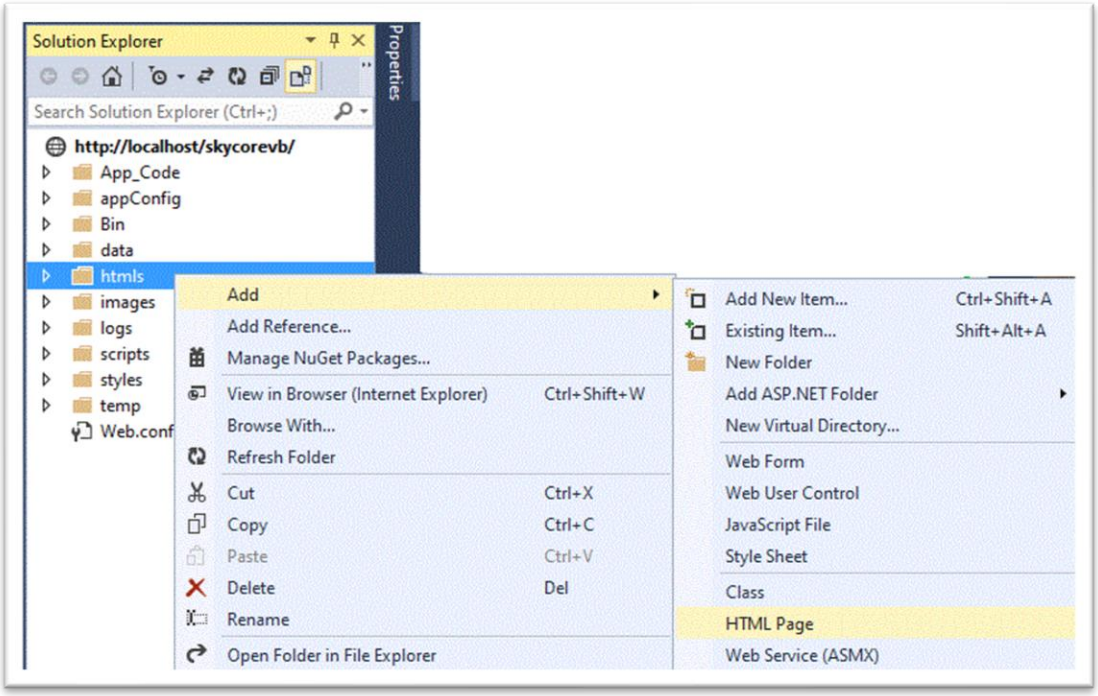
```
using System;
using skycore;
public class Login : WebPage
{
        public Login()
        {
        }
}
```

Fig4. Login.vb

```
Imports Microsoft.VisualBasic
Imports skycore
Public Class Login
   Inherits WebPage

End Class
```

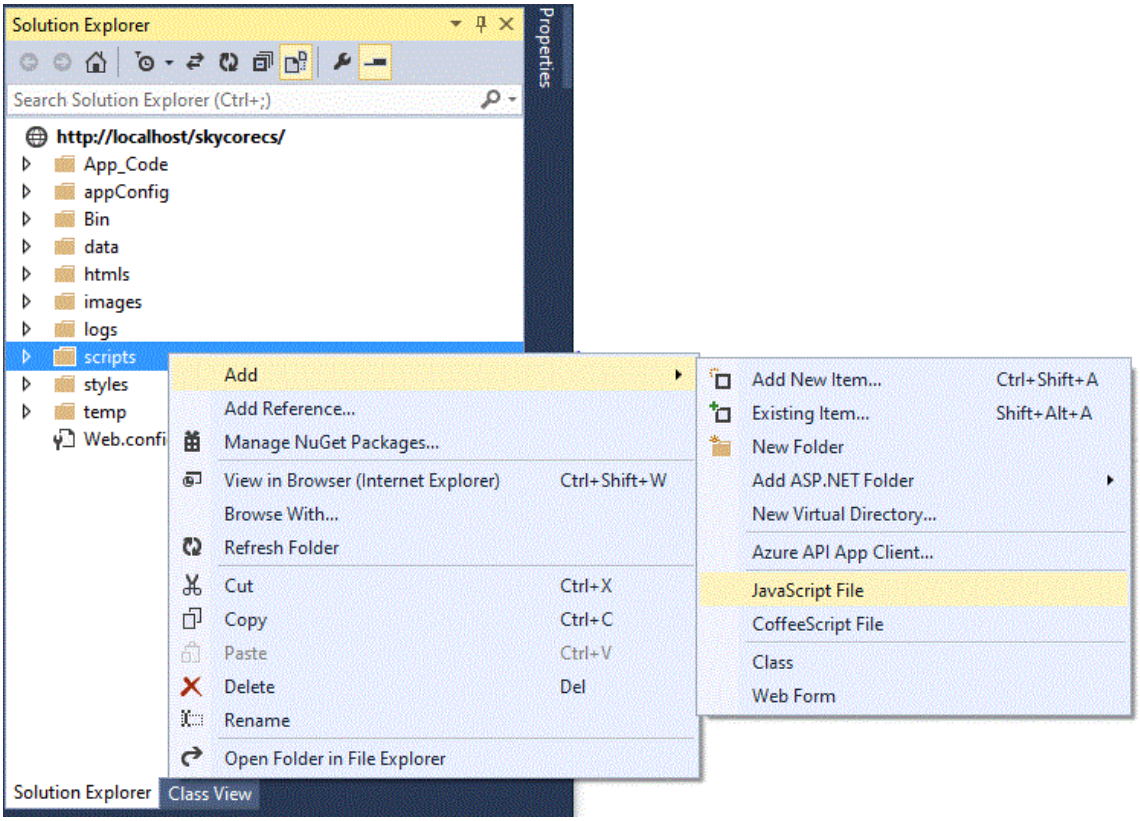
4. Create "Login.html" in htmls folder

Create html file only when you need



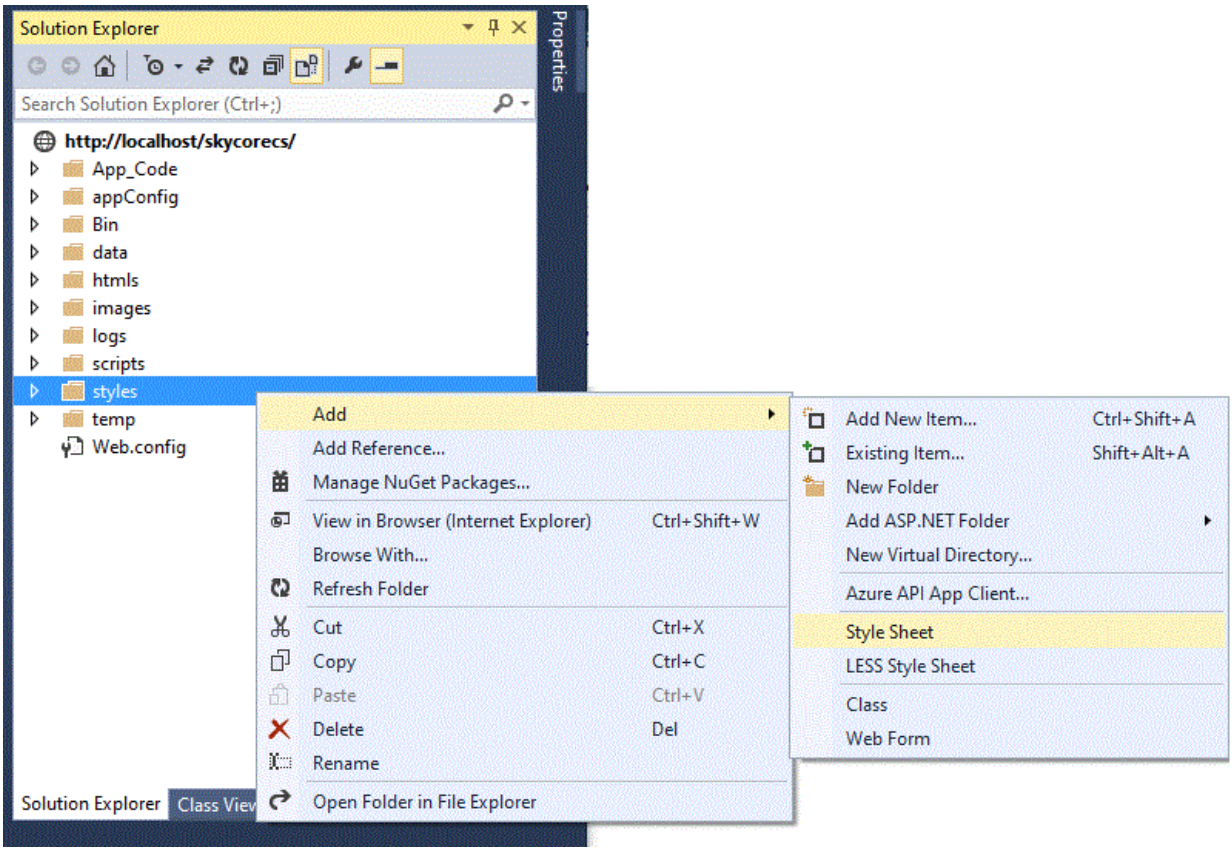
5. Create "Login.js" in scripts folder

Create javascript file only when you need

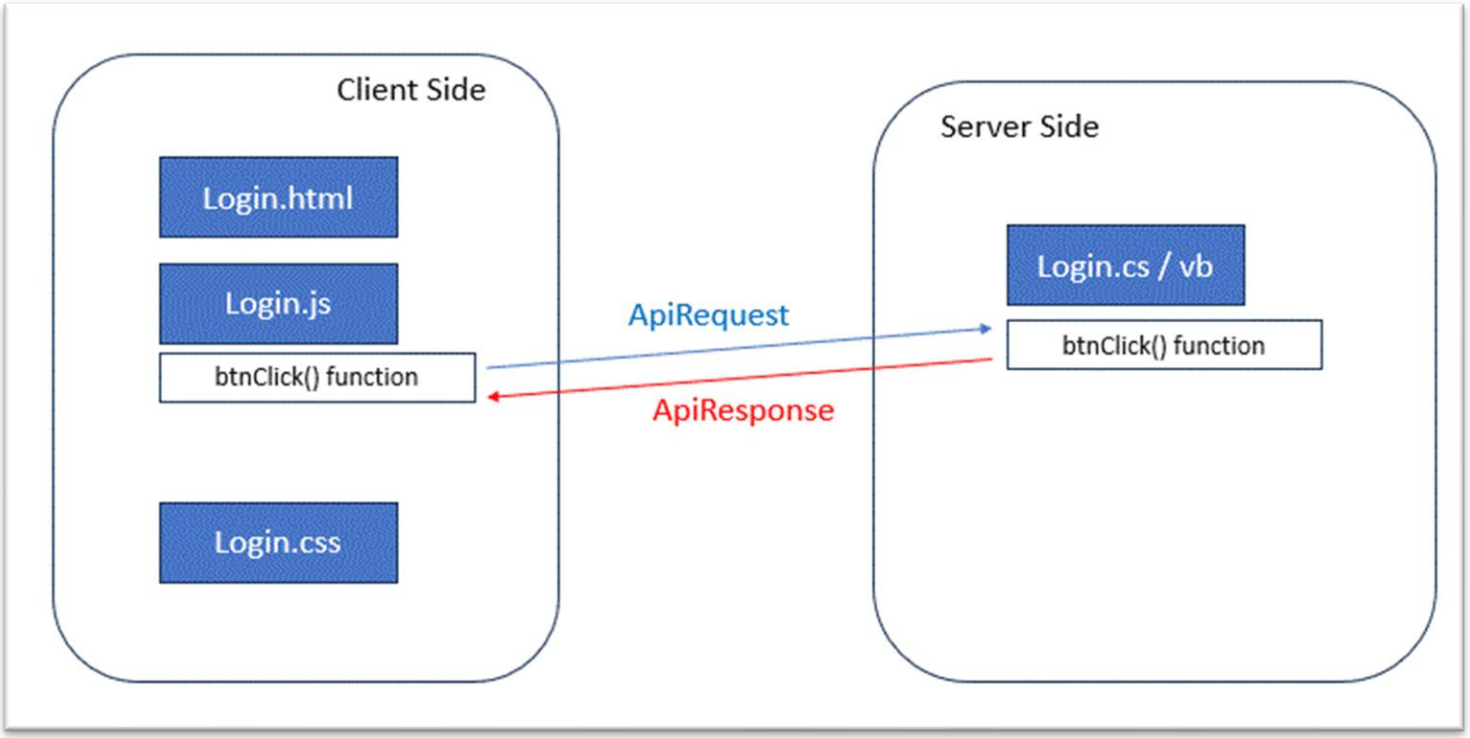


6. Create "Login.css" in styles folder

Create stylesheet file only when you need



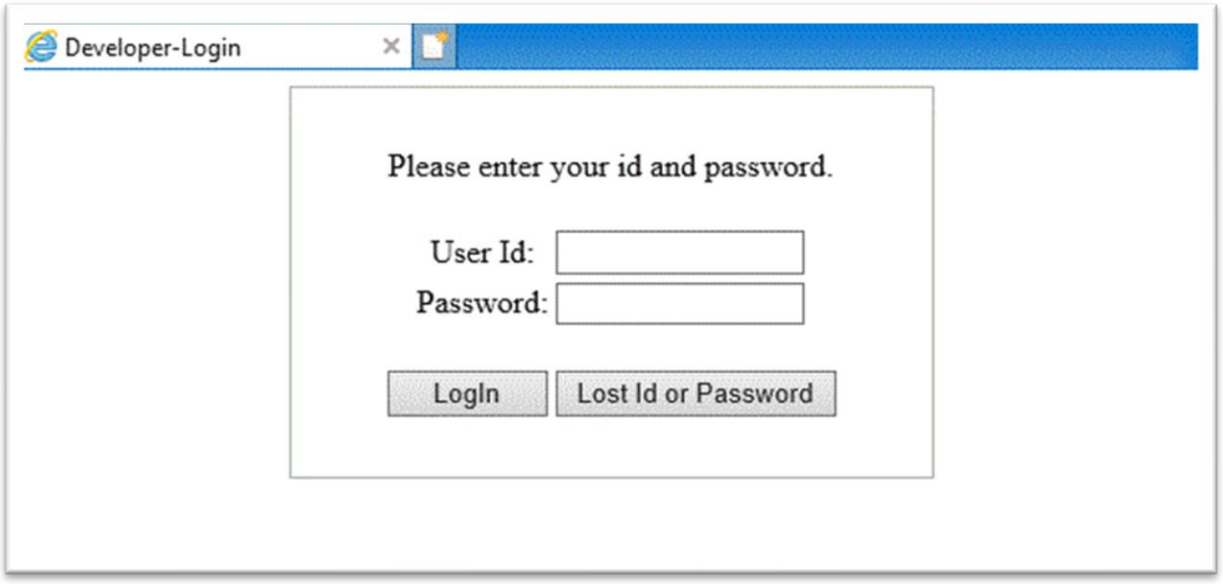
7. Architectural Diagram



Html - User Interface

1. Create User-Interface

- Create "Login.html" file in the ***htmls*** system folder
- UserId, User Password textbox and "Login" button
- When clicks "Login" button, it triggers "btnClick()" function in javascript



2. Html Design

Design the "Login.html" file

It doesn't need to have "<head>" and "<body>" tags.

Fig1. Login.html

```
<div id="loginbox">
  <div style="margin:auto; width:300px; border:1px solid #aaa; text-align:center; padding:10px;">
    <div style="margin:20px;">Please enter your id and password.</div>
    <table style="margin:auto;">
      <tr>
        <td>User Id:</td>
        <td><input type="text" style="width:120px;" id="userid" /></td>
      </tr>
      <tr>
        <td>Password:</td>
```

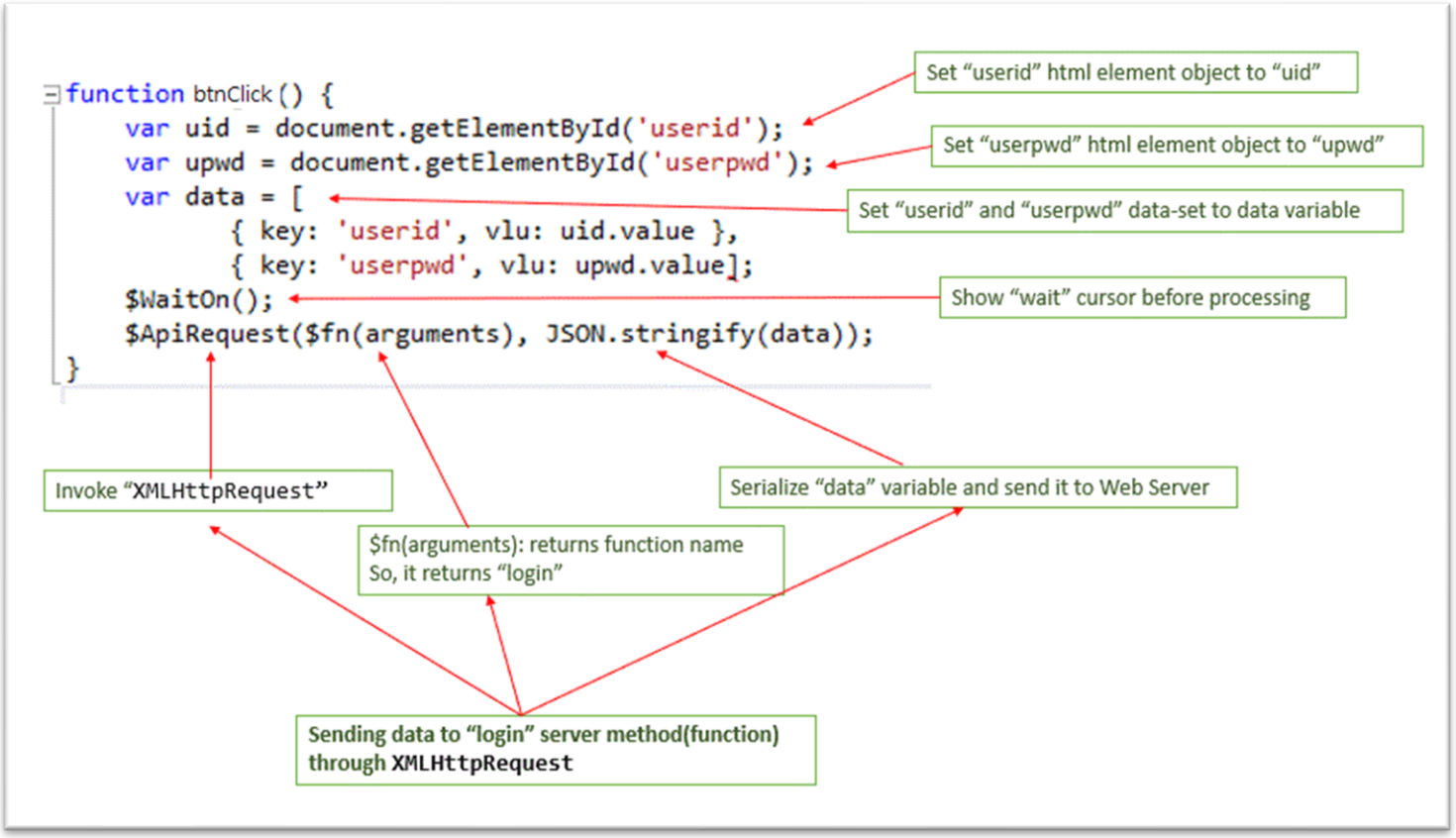
```
            <td><input type="text" style="width:120px;" id="userpwd" /></td>
        </tr>
    </table>
    <div style="margin:20px;">
        <input type="button" style="width:80px;" value="LogIn" onclick="btnClick()" />
        <input type="button" style="width:140px;" value="Lost Id or Password" />
    </div>
  </div>
</div>
```

Javascript Program

- Create "Login.js" file in the ***scripts*** system folder
- Implement "btnClick()" function
- When clicks "Login" button in Login.html, it triggers "btnClick()" function in javascript

Fig1. Login.js

```
function btnClick() {
   var uid = document.getElementById('userid');
   var upwd = document.getElementById('userpwd');
   var data = [
         { key: 'userid', vlu: uid.value },
         { key: 'userpwd', vlu: upwd.value }
   ];
   $WaitOn();
   $ApiRequest($fn(arguments), JSON.stringify(data));
}
```



Stylesheet

- Only if needed.
- Create "Login.css" file in the ***styles*** system folder

Fig1. Login.css

```
body {
    margin:0px;
}
```

Server-Side Program

- Create "Login" class file in the ***App_Code*** system folder
- The "btnClick" javascript function calling a method(server function),
- the method("btnClick") is supposed to be implemented in the server-side.

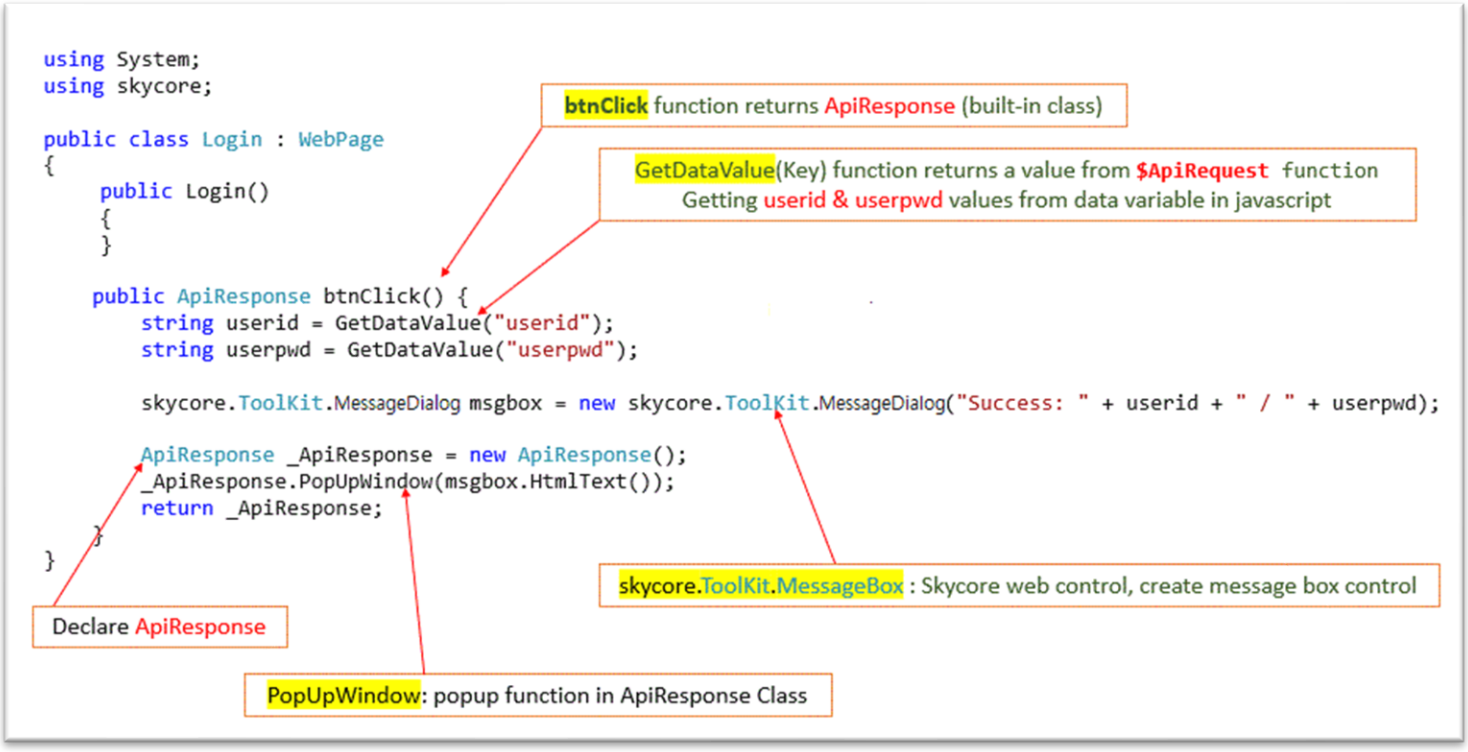


Fig1. Login.cs

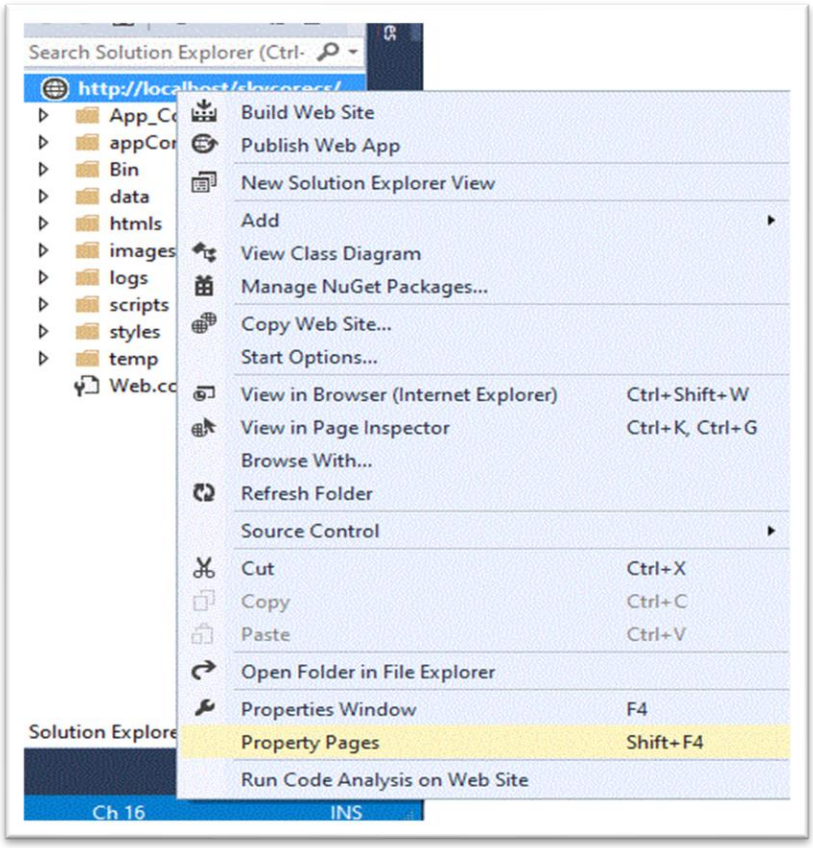
```
using System;
using skycore;

public class Login : WebPage
{
        public Login()
        {
        }

   public ApiResponse btnClick() {
      string userid = GetDataValue("userid");
      string userpwd = GetDataValue("userpwd");

      skycore.ToolKit.MessageDialog msgbox = new skycore.ToolKit.MessageDialog("Success: " + userid + " / " + userpwd);

      ApiResponse _ApiResponse = new ApiResponse();
      _ApiResponse.PopUpWindow(msgbox.HtmlText());
      return _ApiResponse;
   }
}
```

Fig2. Login.vb

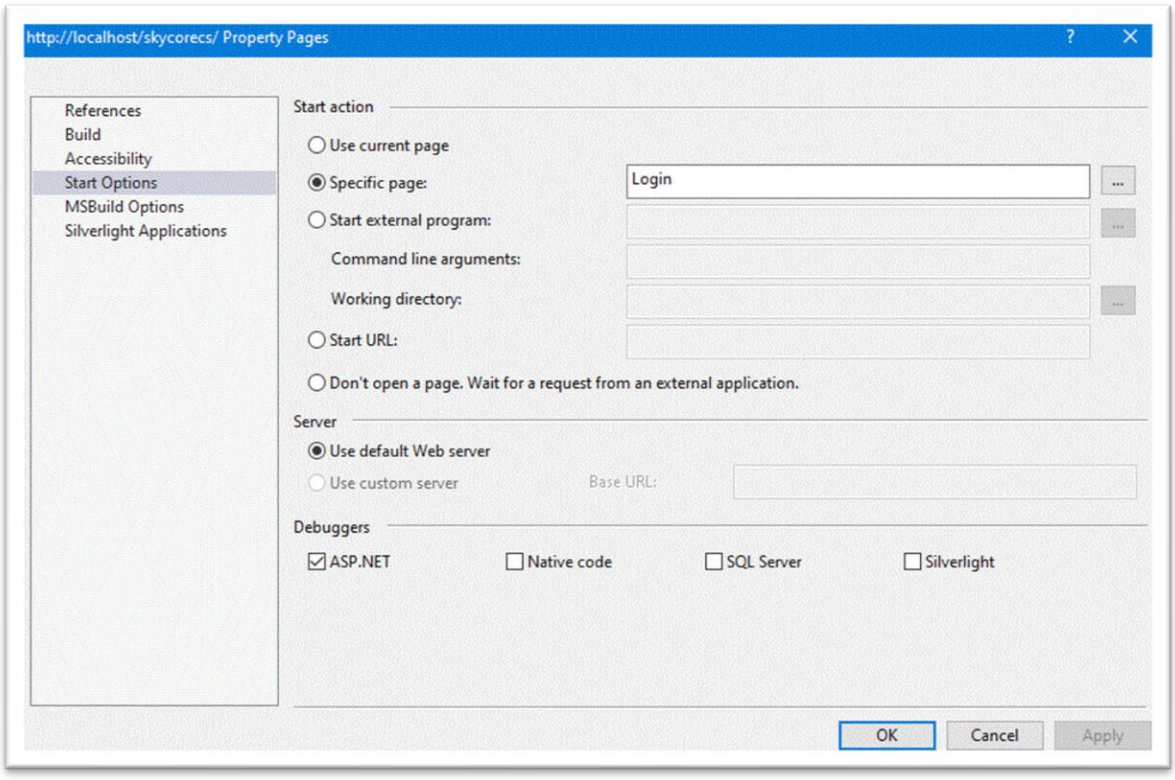
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class Login
    Inherits WebPage

    Public Function btnClick() As ApiResponse
        Dim userid As String = GetDataValue("userid")
        Dim userpwd As String = GetDataValue("userpwd")

        Dim msgbox As New skycore.ToolKit.MessageDialog("Success: " + userid + " / " + userpwd)

        Dim _ApiResponse As New ApiResponse
        _ApiResponse.PopUpWindow(msgbox.HtmlText)
        Return _ApiResponse
    End Function

End Class
```

Run Application

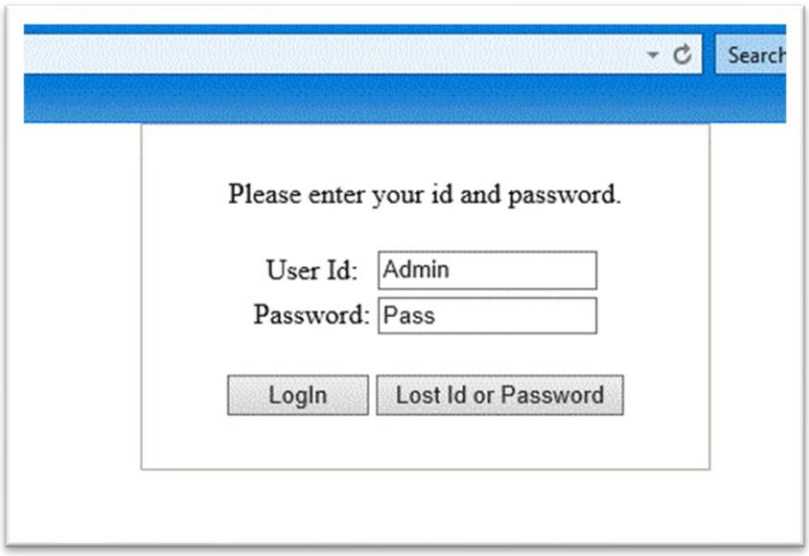
- Right mouse click on web project name, select "property pages" menu



- Set specific page to "Login" as start page



- Run application



- Click "Login" button, a PopUpMessage box will be displayed.

WebPage Class

- A web page needs "Inherit WebPage" statement in SkyCore Framework.
- In other words, a web page needs to use built-in WebPage class in SkyCore.dll.
- The "WebPage" class working with "Webcore" class provides many useful functions.
- A SkyCore web page running, the page goes through a life cycle which has a series of process steps.
- Within each stage of the life cycle, the page raises events that you can handle to run your own code.

WebPage Life Cycle

Event Stage	Description
OnInitialized	Web page is initialized. As default, a page loads html file and set html contents to "HtmlBodyText" property in HtmlDoc object. If a developer sets own html contents to "HtmlBodyText", that should be displayed in the web browser instead of html file.
BeforRender	Web page's head and body elements are ready to render. A developer can add html elements in the stage.
AfterRender	Web page's Html context is created. "HtmlContext" property is filled with Html contents. A developer can manipulate html text directly.

WebPage Properties

Property Name	Description
RequestParam	Parameter string from Caller.
HtmlDoc	Object handles Html contents
HtmlContext	Web page's final Html contents after rendering.
ToTranslate	Boolean, if a web page needs translation, then True else False.
AllowSkyLink	Boolean, if it allows to access to the web page from other SkyLink web applications.

WebPage Methods

Method Name	Description
SetTranslatorPrefix(Prefix)	set translator's prefix. - Prefix: "{$" is a default.
SetTranslatorSuffix(Suffix)	set translator's prefix. - Suffix: "$}" is a default.
AddWordToTranslator(KeyWord, TranslatedWord, IsoCode)	Add a word to translation list. - KeyWord: a target word to be translated - TranslatedWord: a word after translation - IsoCode: language ISO code. "*"(applied to all language) is a default. *eg. {$beforeTranslated$} -> afterTranslated*
TranslateHtmlText(HtmlText)	returns translated text from HtmlText.

1. Create a web page without Html file

A developer can create a web page without Html file by using HtmlBodyText on OnInitialized stage

Fig1. Login.cs

```
using System;
using skycore;
public class Login : WebPage
{
   public Login()
   {
   }
   public override void OnInitialized()
   {
      string HtmlText = @"<table style=""margin:auto;""> " +
                  @"    <tr> " +
                  @"       <td>User Id:</td> " +
                  @"       <td><input type=""text"" style=""width:120px;"" id=""userid"" /></td> " +
                  @"    </tr> " +
```

```
                @"    <tr> " +
                @"       <td>Password:</td> " +
                @"       <td><input type=""text"" style=""width:120px;"" id=""userpwd"" /></td> " +
                @"    </tr> " +
                @"</table> ";
        HtmlDoc.HtmlBodyText = HtmlText;
    }
}
```

2. Manupulate Html file

Fig2. Login.cs

```
using System;
using skycore;
public class Login : WebPage
{
    public Login()
    {}
    public override void OnInitialized()
    {
        string HtmlText = ReadHtmlFile("login");
        HtmlText = HtmlText.Replace("User Id:", "User Name:");
        HtmlDoc.HtmlBodyText = HtmlText;
    }
}
```

WebCore Class

- WebCore class provides basic HttpRequest information and additional useful methods.

WebCore Properties

Property Name	Description
WebAppName	Returns Web Application Title defined in application.cfg. (AppConfig.Application.Name)
WebAppVersion	Returns Web Application Version defined in application.cfg. (AppConfig.Application.Version)
WebAppRelease	Returns Web Application published date defined in application.cfg. (AppConfig.Application.ReleaseDate)
WebAppRunMode	Returns Web Application RunMode defined in application.cfg. (AppConfig.Application.RunMode) 0.Demo, 1.Double Authentication, 2.Traditional Authentication
WebAppLicense	Returns Web Application License Key defined in application.cfg. (AppConfig.Application.License)
WebAppHome	Returns Web Application Homepage Name defined in application.cfg. (AppConfig.Application.Home)
AppName	Returns Web Application Name in Internet Information Server
EncryptKey	Returns Global Encryption Key to be used by application developers in application.cfg. (AppConfig.Application.Encryptkey)
Encrypt	Returns Global Encryption object to be used by application developers in application.cfg.
VirtualPath	Returns Application Virture Path (*eg. "http://localhost/applicationname/"*)
PhysicalPath	Returns Application Physical Path (*eg. "C:\inetpub\wwwroot\applicationname\"*)
ConfigFolder	Returns Application Configuration File Path (*eg. "C:\inetpub\wwwroot\applicationname\appConfig\"*)
IsoCode	Returns Application Main Language ISO Code.(*eg. en-us*)

DefaultFont	Returns Application Default Font Family Name.(*eg. Times New Roman*)
PageTimeout	Returns Application Page Timeout (seconds)
WaitImageUri	Returns page wait cursor location. (*eg. "http://localhost/applicationname/images/ani_wait.gif"*)
CodeFolder	Returns Application Code files' folder defined in application.cfg. (AppConfig.Folders.Code) (*eg. "C:\inetpub\wwwroot\applicationname\Code\"*)
HtmlFolder	Returns Application Html files' folder in application.cfg. (AppConfig.Folders.Html) (*eg. "C:\inetpub\wwwroot\applicationname\htmls\"*)
ScriptFolder	Returns Application Script files' folder in application.cfg. (AppConfig.Folders.Script) (*eg. "C:\inetpub\wwwroot\applicationname\scripts\"*)
ImageFolder	Returns Application Image files' folder in application.cfg. (AppConfig.Folders.Image) (*eg. "C:\inetpub\wwwroot\applicationname\images\"*)
LogFolder	Returns Application Log files' folder in application.cfg. (AppConfig.Folders.Log) (*eg. "C:\inetpub\wwwroot\applicationname\logs\"*)
TempFolder	Returns Application Temporary files' folder in application.cfg. (AppConfig.Folders.Temp) (*eg. "C:\inetpub\wwwroot\applicationname\temp\"*)
DataFolder	Returns Application Data files' folder in application.cfg. (AppConfig.Folders.Data) (*eg. "C:\inetpub\wwwroot\applicationname\data\"*)
BinFolder	Returns Application Reference files' folder in application.cfg. (AppConfig.Folders.Bin) (*eg. "C:\inetpub\wwwroot\applicationname\bin\"*)
HtmlPath	Returns Application Html files' location in application.cfg. (*eg. "http://localhost/applicationname/htmls/"*)
ScriptPath	Returns Application Script files' location in application.cfg. (*eg. "http://localhost/applicationname/scripts/"*)
StylePath	Returns Application Style files' location in application.cfg. (*eg. "http://localhost/applicationname/styles/"*)
ImagePath	Returns Application Image files' location in application.cfg. (*eg. "http://localhost/applicationname/images/"*)

LogPath	Returns Application Log files' location in application.cfg. (*eg. "http://localhost/applicationname/logs/"*)
TempPath	Returns Application Temporary files' location in application.cfg. (*eg. "http://localhost/applicationname/temp/"*)
DataPath	Returns Application Data files' location in application.cfg. (*eg. "http://localhost/applicationname/data/"*)
BinPath	Returns Application Reference files' location in application.cfg. (*eg. "http://localhost/applicationname/bin/"*)
ScriptAliasPath	Returns Application Script files' location alias in application.cfg. (*eg. "/applicationname/scripts/"*)
StyleAliasPath	Returns Application Style files' location alias in application.cfg. (*eg. "/applicationname/styles/"*)
ImageAliasPath	Returns Application Image files' location alias in application.cfg. (*eg. "/applicationname/images/"*)
LogAliasPath	Returns Application Log files' location alias in application.cfg. (*eg. "/applicationname/logs/"*)
GlobalParam	Returns Global Parameter set by a developer.
GlobalAuth	Returns Global Parameter set by system.
GlobalKey	Returns Global Key data set by a developer.
AuthToken	Returns Authentication Token (in Http Request Header) set by a developer.
ReqData	Returns Request Data in $ApiRequest information.
ReqReferrer	Returns Http Request Referrer in Http Request Information.
ClientIPAddress	Returns Client Ip Address in Http Request Information.
ClientLanguage	Returns Client Language Preferences in Http Request Information.
ClientTimeZone	Returns Client Timezone in $ApiRequest information.

ClientTimeOffset	Returns Client Timeoffset in $ApiRequest information.
ClientTime	Returns Client Local Time in $ApiRequest information.
ClientAgent	Returns Client Agent in Http Request Information.
ServerTimeOffset	Returns Server Timeoffset.
ServerTimeZone	Returns Server Timezone.
ServerTime	Returns Server Time.

WebCore Methods

Method Name	Description
PartialPage(Typename, *optional* Parameter)	Get Html Context from another web page. - Typename: another web page name - Parameter: data sent by main web page
ReadHtmlFile(htmlname, *optional* htmlFilePath)	Read all text from Html file. - htmlname: html file name - htmlFilePath: if html file is not in "htmls" folder, must assign a specific folder path
ReadTextFile(FilePath)	Read all text from text file. - FilePath: Text file's full path
CoreAction(functionName)	Call server-side function without defining javascript function. - functionName: server-side function in same type(module)
GetWebEnv(key)	Get appconfig file information. - key: key string to get value. (*eg. GetWebEnv("app.folders.image")*)
WriteLog(logMessage)	Write log file in logs folder. - logMessage: log message.
EncryptString(data)	Encrypt text data. - data : text data to be encrypted.

DecryptString(encdata)	decrypt encrypted data. - encdata : encrypted data.
PageLinkX(pagename, *optional* xparam)	create navigation link with querystring with key x. - pagename : encrypted data. - xparam : optional parameter to be encrypted.
ParamX()	get a x-key value in querystring.
FileSteamParam(filename, filepath)	create a parameter string for downloading file. - filename : download displayname. - filepath : file fullpath to download.
FileSteamLink(FileSteamParam)	create file download link with serialized FileSteamParam. - FileSteamParam : file stream parameter. *eg. ApiResponse.Navigate(FileSteamLink(FileSteamParam))*
QueryValue(QueryStringKey)	get querystring value. - QueryStringKey : Key in querystring
HeaderValue(key)	get httprequest header key value. - key : httprequest header key
ParamValue(key)	get httprequest formdata key value. - key : httprequest formdata key
GetDataValue(key)	get xmlhttprequest data value. - key : XmlHttpRequest formdata key
GetMonthText(date)	get month as text such as "Jan", "Feb", "Mar" .... - date : text date
GetWeekDay(date)	get weekday as text such as "Sun", "Mon", "Tue" .... - date : text date
CreateWorkBook(datatable, filepath, *optional* fileowner)	create a xml excel file. - datatable : datatable - filepath : file fullpath to be created - fileowner : file owner

Translator Class

- Translator Class is one of sub classes in the Web Page Class.
- This class enables a developer to translate a web page to multi language support page.
- Translation based on web browser's language preferences.

Translator Properties

Property Name	Description
KeyPrefix	make an application find the place to be translated. "{$" is a default. *eg. {$beforeTranslated$} -> afterTranslated*
KeySuffix	make an application find the place to be translated. "{$" is a default. *eg. {$beforeTranslated$} -> afterTranslated*
IsoCode	Language preference in Http request "en-US", "ko-KR", "jp-JP"
TDictionary	List<Dictionary> Add translation list A web page is translated automatically after render page

Translator Methods

Method Name	Description
Add(DictionaryKey, DictionaryWord, *optional* IsoCode)	Add a word to translation list. - DictionaryKey: a target word to be translated - DictionaryWord: a word after translation - IsoCode: language ISO code. "*"(applied to all language) is a default. *eg. {$beforeTranslated$} -> afterTranslated*

FileHandler Class

- provides useful functions to handle files.

FileHandler Methods

Method Name	Description
GetDirectoryFiles(Folder, *optional* option)	Get List<string> of all files under a specific folder. - Folder: search folder - option: 0.full name, 1.file name.
CreateZipFromDir(SourceFolder, TargetPath)	create a zip compressed file. - SourceFolder: a folder to create zip file - TargetPath: zip file full path.
WriteByteToFile(filepath, filebyte())	write a file from byte array. - filepath: a file to be created - filebyte: file contents as byte array.
ReadByteFromFile(filepath)	read bytes from file. - filepath: a file full path
ReadObjectFromFile(filepath)	read serialized object from file. - filepath: a file full path
WriteObjectToFile(filepath, object)	write serialized object to file. - filepath: a file full path
IsFileInUse(filepath)	returns true if the file is in use. - filepath: a file full path
DeleteFiles(folder, daysBefor)	delete files x days before in the folder. - folder: folder path - daysBefor: integer
GetDirectoryFiles(folder)	returns all files in the folder as Dictionary<strin, string>. - folder: folder path

ImageHandler Class

- provides useful functions to handle images.

ImageHandler Methods

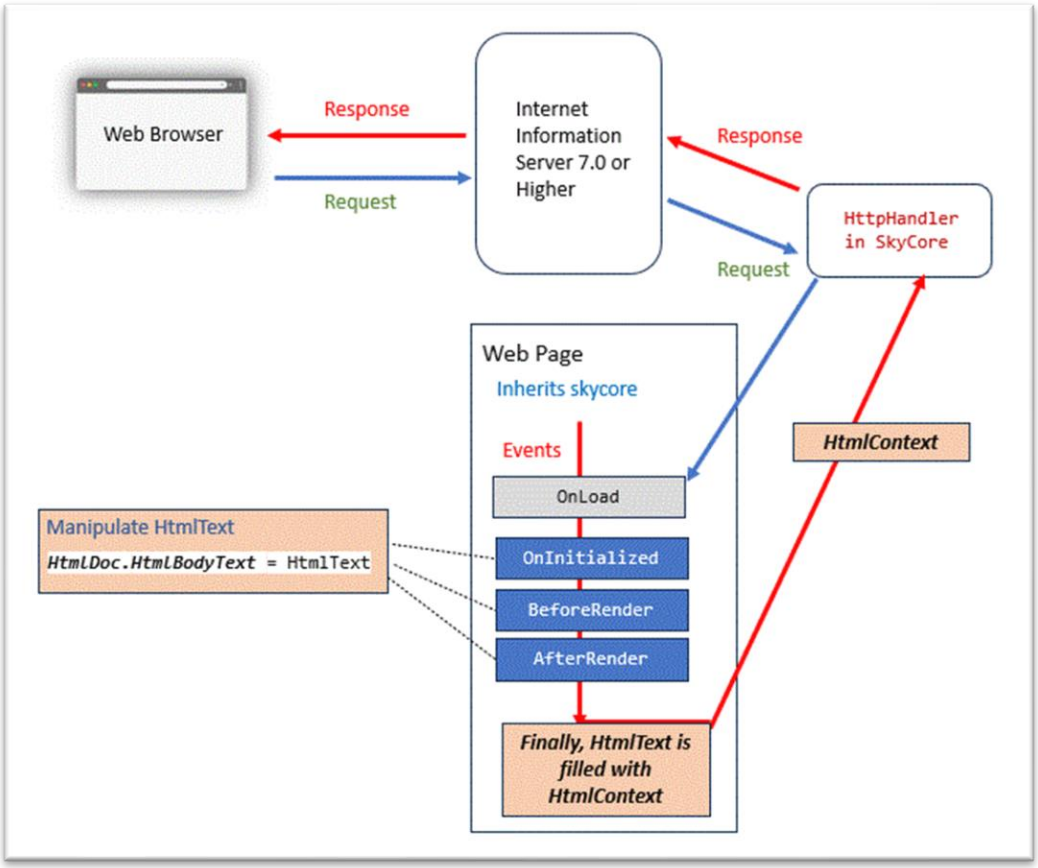
Method Name	Description
ImageRotated(Bitmap)	returns bitmap image orientation. - Bitmap: Bitmap image
RotateImage(Bitmap)	returns rotated Bitmap image. - Bitmap: Bitmap image
RotateImage(srcPath, targetPth)	save a rotated Bitmap file. - srcPath: Bitmap image file path - targetPth: Bitmap image output file path
ResizeImageH(Bitmap, Height)	resize Bitmap image vertically. - Bitmap: Bitmap image - Height: height to resize
ResizeImageW(Bitmap, Width)	resize Bitmap image horizontally. - Bitmap: Bitmap image - Width: Width to resize
ImageBase64(Bitmap)	convert Bitmap to text. - Bitmap: Bitmap image

Web Controls

- SkyCore Framework includes built-in web controls that are pre-designed to make a developer display Html UI easily.
- Using SetStyle or SetAttribute method, a developer is able to modify html design.
- This server-side controls create HtmlText and send it to a client(web browser) through ApiResponse.
- Web controls are usually used to create html contents when initial page request and Ajax Request.

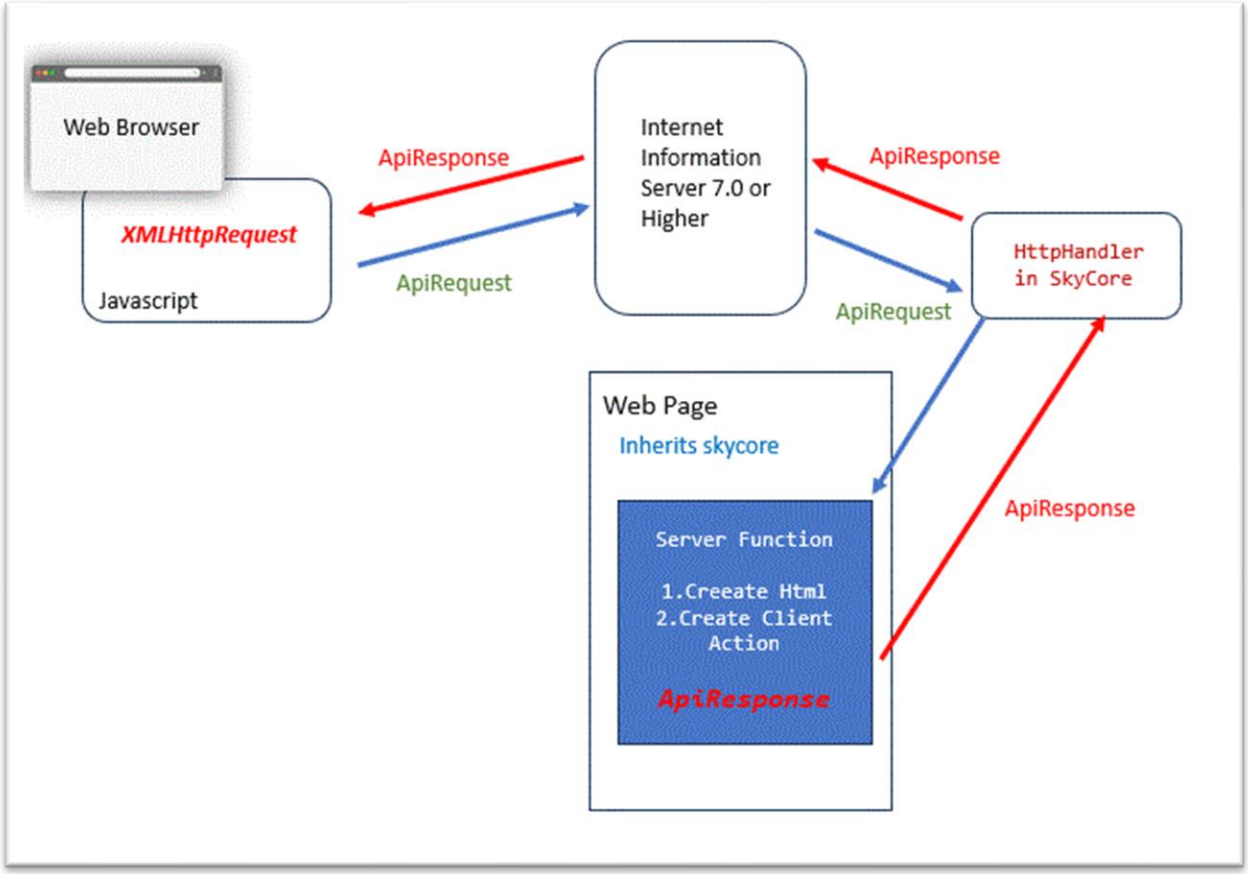
In OnInitialized Event

- During a webpage events, a developer can create html contents and finally it goes to *HtmlContext* property.
- In OnInitialized events, a developer uses webcontrols to create a initial web page.
- HttpHandler returns *HtmlContext* to a client(web browser).
- *HtmlContext* is going to be Html Body tag contents.



In AJAX request

- After initial page request, a developer can implement javascript functions to communicate with server-side function without reloading(AJAX).
- During this process, a developer can update any of html elements without refreshing page.
- Server-side functions getting a HttpRequest, it creates html contents and ApiResponse sending Html contents with *Action* to a client(web browser).
- As soon as a client gets ApiResponse, *XmlHttpRequest* displays html contents based on specific actions which is sent from server.



WebControl - ArrowIcon

- An Arrow Control is used to display arrow image icons for web page.
- A developer is able to create an event through setting Icon Attribute.

IconType Enum

Member	Value	Description
up	1	Upward
down	2	Downward
left	3	Leftward
right	4	Rightward
upleft	5	Upleft
upright	6	Upright
downleft	7	Downleft
downright	8	Downright

Code Example

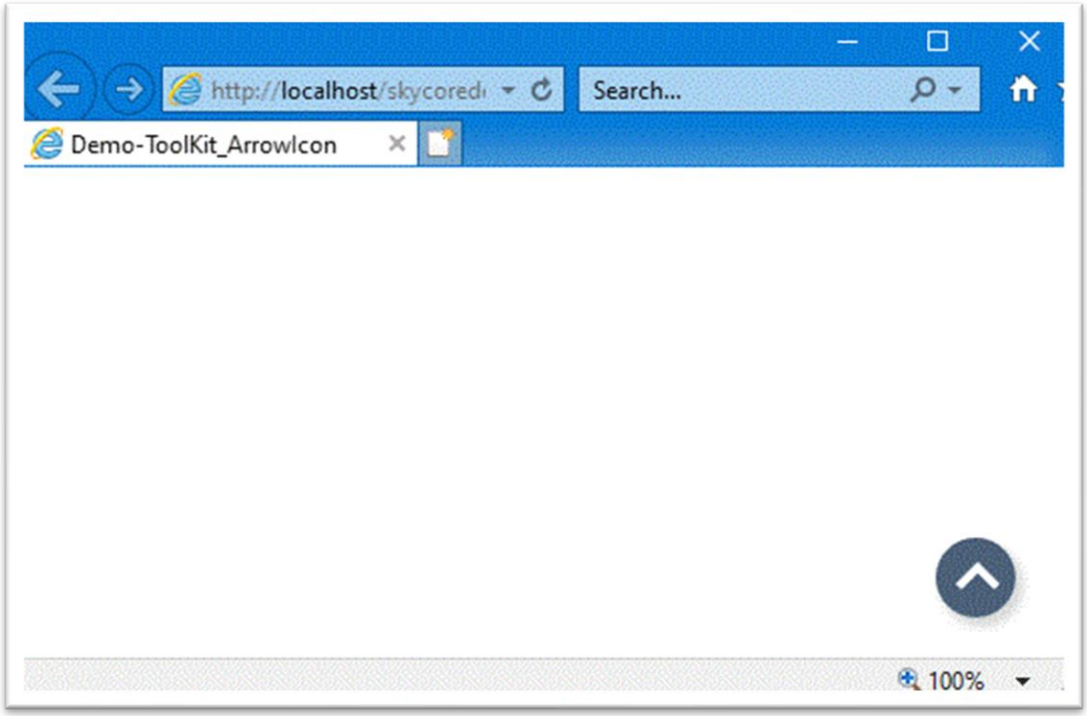
Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ArrowIcon
    Inherits WebPage
    Public Overrides Sub OnInitialized()
        Dim _ArrowIcon As New ToolKit.ArrowIcon
        _ArrowIcon.SetAttribute("onclick", "alert('up')")
        _ArrowIcon.SetStyle("bottom", "20px")
        _ArrowIcon.Style = ToolKit.ArrowIcon.IconType.up
        HtmlDoc.HtmlBodyText = _ArrowIcon.HtmlText
    End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;
public class ToolKit_ArrowIcon : WebPage
{
    public ToolKit_ArrowIcon()
    {
    }
    public override void OnInitialized()
    {
        ToolKit.ArrowIcon _ArrowIcon = new ToolKit.ArrowIcon();
        _ArrowIcon.SetAttribute("onclick", "alert('up')");
        _ArrowIcon.SetStyle("bottom", "20px");
        _ArrowIcon.Style = ToolKit.ArrowIcon.IconType.up;
        HtmlDoc.HtmlBodyText = _ArrowIcon.HtmlText();
    }
}
```

Output


http://localhost/skycored
Search...
Demo-ToolKit_ArrowIcon
100%

WebControl - Button

- The Button Control is the most basic way to get user input on web page.
- By clicking a button, the user requests that a specific action be taken in the program.
- A developer is able to create an event through setting Button Attribute.

Code Example

Fig1. Visual Basic

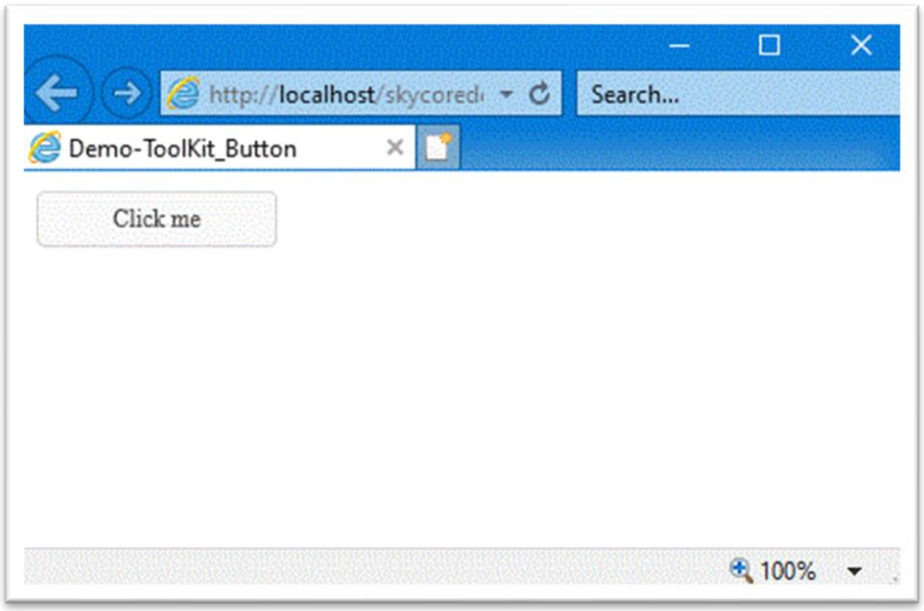
```
Imports Microsoft.VisualBasic
Imports skycore
Public Class ToolKit_Button
   Inherits WebPage
   Public Overrides Sub OnInitialized()
      Dim _Button As New ToolKit.Button
      _Button.SetAttribute("value", "Click me")
      _Button.SetAttribute("onclick", "alert('me')")
      HtmlDoc.HtmlBodyText = _Button.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;
public class ToolKit_Button : WebPage
{
            public ToolKit_Button()
            {
            }
   public override void OnInitialized()
   {
      ToolKit.Button _Button = new ToolKit.Button();
```

```
    _Button.SetAttribute("value", "Click me");
    _Button.SetAttribute("onclick", "alert('me')");
    HtmlDoc.HtmlBodyText = _Button.HtmlText();
  }
}
```

Output



WebControl - Submit

- The Submit Control submits all form values to a form-handler on web page.
- The form-handler is typically a server page with a script for processing the input data.

Code Example

Fig1. Visual Basic

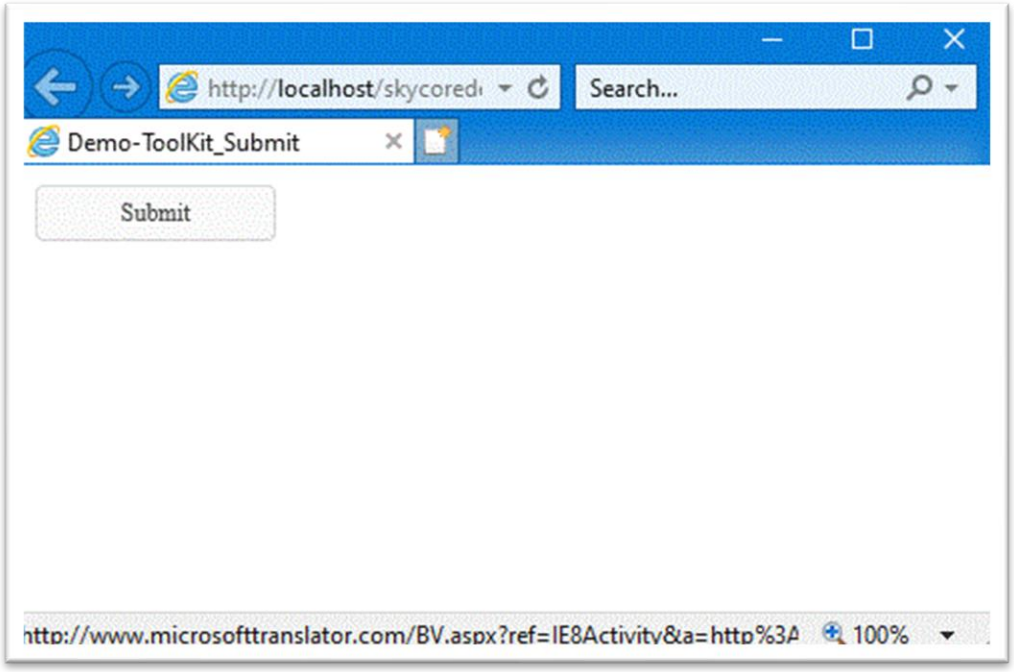
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_Submit
   Inherits WebPage
   Public Overrides Sub OnInitialized()
      Dim _Submit As New ToolKit.Submit
      _Submit.SetAttribute("value", "Submit")
      HtmlDoc.HtmlBodyText = _Submit.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;
public class ToolKit_Submit : WebPage
{
            public ToolKit_Submit()
            {
            }
   public override void OnInitialized()
   {
      ToolKit.Submit _Submit = new ToolKit.Submit();
      _Submit.SetAttribute("value", "Submit");
```

```
    HtmlDoc.HtmlBodyText = _Submit.HtmlText();
  }
}
```

Output



WebControl - Reset

- The Submit Control resets all form values to its initial values on web page.

Code Example

Fig1. Visual Basic

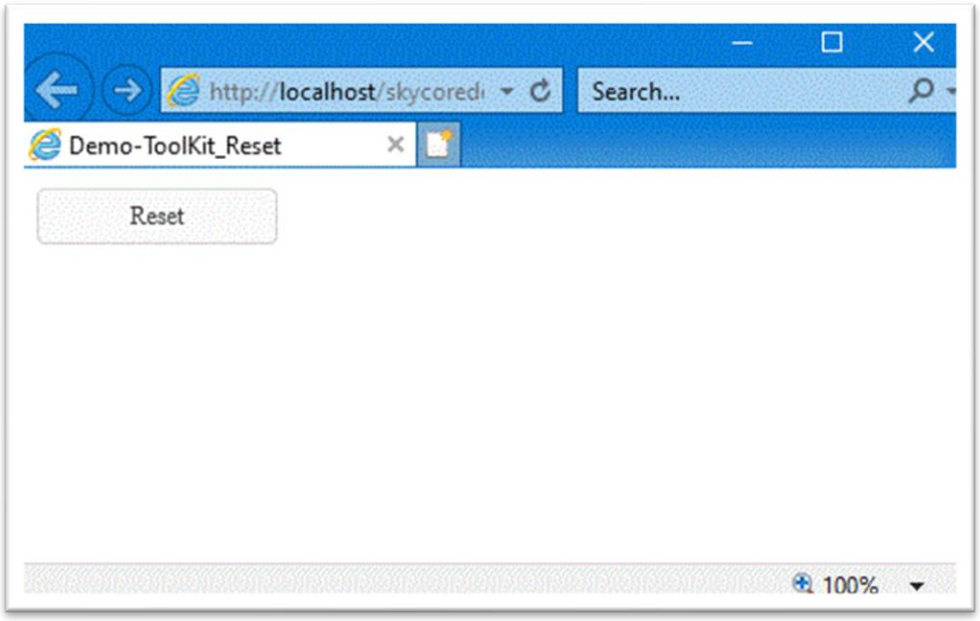
```
Imports Microsoft.VisualBasic
Imports skycore
Public Class ToolKit_Reset
   Inherits WebPage
   Public Overrides Sub OnInitialized()
      Dim _Reset As New ToolKit.Reset
      _Reset.SetAttribute("value", "Reset")
      HtmlDoc.HtmlBodyText = _Reset.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;
public class ToolKit_Reset : WebPage
{
            public ToolKit_Reset()
            {
            }
   public override void OnInitialized()
   {
      ToolKit.Reset _Reset = new ToolKit.Reset();
      _Reset.SetAttribute("value", "Reset");

      HtmlDoc.HtmlBodyText = _Reset.HtmlText();
```

```
    }
}
```

Output


http://localhost/skycored
Search...
Demo-ToolKit_Reset
Reset
100%

WebControl - ContentsBox

- The ContentsBox Control contains html elements inside a section on web page.

Code Example

- The ContentsBox Control contains a TextSearch filter.

Fig1. Visual Basic

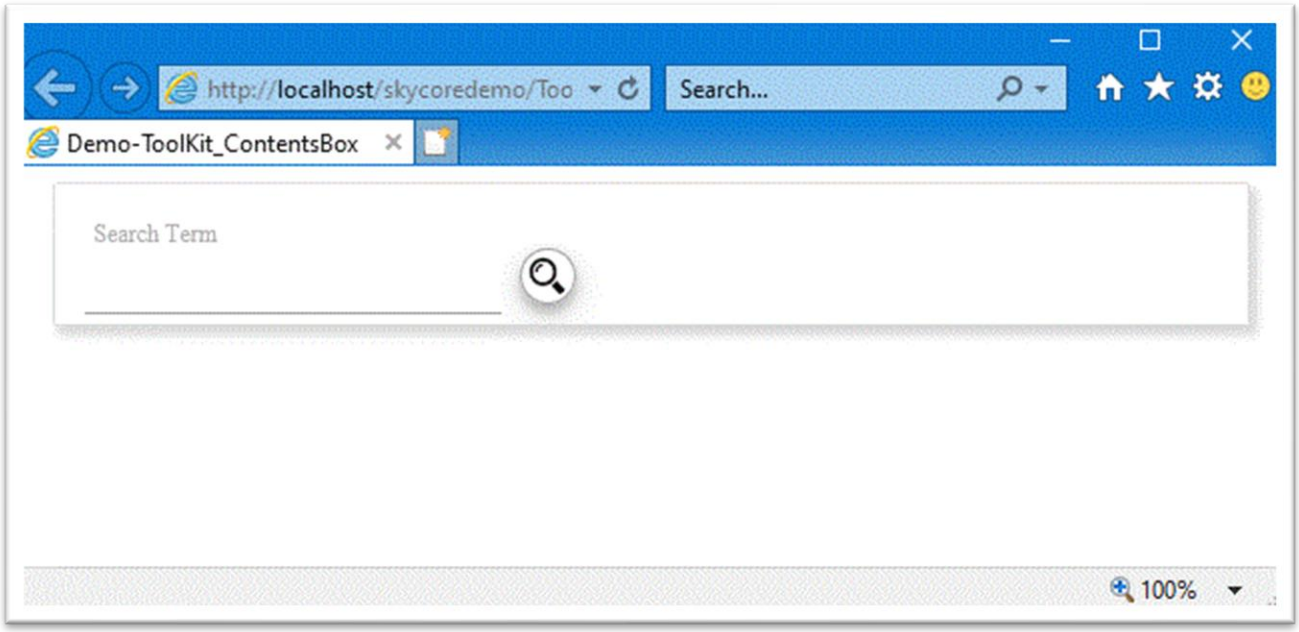
```
Imports Microsoft.VisualBasic
Imports skycore
Public Class ToolKit_ContentsBox
   Inherits WebPage
   Public Overrides Sub OnInitialized()
      Dim filter As New ToolKit.TextSearch
      filter.Label.InnerText = "Search Term"
      Dim ContentsBox As New ToolKit.ContentsBox
      ContentsBox.Border = True
      ContentsBox.AddContents(filter.HtmlText)
      HtmlDoc.HtmlBodyText = ContentsBox.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;
public class ToolKit_ContentsBox : WebPage
{
        public ToolKit_ContentsBox()
        {
        }
   public override void OnInitialized()
```

```
  {
    ToolKit.TextSearch filter = new ToolKit.TextSearch();
    filter.Label.InnerText = "Search Term";

    ToolKit.ContentsBox  ContentsBox = new ToolKit.ContentsBox();
    ContentsBox.Border = true;
    ContentsBox.AddContents(filter.HtmlText());

    HtmlDoc.HtmlBodyText = ContentsBox.HtmlText();
  }
}
```

Output



WebControl - Custom

- The Custom Control contains html elements on web page.

Code Example

Fig1. Visual Basic

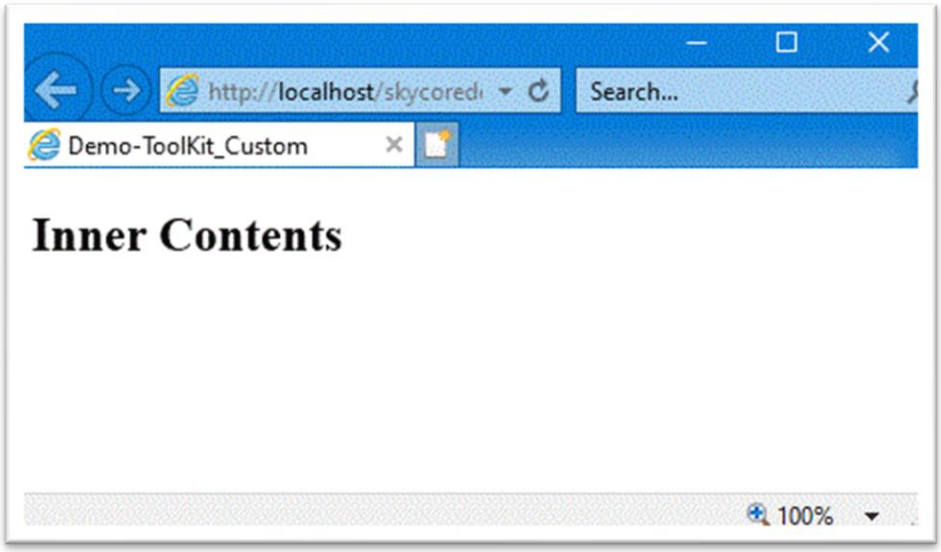
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_Custom
   Inherits WebPage

   Public Overrides Sub OnInitialized()

      Dim _Custom As New ToolKit.Custom
      _Custom.InnerContents = "<h2>Inner Contents</h2>"

      HtmlDoc.HtmlBodyText = _Custom.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_Custom : WebPage
{
        public ToolKit_Custom()
```

```
        {
        }
  public override void OnInitialized()
  {
    ToolKit.Custom _Custom = new ToolKit.Custom();
    _Custom.InnerContents = @"<h2>Inner Contents</h2>";

    HtmlDoc.HtmlBodyText = _Custom.HtmlText();
  }
}
```

Output



WebControl - DataList

- The DataList Control specifies a list of pre-defined options for an input element.
- The DataList Control provide an "autocomplete" feature for input elements.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_DataList
    Inherits WebPage

    Public Overrides Sub OnInitialized()
        Dim filter As New ToolKit.TextSearch
        filter.Label.InnerText = "Search Term"
        filter.Text.SetAttribute(HtmlAttributes.list, "StudentList")

        Dim _NameValues As New List(Of NameValue)
        _NameValues.Add(New NameValue With {.name = "Alex", .value = "90"})
        _NameValues.Add(New NameValue With {.name = "David", .value = "87"})
        _NameValues.Add(New NameValue With {.name = "Julie", .value = "91"})
        _NameValues.Add(New NameValue With {.name = "Rose", .value = "75"})

        Dim _datalist As New ToolKit.DataList
        _datalist.SetAttribute(HtmlAttributes.id, "StudentList")
        _datalist.SetOption(_NameValues)

        Dim _wrap As New ToolKit.Wrap
        _wrap.AddItem(filter.HtmlText)
        _wrap.AddItem(_datalist.HtmlText)
```

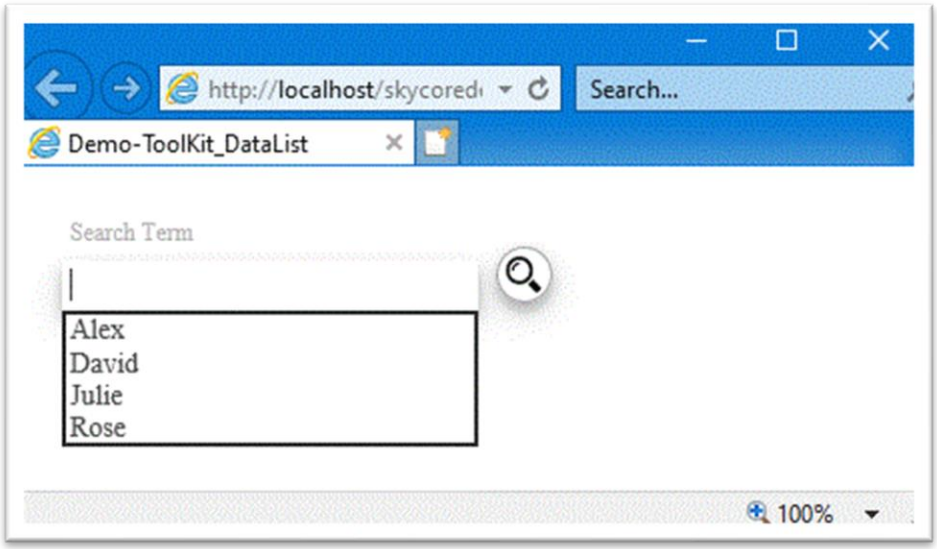
```
    HtmlDoc.HtmlBodyText = _wrap.HtmlText
  End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_DataList : WebPage
{
        public ToolKit_DataList()
        {
        }
  public override void OnInitialized()
  {
    ToolKit.TextSearch filter = new ToolKit.TextSearch();
    filter.Label.InnerText = "Search Term";
    filter.Text.SetAttribute(HtmlAttributes.list, "StudentList");

    List<NameValue> _NameValues = new List<NameValue>();
    _NameValues.Add(new NameValue() {name = "Alex", value = "90"});
    _NameValues.Add(new NameValue() {name = "David", value = "87"});
    _NameValues.Add(new NameValue() {name = "Julie", value = "91"});
    _NameValues.Add(new NameValue() {name = "Rose", value = "75"});

    ToolKit.DataList _datalist = new ToolKit.DataList();
     _datalist.SetAttribute(HtmlAttributes.id, "StudentList");
     _datalist.SetOption(_NameValues);

    ToolKit.Wrap _wrap = new ToolKit.Wrap();
```

```
    _wrap.AddItem(filter.HtmlText());
    _wrap.AddItem(_datalist.HtmlText());

    HtmlDoc.HtmlBodyText = _wrap.HtmlText();
  }
}
```

Output



WebControl - ConfirmDialog

- The ConfirmDialog Control makes it easy to create popup dialogs and modals on a web page.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ConfirmDialog
    Inherits WebPage

    Public Overrides Sub OnInitialized()
        Dim htmlText As String = "<div><input type=""button"" value=""Confirm"" onclick=""PopupConfirmDialog()""
/></div>"
        htmlText += "<script>" +
                    " function PopupConfirmDialog() {" +
                    " $ApiRequest($fn(arguments)); " +
                    " } " +
                    "</script>"

        HtmlDoc.HtmlBodyText = htmlText
    End Sub

    Public Function PopupConfirmDialog() As ApiResponse

        Dim confirm As New ToolKit.ConfirmDialog("Confirm Title", "Would you like to add a new data?", "alert('confirm')")

        Dim _ApiResponse As New ApiResponse
        _ApiResponse.PopUpWindow(confirm.HtmlText)
        Return _ApiResponse
```

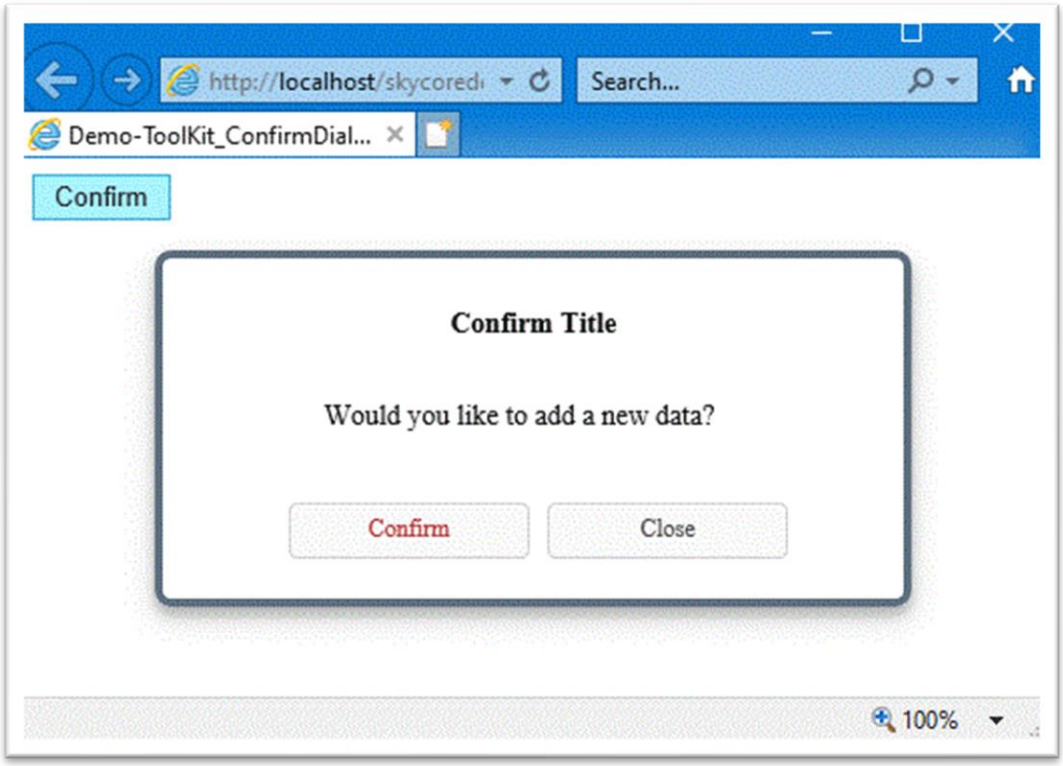
```
    End Function


End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_ConfirmDialog : WebPage
{
        public ToolKit_ConfirmDialog()
        {
        }
    public override void OnInitialized()
    {
        string htmlText = @"<div><input type=""button"" value=""Confirm"" onclick=""PopupConfirmDialog()"" /></div>";
        htmlText += @"<script>" +
                " function PopupConfirmDialog() {" +
                " $ApiRequest($fn(arguments)); " +
                " } " +
                "</script>";

        HtmlDoc.HtmlBodyText = htmlText;
    }
    public ApiResponse PopupConfirmDialog() {
        ToolKit.ConfirmDialog confirm = new ToolKit.ConfirmDialog("Confirm Title", "Would you like to add a new data?",
"alert('confirm')");
```

```
    ApiResponse _ApiResponse = new ApiResponse();
    _ApiResponse.PopUpWindow(confirm.HtmlText());
    return _ApiResponse;
  }
}
```

Output



WebControl - MessageDialog

- The MessageDialog Control makes it easy to create a popup message on a web page.

Code Example

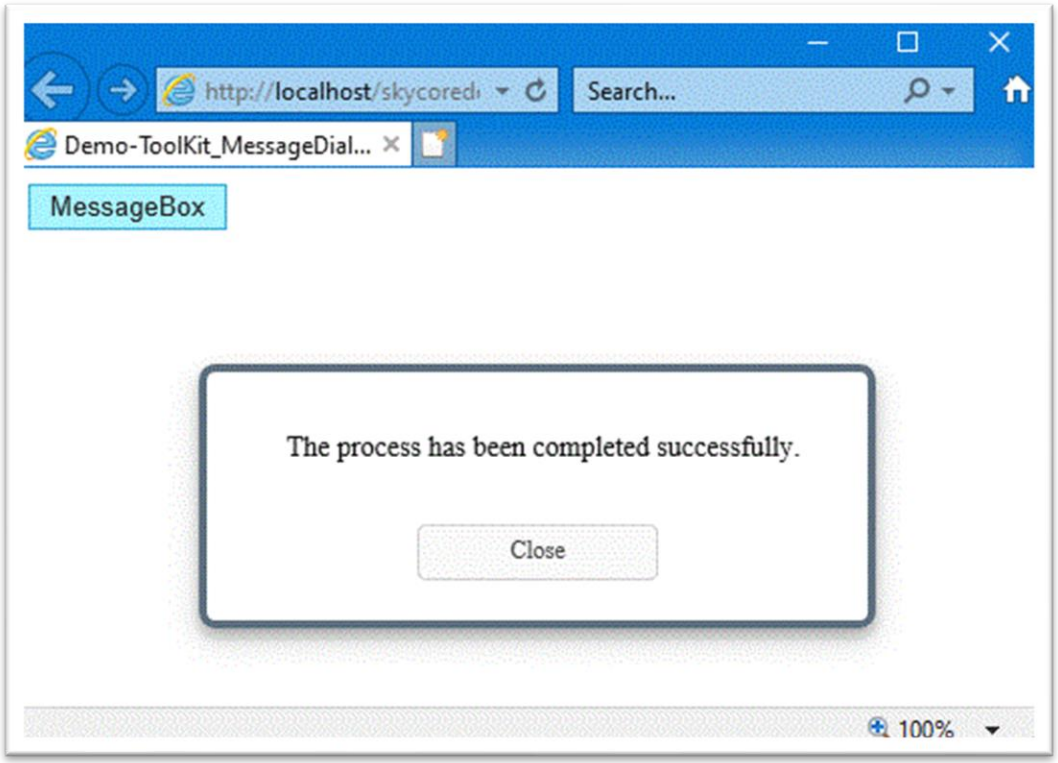
Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_MessageDialog
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim htmlText As String = "<div><input type=""button"" value=""MessageBox"" onclick=""PopupMessageDialog()"" /></div>"
      htmlText += "<script>" +
            " function PopupMessageDialog() {" +
            " $ApiRequest($fn(arguments)); " +
            " } " +
            "</script>"

      HtmlDoc.HtmlBodyText = htmlText
   End Sub

   Public Function PopupMessageDialog() As ApiResponse
      Dim message As New ToolKit.MessageDialog("The process has been completed successfully.")

      Dim _ApiResponse As New ApiResponse
      _ApiResponse.PopUpWindow(message.HtmlText)
      Return _ApiResponse
   End Function
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_MessageDialog : WebPage
{
        public ToolKit_MessageDialog()
        {
        }
   public override void OnInitialized()
   {
      string htmlText = @"<div><input type=""button"" value=""MessageBox"" onclick=""PopupMessageDialog()""
/></div>";
      htmlText += @"<script>" +
                " function PopupMessageDialog() {" +
                " $ApiRequest($fn(arguments)); " +
                " } " +
                "</script>";

      HtmlDoc.HtmlBodyText = htmlText;
   }
   public ApiResponse PopupMessageDialog()
   {
      ToolKit.MessageDialog message = new ToolKit.MessageDialog("The process has been completed successfully.");

      ApiResponse _ApiResponse = new ApiResponse();
      _ApiResponse.PopUpWindow(message.HtmlText());
      return _ApiResponse;
   }
}
```

Output


http://localhost/skycored
Search...
Demo-ToolKit_MessageDial...
MessageBox
The process has been completed successfully.
Close
100%

WebControl - FileUpload

- The FileUpload Control allows to upload files to server.

Code Example

Fig1. Visual Basic

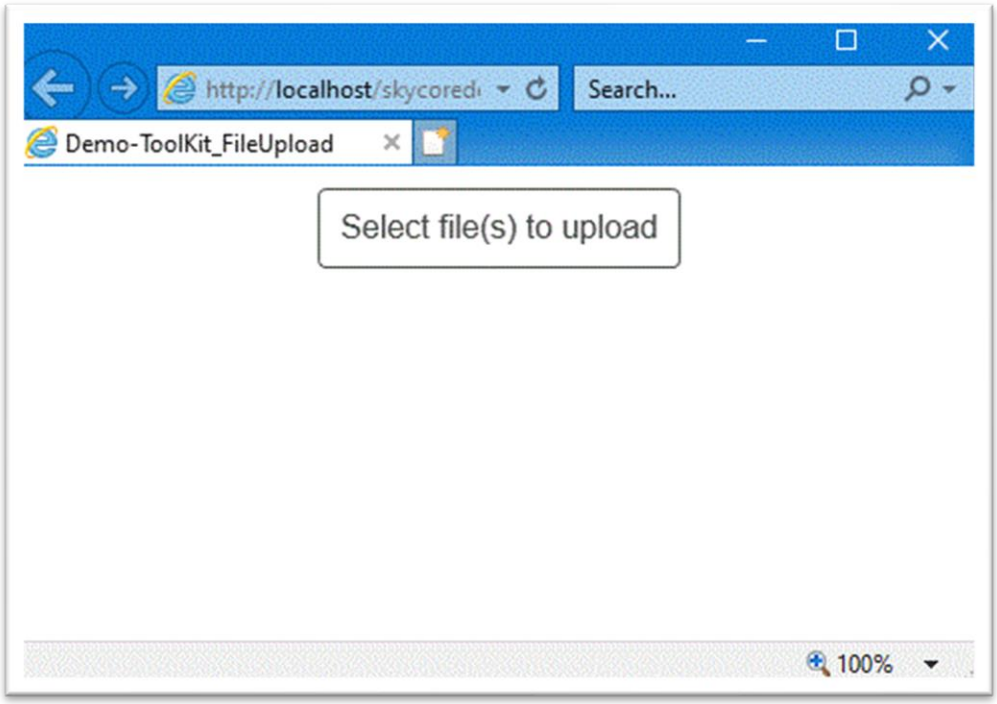
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_FileUpload
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _file As New ToolKit.FileUpload
      _file.FileText.SetAttribute(HtmlAttributes.multiple, String.Empty)

      HtmlDoc.HtmlBodyText = _file.HtmlText

   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_FileUpload : WebPage
{
        public ToolKit_FileUpload()
```

```
        {

        }
  public override void OnInitialized()
  {
    ToolKit.FileUpload _file = new ToolKit.FileUpload();
    _file.FileText.SetAttribute(HtmlAttributes.multiple, String.Empty);


    HtmlDoc.HtmlBodyText = _file.HtmlText();
  }
}
```

Output



WebControl - DateSearch Filter

- The DateSearch Control allows to pick a date with search image button.

Code Example

Fig1. Visual Basic

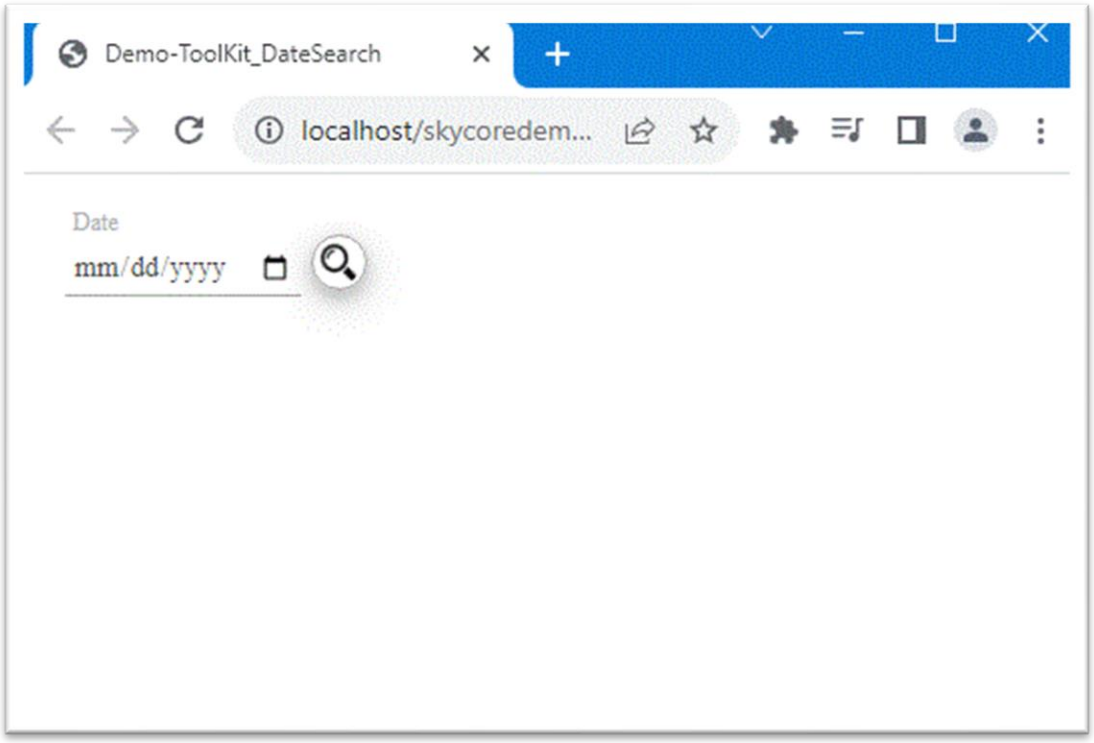
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_DateSearch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _DateSearch As New ToolKit.DateSearch
      _DateSearch.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')")

      HtmlDoc.HtmlBodyText = _DateSearch.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_DateSearch : WebPage
{
        public ToolKit_DateSearch()
        {
        }
   public override void OnInitialized()
```

```
  {
    ToolKit.DateSearch _DateSearch = new ToolKit.DateSearch();
    _DateSearch.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')");

    HtmlDoc.HtmlBodyText = _DateSearch.HtmlText();
  }
}
```

Output



WebControl - DateDateSearch Filter

- The DateDateSearch Control allows to set from-date-to-date with search image button.

Code Example

Fig1. Visual Basic

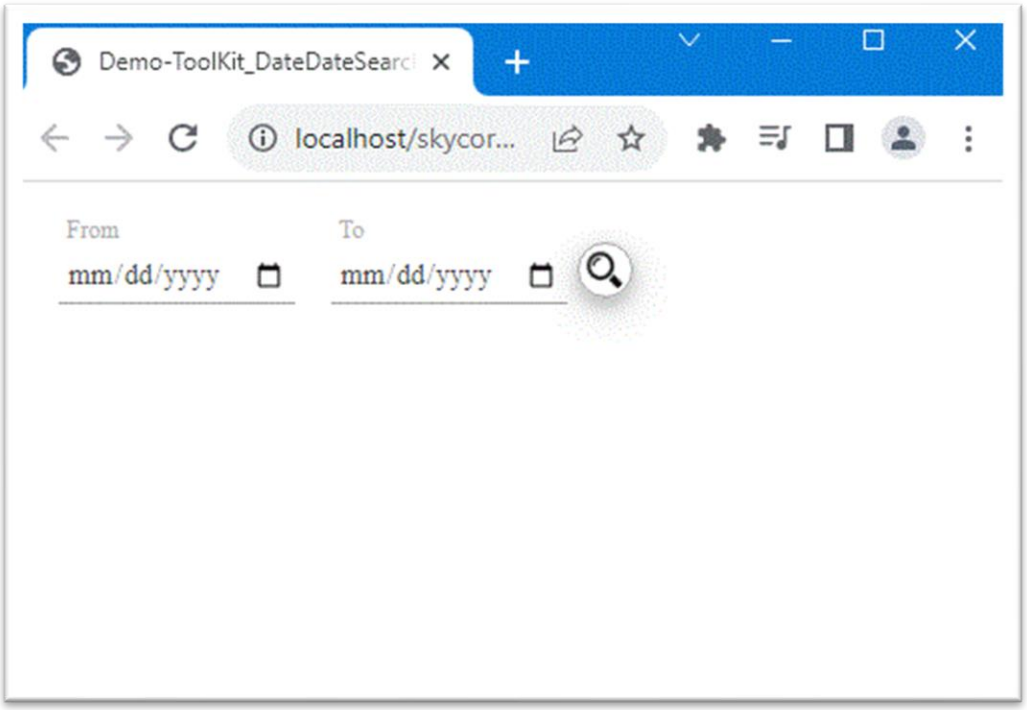
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_DateDateSearch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _DateSearch As New ToolKit.DateDateSearch
      _DateSearch.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')")

      HtmlDoc.HtmlBodyText = _DateSearch.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_DateDateSearch : WebPage
{
        public ToolKit_DateDateSearch()
```

```
        {
        }
  public override void OnInitialized()
  {
    ToolKit.DateDateSearch _DateSearch = new ToolKit.DateDateSearch();
    _DateSearch.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')");

    HtmlDoc.HtmlBodyText = _DateSearch.HtmlText();
  }
}
```

Output



WebControl - DblDropDownSearch Filter

- The DblDropDownSearch Control allows to set 2 dropdown html elements with search image button.

Code Example

Fig1. Visual Basic

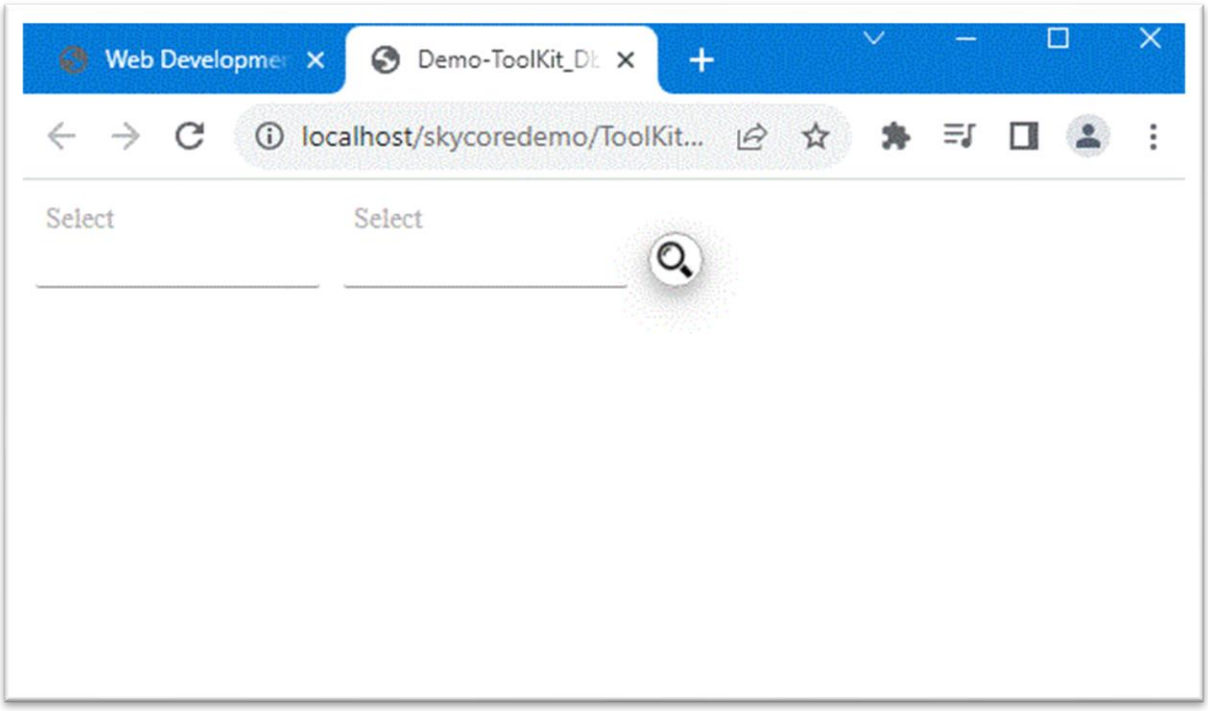
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_DblDropDownSearch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _DblDrop As New ToolKit.DblDropDownSearch
      _DblDrop.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')")

      HtmlDoc.HtmlBodyText = _DblDrop.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_DblDropDownSearch : WebPage
{
        public ToolKit_DblDropDownSearch()
        {
        }
   public override void OnInitialized()
   {
```

```
    ToolKit.DblDropDownSearch _DblDrop = new ToolKit.DblDropDownSearch();
    _DblDrop.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')");

    HtmlDoc.HtmlBodyText = _DblDrop.HtmlText();
  }
}
```

Output



WebControl - DropDownSearch Filter

- The DblDropDownSearch Control creates a dropdown html element with search image button.

Code Example

Fig1. Visual Basic

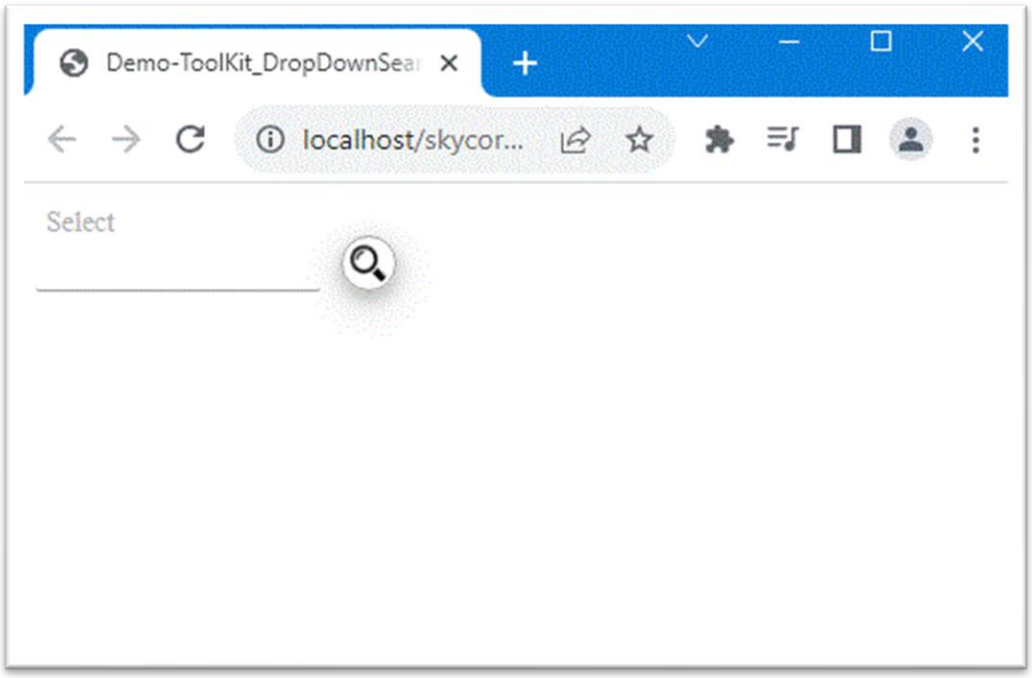
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_DropDownSearch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _Drop As New ToolKit.DropDownSearch
      _Drop.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')")

      HtmlDoc.HtmlBodyText = _Drop.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_DropDownSearch : WebPage
{
        public ToolKit_DropDownSearch()
        {
        }
   public override void OnInitialized()
```

```
  {
    ToolKit.DropDownSearch _Drop = new ToolKit.DropDownSearch();
    _Drop.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')");

    HtmlDoc.HtmlBodyText = _Drop.HtmlText();
  }
}
```

Output



WebControl - MonthSearch Filter

- The MonthSearch Control creates a month input element with search image button.

Code Example

Fig1. Visual Basic

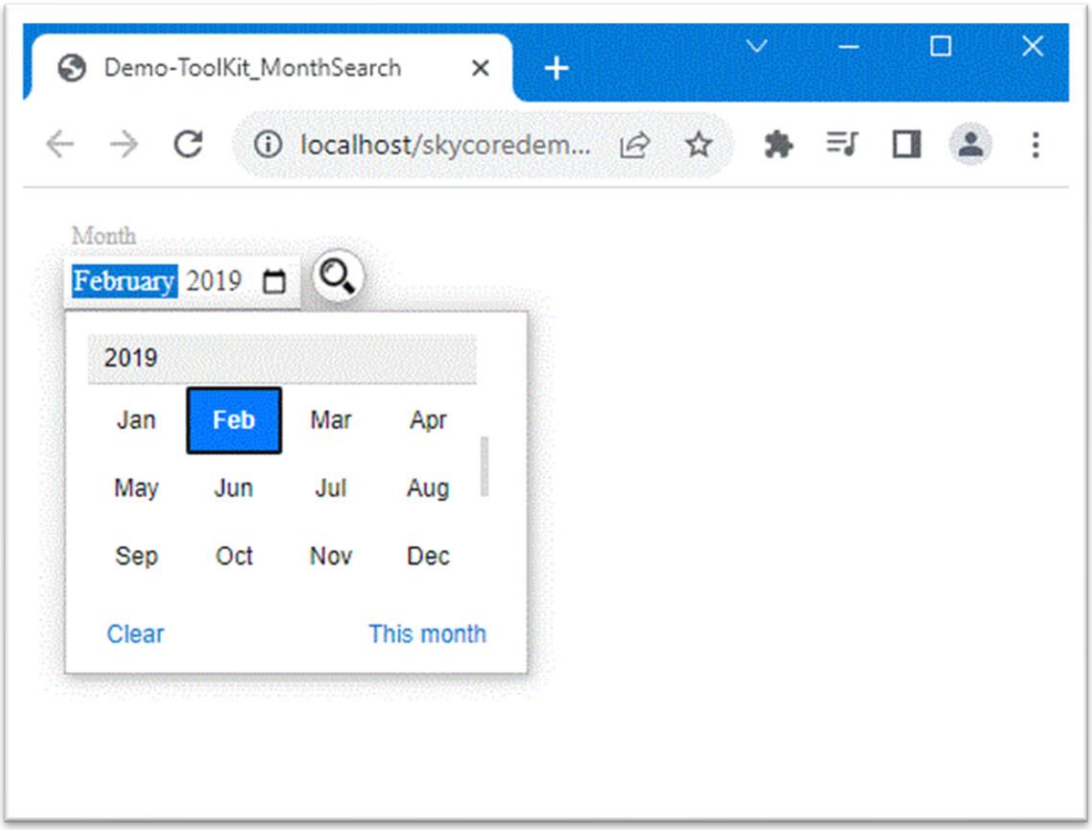
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_MonthSearch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _month As New ToolKit.MonthSearch
      _month.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')")

      HtmlDoc.HtmlBodyText = _month.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_MonthSearch : WebPage
{
        public ToolKit_MonthSearch()
        {
        }
   public override void OnInitialized()
```

```
  {
    ToolKit.MonthSearch _month = new ToolKit.MonthSearch();
    _month.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')");

    HtmlDoc.HtmlBodyText = _month.HtmlText();
  }
}
```

Output



WebControl - MonthMonthSearch Filter

- The MonthMonthSearch Control creates 2 month input elements(from-to) with search image button.

Code Example

Fig1. Visual Basic

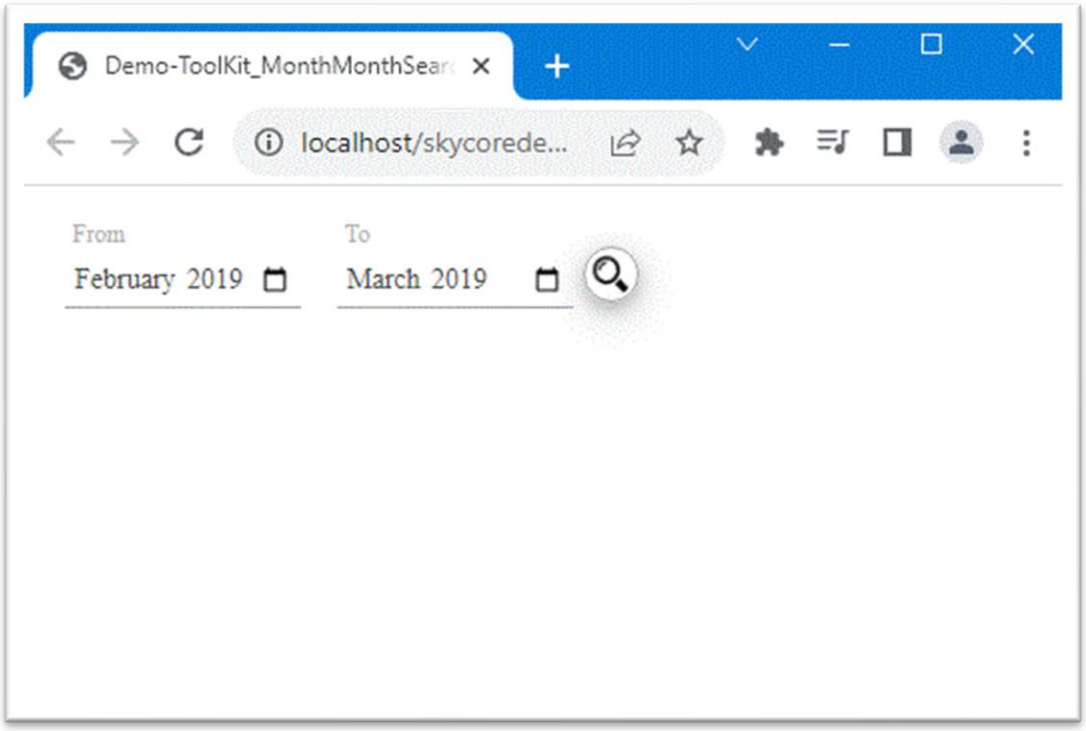
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_MonthMonthSearch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _month As New ToolKit.MonthMonthSearch
      _month.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')")

      HtmlDoc.HtmlBodyText = _month.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_MonthMonthSearch : WebPage
{
        public ToolKit_MonthMonthSearch()
        {
        }
   public override void OnInitialized()
   {
```

```
    ToolKit.MonthMonthSearch _month = new ToolKit.MonthMonthSearch();
    _month.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')");

    HtmlDoc.HtmlBodyText = _month.HtmlText();
  }
}
```

Output



WebControl - TextSearch Filter

- The TextSearch Control creates a text input element with search image button.

Code Example

Fig1. Visual Basic

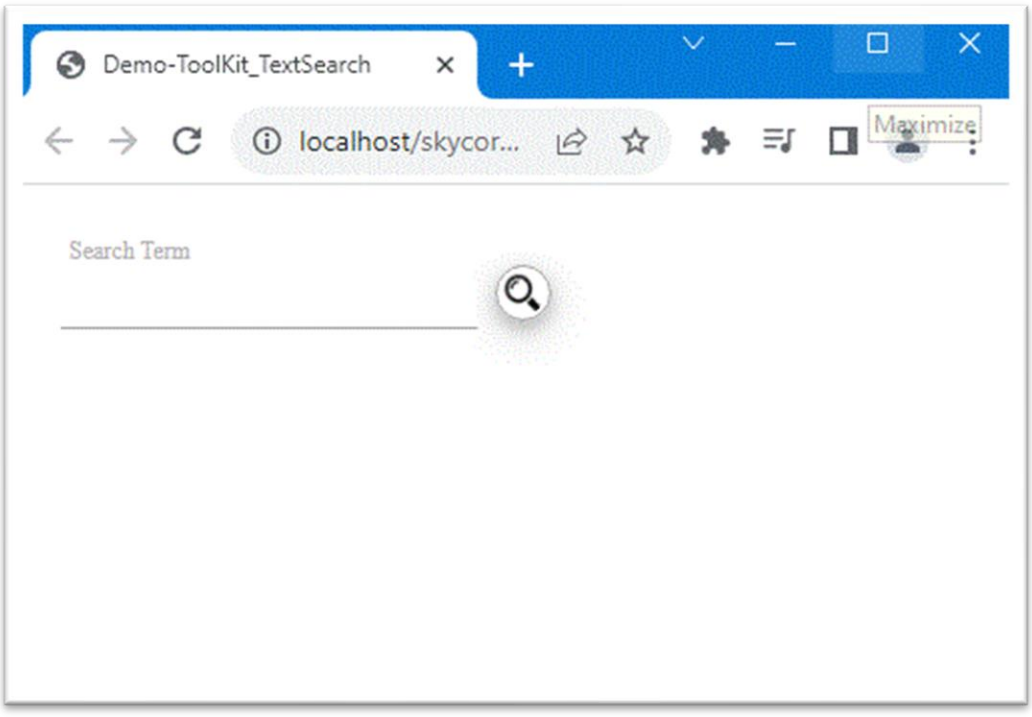
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_TextSearch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _text As New ToolKit.TextSearch
      _text.Label.InnerText = "Search Term"
      _text.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')")

      HtmlDoc.HtmlBodyText = _text.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_TextSearch : WebPage
{
        public ToolKit_TextSearch()
        {
        }
```

```
  public override void OnInitialized()
  {
    ToolKit.TextSearch _text = new ToolKit.TextSearch();
    _text.Label.InnerText = "Search Term";
    _text.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')");

    HtmlDoc.HtmlBodyText = _text.HtmlText();
  }
}
```

Output



WebControl - TextCheckSearch Filter

- The TextCheckSearch Control creates a text and check input elements with search image button.

Code Example

Fig1. Visual Basic

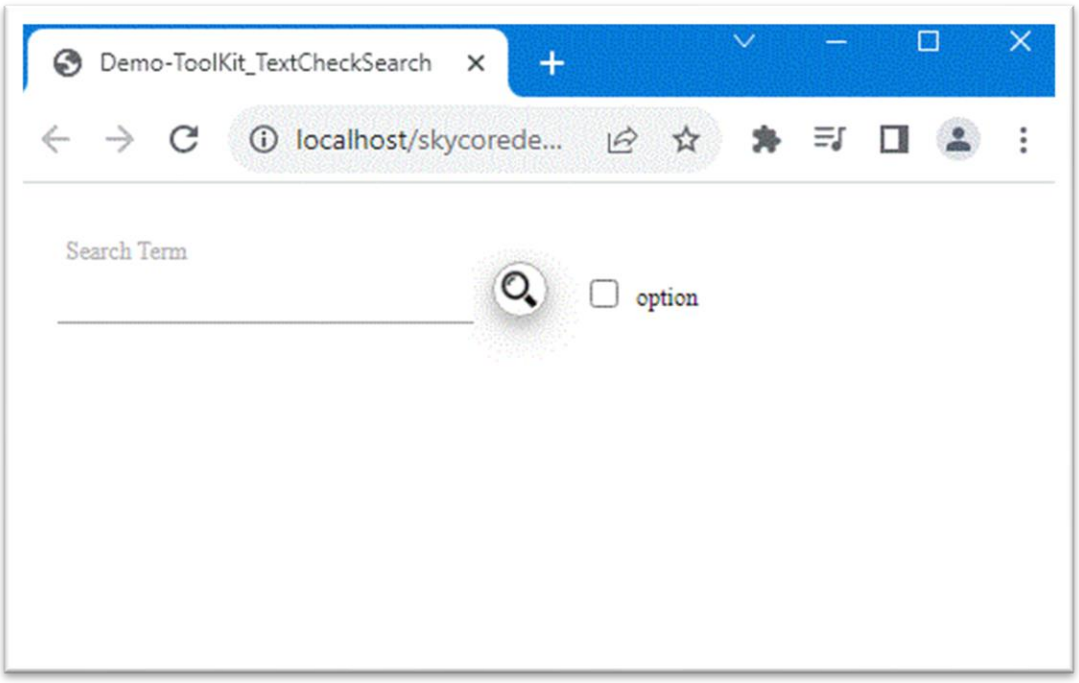
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_TextCheckSearch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _text As New ToolKit.TextCheckSearch
      _text.Label.InnerText = "Search Term"
      _text.Check.Label.InnerText = "option"
      _text.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')")

      HtmlDoc.HtmlBodyText = _text.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_TextCheckSearch : WebPage
{
        public ToolKit_TextCheckSearch()
        {
```

```
        }
  public override void OnInitialized()
  {
     ToolKit.TextCheckSearch _text = new ToolKit.TextCheckSearch();
     _text.Label.InnerText = "Search Term";
     _text.Check.Label.InnerText = "option";
     _text.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')");

     HtmlDoc.HtmlBodyText = _text.HtmlText();
  }
}
```

Output



WebControl - TextDateSearch Filter

- The TextCheckSearch Control creates a text and date input elements with search image button.

Code Example

Fig1. Visual Basic

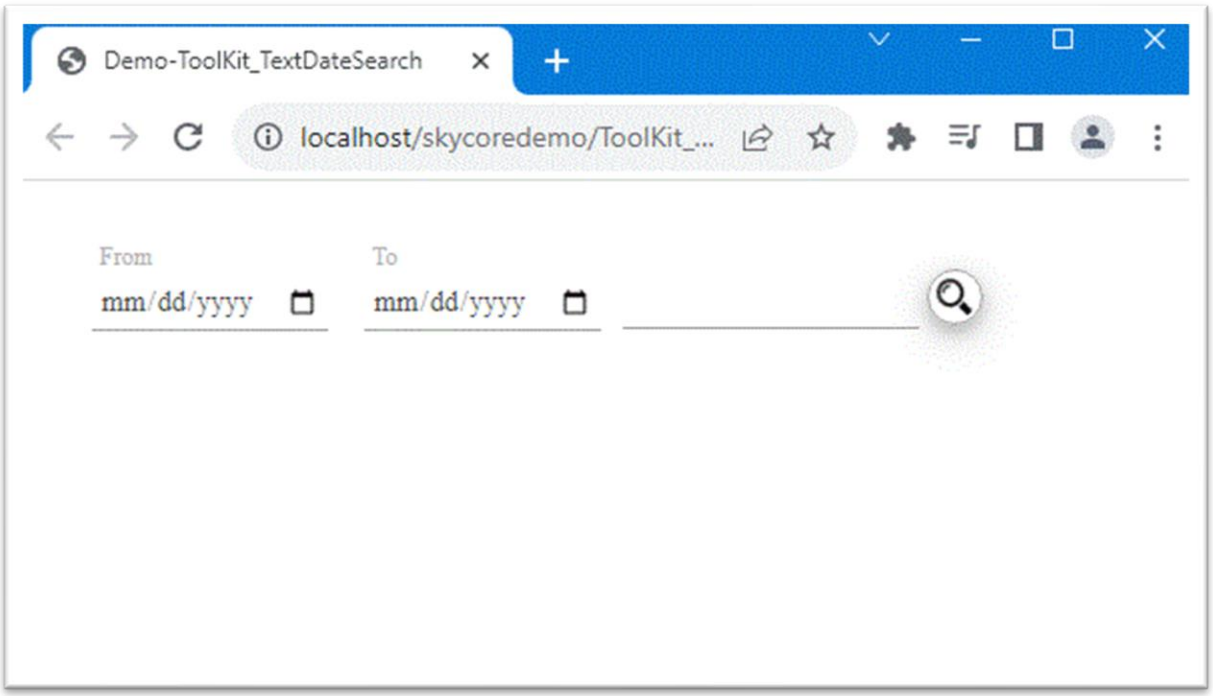
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_TextDateSearch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _text As New ToolKit.TextDateSearch
      _text.Label.InnerText = "Search Term"
      _text.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')")

      HtmlDoc.HtmlBodyText = _text.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_TextDateSearch : WebPage
{
        public ToolKit_TextDateSearch()
        {
        }
```

```
  public override void OnInitialized()
  {
    ToolKit.TextDateSearch _text = new ToolKit.TextDateSearch();
    _text.Label.InnerText = "Search Term";
    _text.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')");

    HtmlDoc.HtmlBodyText = _text.HtmlText();
  }
}
```

Output



WebControl - TextMonthSearch Filter

- The TextMonthSearch Control creates a text and month input elements with search image button.

Code Example

Fig1. Visual Basic

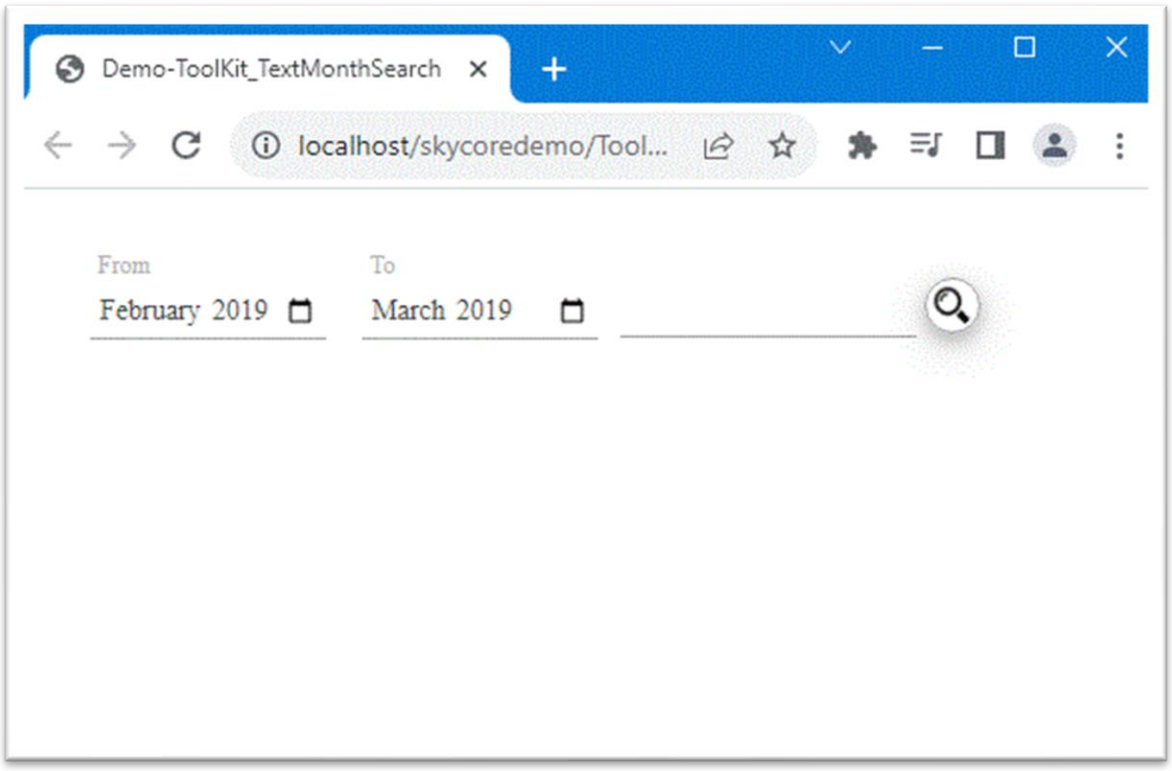
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_TextMonthSearch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _text As New ToolKit.TextMonthSearch
      _text.Label.InnerText = "Search Term"
      _text.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')")

      HtmlDoc.HtmlBodyText = _text.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_TextMonthSearch : WebPage
{
        public ToolKit_TextMonthSearch()
        {
        }
```

```
  public override void OnInitialized()
  {
    ToolKit.TextMonthSearch _text = new ToolKit.TextMonthSearch();
    _text.Label.InnerText = "Search Term";
    _text.Image.SetAttribute(HtmlEvents.onclick, "alert('Clicked')");

    HtmlDoc.HtmlBodyText = _text.HtmlText();
  }
}
```

Output



WebControl - GridX

- The GridX Control creates a data grid on webpage.
- The GridX Control uses a datatable as a datasource to create a datagrid.

Code Example

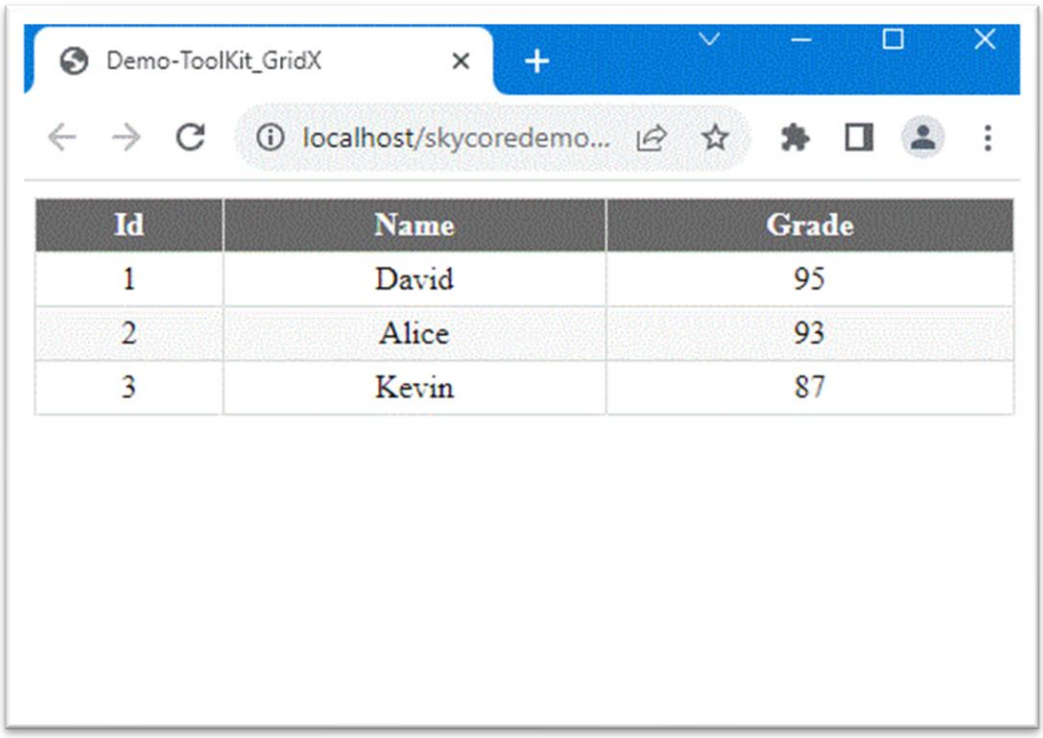
Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore
Imports System.Data
Public Class ToolKit_GridX
   Inherits WebPage
   Public Overrides Sub OnInitialized()
      Dim dt As New DataTable
      dt.Columns.Add("Id")
      dt.Columns.Add("Name")
      dt.Columns.Add("Grade")

      dt.Rows.Add({"1", "David", "95"})
      dt.Rows.Add({"2", "Alice", "93"})
      dt.Rows.Add({"3", "Kevin", "87"})

      Dim _GridX As New ToolKit.GridX
      _GridX.DataSource(dt)

      For i As Integer = 0 To _GridX.Headers.Count - 1
         _GridX.ColumnStyle(i, HtmlStyles.textAlign, "center")
      Next
      HtmlDoc.HtmlBodyText = _GridX.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;
using System.Data;

public class ToolKit_GridX : WebPage
{
        public ToolKit_GridX()
        {
        }
   public override void OnInitialized()
   {
      DataTable dt = new DataTable();
      dt.Columns.Add("Id");
      dt.Columns.Add("Name");
      dt.Columns.Add("Grade");
      dt.Rows.Add("1", "David", "95");
      dt.Rows.Add("2", "Alice", "93");
      dt.Rows.Add("3", "Kevin", "87");

      ToolKit.GridX _GridX = new ToolKit.GridX();
      _GridX.DataSource(dt);

      for (int i = 0; i < _GridX.Headers.Count; i++)
      {
         _GridX.ColumnStyle(i, HtmlStyles.textAlign, "center");
      }
      HtmlDoc.HtmlBodyText = _GridX.HtmlText();
   }
}
```

Output


Demo-ToolKit_GridX
localhost/skycoredemo...
Id
Name
Grade
1
David
95
2
Alice
93
3
Kevin
87

WebControl - Grid

- The Grid Control creates a data grid on webpage.
- The Grid Control uses a datatable as a datasource to create a datagrid.
- The Grid Control has more powerful methods than the GridX control.

Code Example

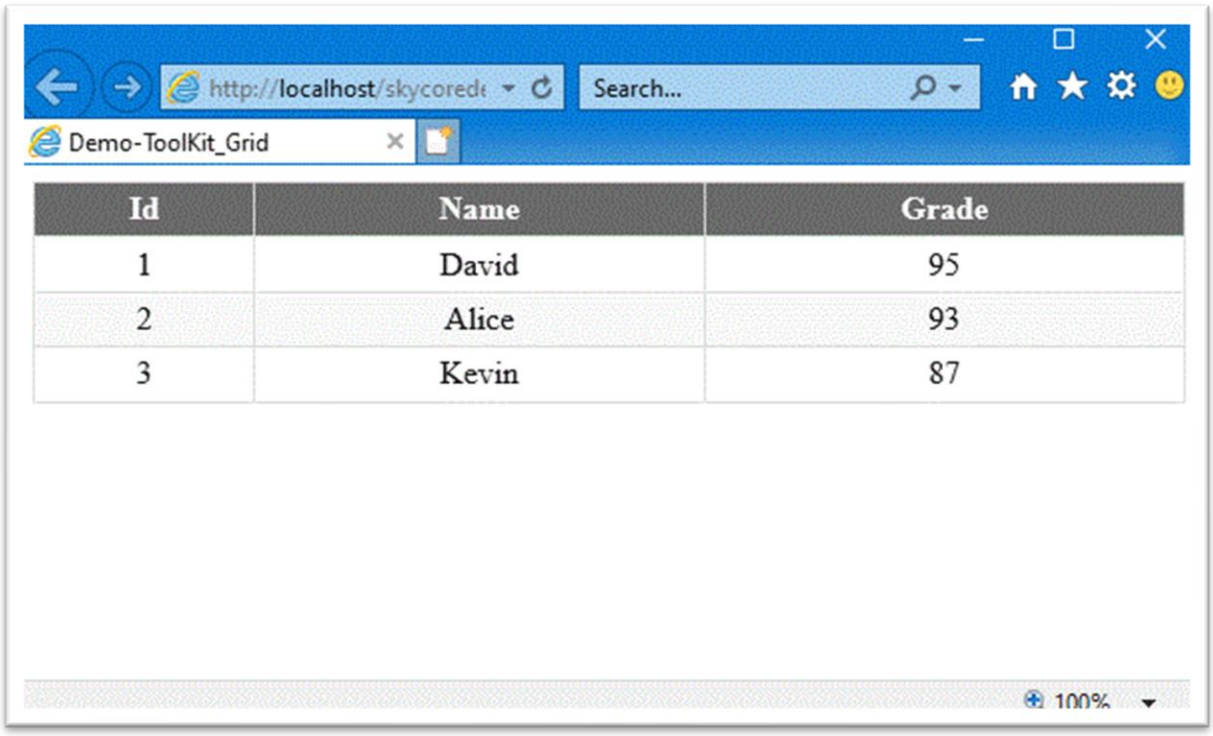
Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore
Imports System.Data

Public Class ToolKit_Grid
    Inherits WebPage
    Public Overrides Sub OnInitialized()
        Dim dt As New DataTable
        dt.Columns.Add("Id")
        dt.Columns.Add("Name")
        dt.Columns.Add("Grade")
        dt.Rows.Add({"1", "David", "95"})
        dt.Rows.Add({"2", "Alice", "93"})
        dt.Rows.Add({"3", "Kevin", "87"})

        Dim _Grid As New ToolKit.Grid(dt)
        For i As Integer = 0 To _Grid.TableColumns.Count - 1
            _Grid.TableColumns(i).SetColumnStyle(HtmlStyles.textAlign, "center")
        Next
        HtmlDoc.HtmlBodyText = _Grid.HtmlText
    End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;
using System.Data;

public class ToolKit_Grid : WebPage
{
        public ToolKit_Grid()
        {
        }
   public override void OnInitialized()
   {
      DataTable dt = new DataTable();
      dt.Columns.Add("Id");
      dt.Columns.Add("Name");
      dt.Columns.Add("Grade");
      dt.Rows.Add("1", "David", "95");
      dt.Rows.Add("2", "Alice", "93");
      dt.Rows.Add("3", "Kevin", "87");

      ToolKit.Grid _Grid = new ToolKit.Grid(dt);

      for (int i = 0; i < _Grid.TableColumns.Count; i++)
      {
         _Grid.TableColumns[i].SetColumnStyle(HtmlStyles.textAlign, "center");
      }

      HtmlDoc.HtmlBodyText = _Grid.HtmlText();
   }
}
```

Output


http://localhost/skycored
Search...
Demo-ToolKit_Grid
Id
Name
Grade
1
David
95
2
Alice
93
3
Kevin
87
100%

WebControl - ImageButton

- The ImageButton Control creates a image icon on webpage.

Code Example

Fig1. Visual Basic

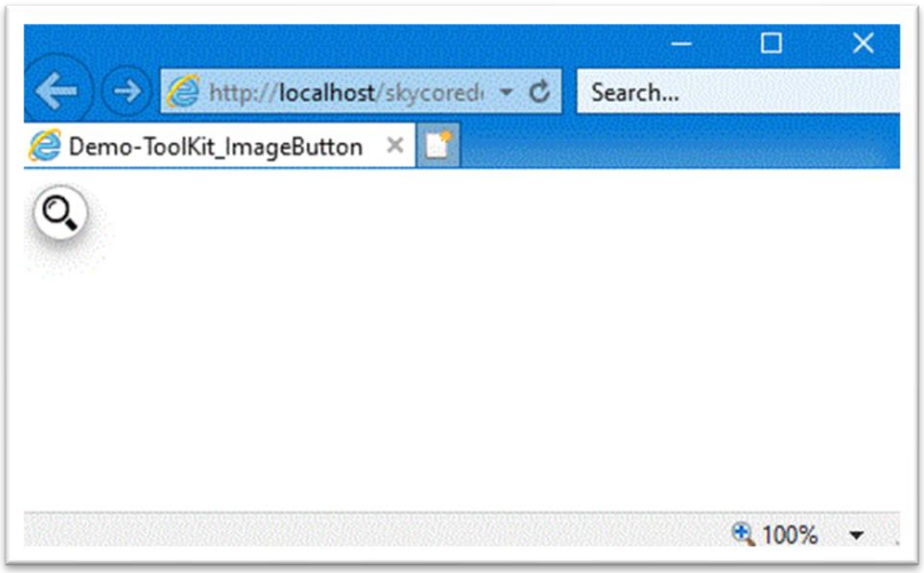
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ImageButton
    Inherits WebPage

    Public Overrides Sub OnInitialized()
        Dim _imgBtn As New ToolKit.ImageButton
        _imgBtn.SetAttribute("src", ImageAliasPath + "search.jpg")
        _imgBtn.SetAttribute("onclick", "alert('1')")

        HtmlDoc.HtmlBodyText = _imgBtn.HtmlText
    End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_ImageButton : WebPage
{
        public ToolKit_ImageButton()
```

```
        {
        }
  public override void OnInitialized()
  {
    ToolKit.ImageButton _imgBtn = new ToolKit.ImageButton();
    _imgBtn.SetAttribute("src", ImageAliasPath + "search.jpg");
    _imgBtn.SetAttribute("onclick", "alert('1')");

    HtmlDoc.HtmlBodyText = _imgBtn.HtmlText();
  }
}
```

Output



WebControl - ItemX

- The ItemX Control creates a controlable item on webpage.

Code Example

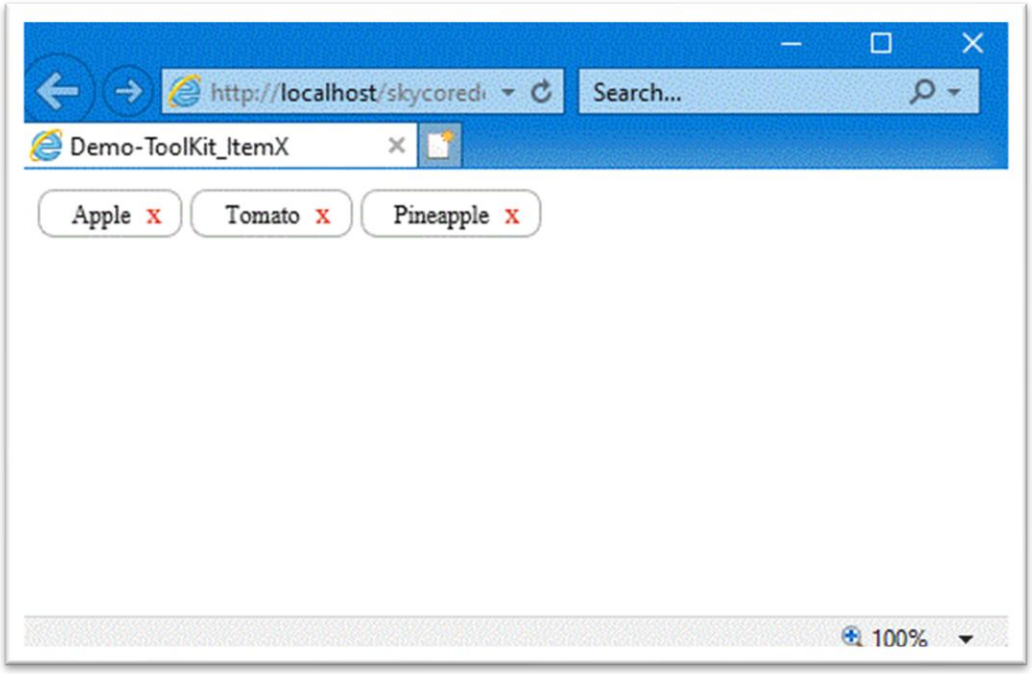
Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ItemX
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _itm As New ToolKit.ItemX
      _itm.Label.InnerText = "Apple"
      Dim _itm1 As New ToolKit.ItemX
      _itm1.Label.InnerText = "Tomato"
      Dim _itm2 As New ToolKit.ItemX
      _itm2.Label.InnerText = "Pineapple"

      Dim _wrap As New ToolKit.Wrap
      _wrap.AddItem(_itm.HtmlText)
      _wrap.AddItem(_itm1.HtmlText)
      _wrap.AddItem(_itm2.HtmlText)

      HtmlDoc.HtmlBodyText = _wrap.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_ItemX : WebPage
{
        public ToolKit_ItemX()
        {
        }
   public override void OnInitialized()
   {
      ToolKit.ItemX _itm = new ToolKit.ItemX();
      _itm.Label.InnerText = "Apple";
      ToolKit.ItemX _itm1 = new ToolKit.ItemX();
      _itm1.Label.InnerText = "Tomato";
      ToolKit.ItemX _itm2 = new ToolKit.ItemX();
      _itm2.Label.InnerText = "Pineapple";

      ToolKit.Wrap _wrap = new ToolKit.Wrap();
      _wrap.AddItem(_itm.HtmlText());
      _wrap.AddItem(_itm1.HtmlText());
      _wrap.AddItem(_itm2.HtmlText());

      HtmlDoc.HtmlBodyText = _wrap.HtmlText();
   }
}
```

Output


http://localhost/skycored
Search...
Demo-ToolKit_ItemX
Apple x
Tomato x
Pineapple x
100%

WebControl - ItemList

- The ItemList Control creates item-list with title on webpage.

Code Example

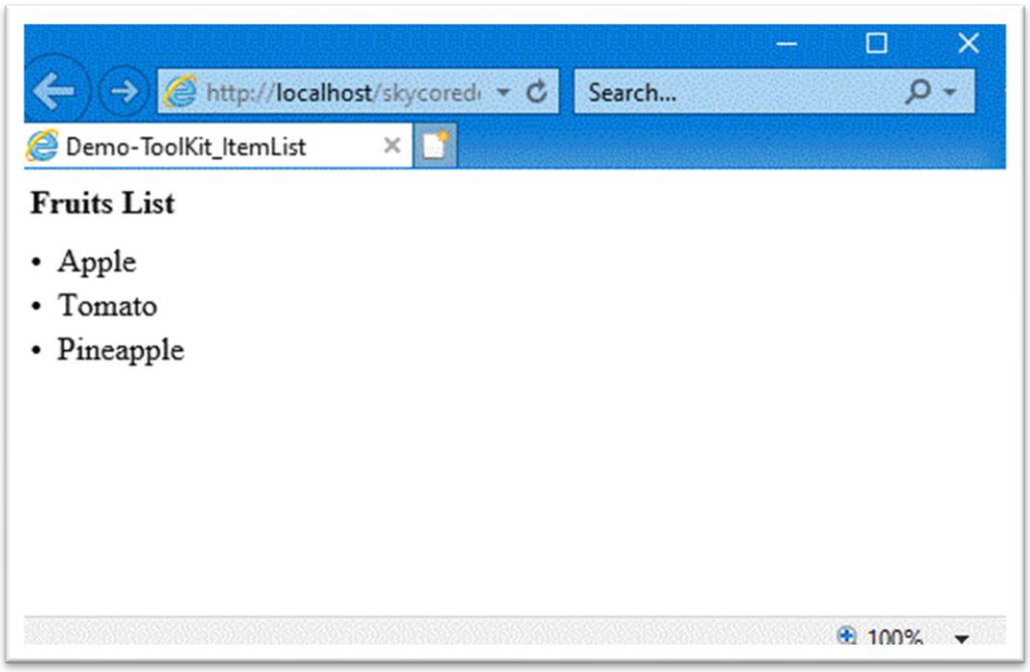
Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ItemList
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _itmlist As New ToolKit.ItemList
      _itmlist.Title.InnerText = "Fruits List"

      _itmlist.AddItem("Apple", "alert(this.innerHTML)")
      _itmlist.AddItem("Tomato", "alert(this.innerHTML)")
      _itmlist.AddItem("Pineapple", "alert(this.innerHTML)")

      For i As Integer = 0 To _itmlist.Items.Count - 1
         _itmlist.Items(i).SetStyle(HtmlStyles.fontSize, "16px")
      Next

      HtmlDoc.HtmlBodyText = _itmlist.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_ItemList : WebPage
{
        public ToolKit_ItemList()
        {
        }
   public override void OnInitialized()
   {
      ToolKit.ItemList _itmlist = new ToolKit.ItemList();
      _itmlist.Title.InnerText = "Fruits List";

      _itmlist.AddItem("Apple", "alert(this.innerHTML)");
      _itmlist.AddItem("Tomato", "alert(this.innerHTML)");
      _itmlist.AddItem("Pineapple", "alert(this.innerHTML)");

      for (int i = 0; i < _itmlist.Items.Count; i++)
      {
         _itmlist.Items[i].SetStyle(HtmlStyles.fontSize, "16px");
      }

      HtmlDoc.HtmlBodyText = _itmlist.HtmlText();
   }
}
```

Output


http://localhost/skycored
Search...
Demo-ToolKit_ItemList
Fruits List
Apple
Tomato
Pineapple
100%

WebControl - ImageIcon

- The ImageIcon Control creates a image icon on webpage.

Code Example

Fig1. Visual Basic

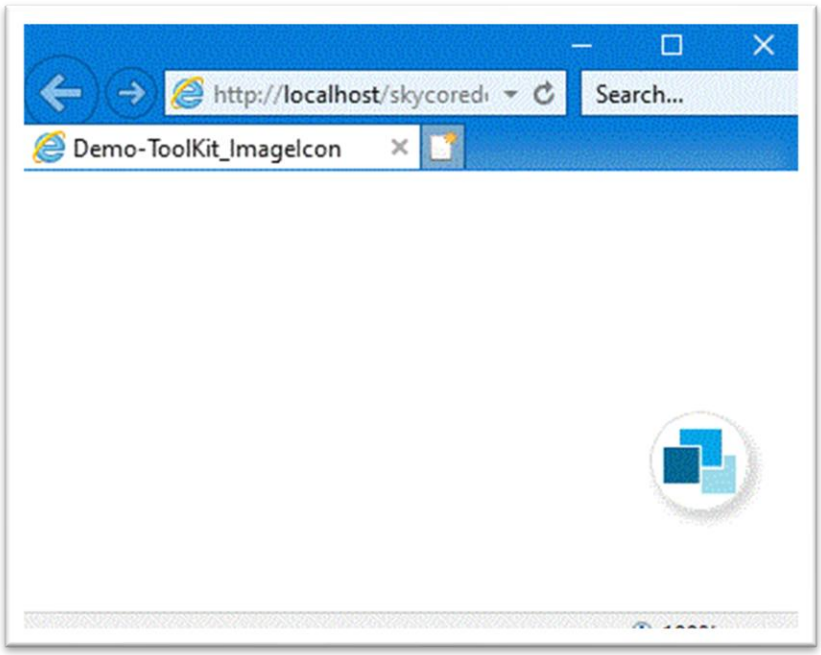
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ImageIcon
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _imgIcon As New ToolKit.ImageIcon
      _imgIcon.Icon.SetAttribute("title", "image")
      _imgIcon.Icon.SetAttribute("src", ImageAliasPath + "logo.jpg")

      HtmlDoc.HtmlBodyText = _imgIcon.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_ImageIcon : WebPage
{
        public ToolKit_ImageIcon()
        {
        }
```

```
  public override void OnInitialized()
  {
    ToolKit.ImageIcon _imgIcon = new ToolKit.ImageIcon();
    _imgIcon.Icon.SetAttribute("title", "image");
    _imgIcon.Icon.SetAttribute("src", ImageAliasPath + "logo.jpg");

    HtmlDoc.HtmlBodyText = _imgIcon.HtmlText();
  }
}
```

Output



WebControl - MenuList

- The MenuList Control creates menu list with title on webpage.

Code Example

Fig1. Visual Basic

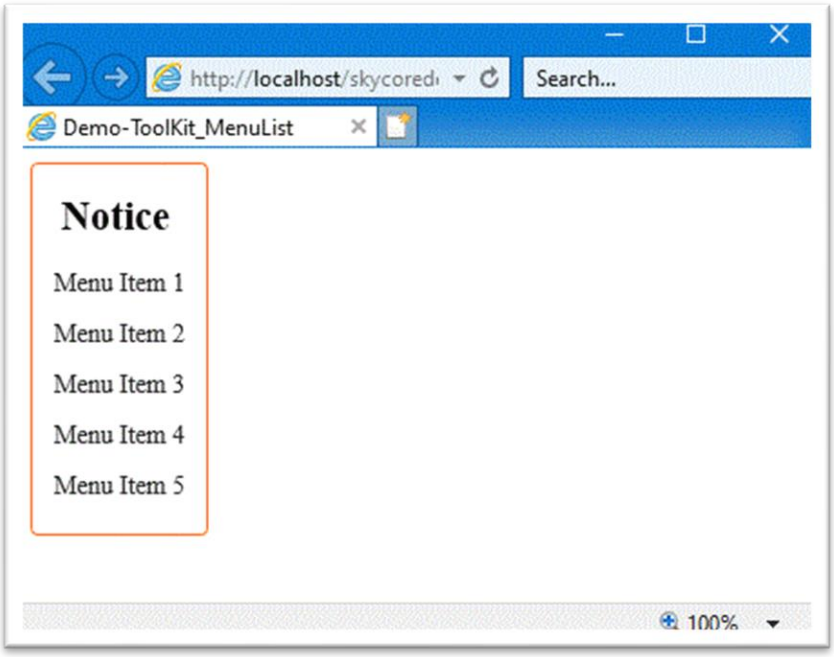
```
Imports Microsoft.VisualBasic
Imports skycore
Public Class ToolKit_MenuList
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim mnu As New ToolKit.MenuList
      mnu.Align = ToolKit.Alignment.Vertical
      mnu.SetStyle(HtmlStyles.border, "1px solid #ff6600")
      mnu.Title.InnerText = "Notice"
      mnu.Add("Menu Item 1")
      mnu.Add("Menu Item 2")
      mnu.Add("Menu Item 3")
      mnu.Add("Menu Item 4")
      mnu.Add("Menu Item 5")

      HtmlDoc.HtmlBodyText = mnu.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_MenuList : WebPage
```

```
{
        public ToolKit_MenuList()
        {
        }
  public override void OnInitialized()
  {
    ToolKit.MenuList mnu = new ToolKit.MenuList();
    mnu.Align = ToolKit.Alignment.Vertical;
    mnu.SetStyle(HtmlStyles.border, "1px solid #ff6600");
    mnu.Title.InnerText = "Notice";
    mnu.Add("Menu Item 1");
    mnu.Add("Menu Item 2");
    mnu.Add("Menu Item 3");
    mnu.Add("Menu Item 4");
    mnu.Add("Menu Item 5");
    HtmlDoc.HtmlBodyText = mnu.HtmlText();
  }
}
```

Output



WebControl - MenuPanel

- The MenuPanel Control creates menu list with title on webpage.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_MenuPanel
    Inherits WebPage

    Public Overrides Sub OnInitialized()

        Dim mp As New ToolKit.MenuPanel
        mp.SetStyle("text-align", "left")
        mp.SetStyle("margin", "auto")

        Dim col1 As New ToolKit.MenuPanel.Column
        col1.SetStyle(HtmlStyles.padding, "10px")
        col1.Title.InnerText = "Column1"
        col1.AddItem("Column1-Item1", "alert('1')")
        col1.AddItem("Column1-Item2")
        col1.AddItem("Column1-Item3")
        col1.AddItem("Column1-Item4")
        col1.AddItem("Column1-Item5")
        mp.Columns.Add(col1)

        Dim col2 As New ToolKit.MenuPanel.Column
        col2.SetStyle(HtmlStyles.padding, "10px")
        col2.Title.InnerText = "Column2"
        col2.AddItem("Column2-Item1", "alert('12')")
```

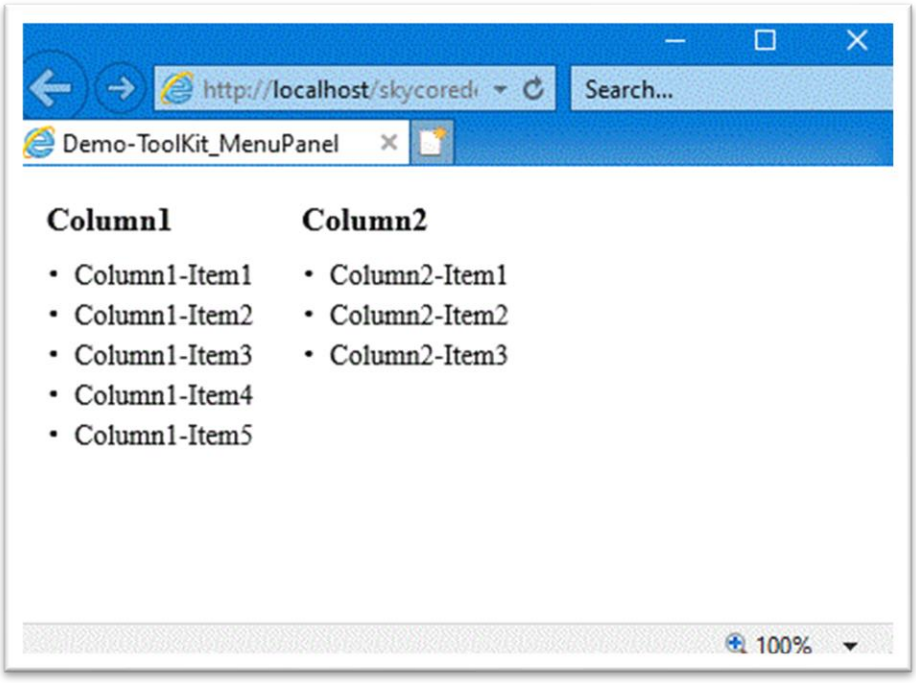
```
    col2.AddItem("Column2-Item2")
    col2.AddItem("Column2-Item3")
    mp.Columns.Add(col2)

    HtmlDoc.HtmlBodyText = mp.HtmlText

  End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_MenuPanel : WebPage
{
        public ToolKit_MenuPanel()
        {
        }
  public override void OnInitialized()
  {
    ToolKit.MenuPanel mp = new ToolKit.MenuPanel();
    mp.SetStyle("text-align", "left");
    mp.SetStyle("margin", "auto");

    ToolKit.MenuPanel.Column col1 = new ToolKit.MenuPanel.Column();
    col1.SetStyle(HtmlStyles.padding, "10px");
    col1.Title.InnerText = "Column1";
    col1.AddItem("Column1-Item1", "alert('1')");
    col1.AddItem("Column1-Item2");
    col1.AddItem("Column1-Item3");
    col1.AddItem("Column1-Item4");
```

```
    col1.AddItem("Column1-Item5");
    mp.Columns.Add(col1);

    ToolKit.MenuPanel.Column col2 = new ToolKit.MenuPanel.Column();
    col2.SetStyle(HtmlStyles.padding, "10px");
    col2.Title.InnerText = "Column1";
    col2.AddItem("Column1-Item1", "alert('1')");
    col2.AddItem("Column1-Item2");
    col2.AddItem("Column1-Item3");
    col2.AddItem("Column1-Item4");
    col2.AddItem("Column1-Item5");
    mp.Columns.Add(col2);

    HtmlDoc.HtmlBodyText = mp.HtmlText();
  }
}
```

Output



WebControl - MultiInputs

- The MultiInputs Control is able to includes multiple html elements in a section.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_MultiInput
    Inherits WebPage

    Public Overrides Sub OnInitialized()

        Dim mInput As New ToolKit.MultiInputs

        Dim elm As New HtmlTag(HtmlTags.div)
        elm.InnerText = "Find Id"
        elm.SetAttribute(HtmlEvents.onclick, "alert(this.innerHTML)")
        elm.SetStyle(HtmlStyles.cursor, "pointer")
        elm.SetStyle(HtmlStyles.margin, "20px")

        Dim elm1 As New HtmlTag(HtmlTags.div)
        elm1.InnerText = "Reset Password"
        elm1.SetAttribute(HtmlEvents.onclick, "alert(this.innerHTML)")
        elm1.SetStyle(HtmlStyles.cursor, "pointer")
        elm1.SetStyle(HtmlStyles.margin, "20px")

        Dim btn As New ToolKit.Button
        btn.SetAttribute(HtmlAttributes.value, "Cancel")
```

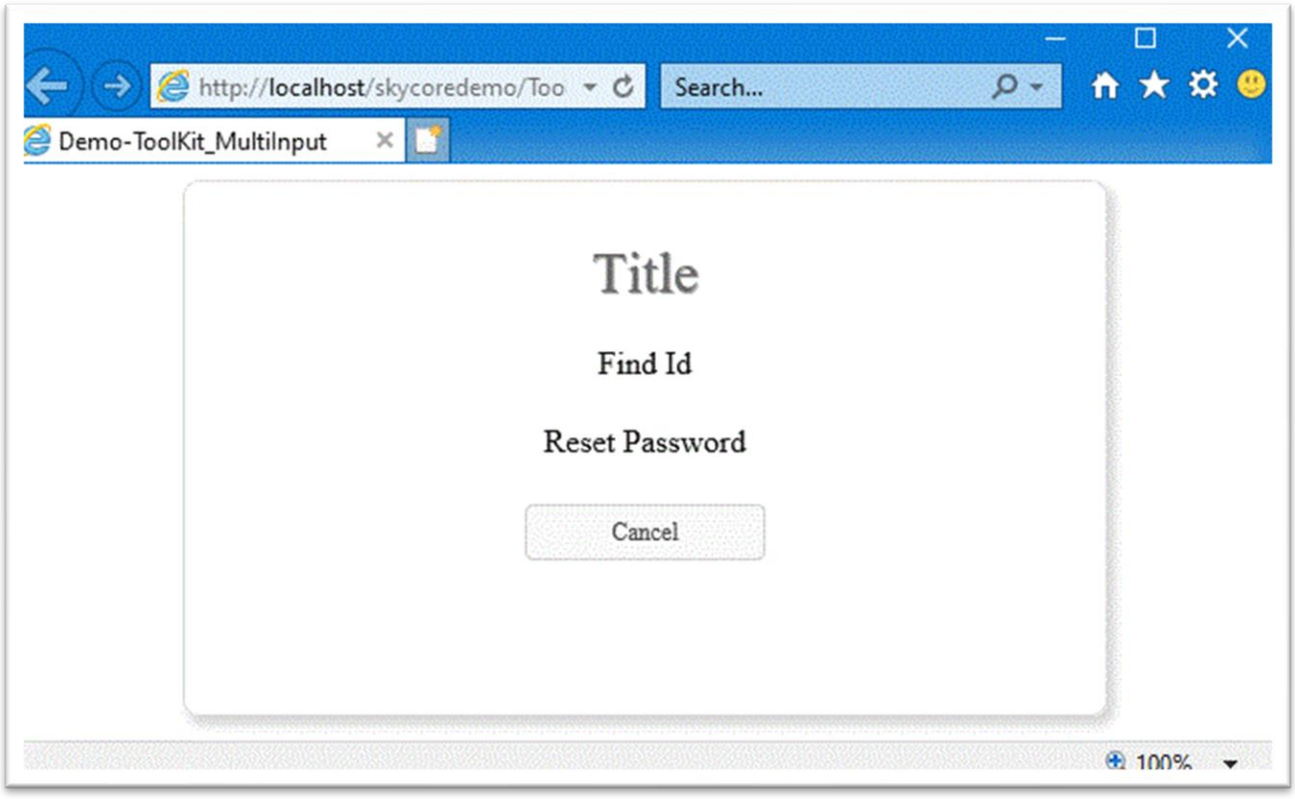
```
    mInput.Items.Add(elm)
    mInput.Items.Add(elm1)
    mInput.Items.Add(btn)

    HtmlDoc.HtmlBodyText = mInput.HtmlText

  End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_MultiInput : WebPage
{
        public ToolKit_MultiInput()
        {
        }
  public override void OnInitialized()
  {
    ToolKit.MultiInputs mInput = new ToolKit.MultiInputs();

    HtmlTag elm = new HtmlTag(HtmlTags.div);
    elm.InnerText = "Find Id";
    elm.SetAttribute(HtmlEvents.onclick, "alert(this.innerHTML)");
    elm.SetStyle(HtmlStyles.cursor, "pointer");
    elm.SetStyle(HtmlStyles.margin, "20px");

    HtmlTag elm1 = new HtmlTag(HtmlTags.div);
    elm1.InnerText = "Reset Password";
    elm1.SetAttribute(HtmlEvents.onclick, "alert(this.innerHTML)");
```

```
    elm1.SetStyle(HtmlStyles.cursor, "pointer");
    elm1.SetStyle(HtmlStyles.margin, "20px");

    ToolKit.Button btn = new ToolKit.Button();
    btn.SetAttribute(HtmlAttributes.value, "Cancel");

    mInput.Items.Add(elm);
    mInput.Items.Add(elm1);
    mInput.Items.Add(btn);

    HtmlDoc.HtmlBodyText = mInput.HtmlText();
  }
}
```

Output



WebControl - Paging

- The Paging Control creates paging interface elements.

Code Example

Fig1. Visual Basic

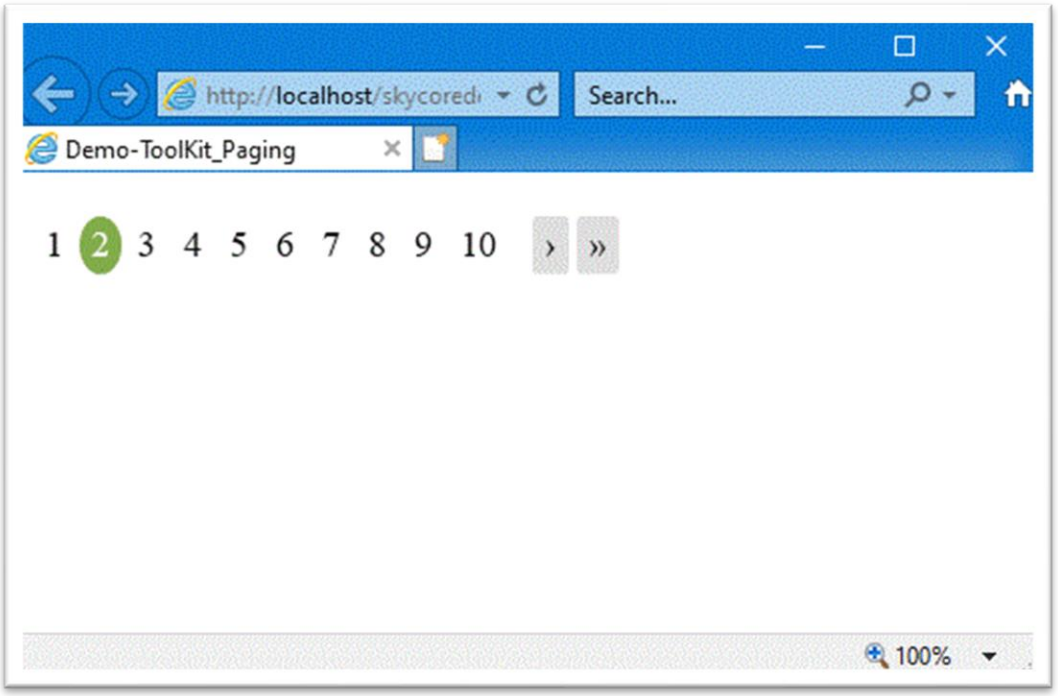
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_Paging
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim TotalRecords As Integer = 1250
      Dim CurrentPage As Integer = 2
      Dim LinePerPage As Integer = 30
      Dim JsEvents As String = "PageClicked"

      Dim _Paging As New ToolKit.Paging(TotalRecords, CurrentPage, LinePerPage, JsEvents)

      HtmlDoc.HtmlBodyText = _Paging.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_Paging : WebPage
{
```

```
        public ToolKit_Paging()
        {
        }
  public override void OnInitialized()
  {
    int TotalRecords = 1250;
    int CurrentPage = 2;
    int LinePerPage = 30;
    string JsEvents = "PageClicked";

    ToolKit.Paging _Paging = new ToolKit.Paging(TotalRecords, CurrentPage, LinePerPage, JsEvents);

    HtmlDoc.HtmlBodyText = _Paging.HtmlText();
  }
}
```

Output



WebControl - Stacker

- The Stacker Control displays html contents horizontally.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_Stacker
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim mnu As New ToolKit.MenuList
      mnu.Align = ToolKit.Alignment.Vertical
      mnu.Title.InnerText = "Column1"
      mnu.Add("Menu Item Column1-1")
      mnu.Add("Menu Item Column1-2")
      mnu.Add("Menu Item Column1-3")

      Dim mnu1 As New ToolKit.MenuList
      mnu1.Align = ToolKit.Alignment.Vertical
      mnu1.Title.InnerText = "Column2"
      mnu1.Add("Menu Item Column2-1")
      mnu1.Add("Menu Item Column2-2")
      mnu1.Add("Menu Item Column2-3")

      Dim _Stacker As New ToolKit.Stacker()
      _Stacker.AddColumn(mnu.HtmlText)
      _Stacker.AddColumn(mnu1.HtmlText)
```

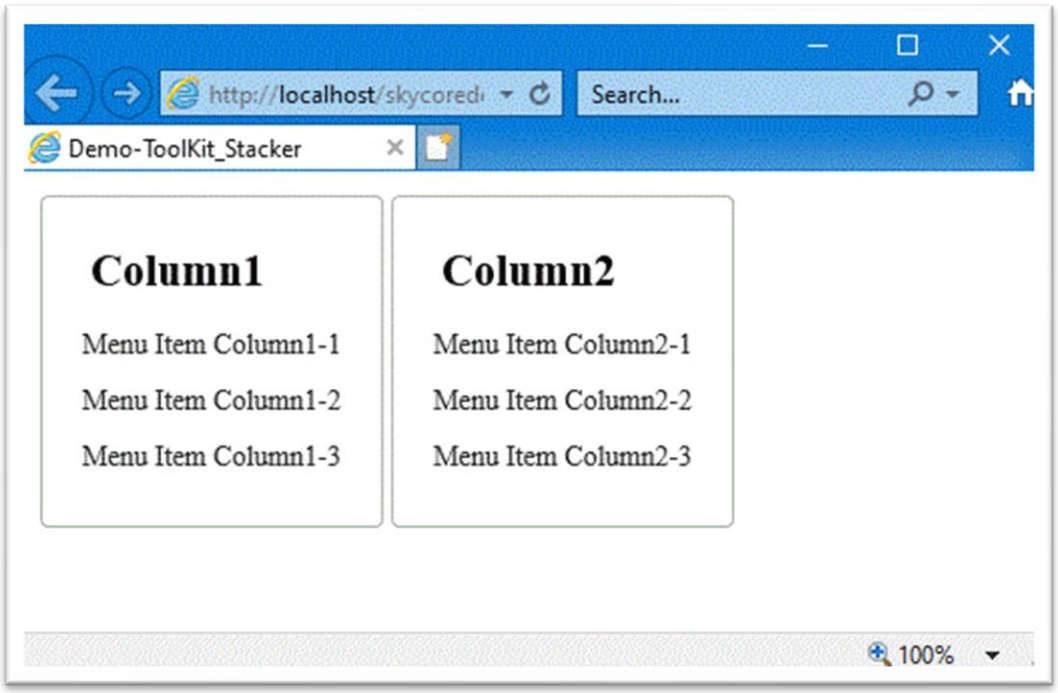
```
    HtmlDoc.HtmlBodyText = _Stacker.HtmlText
  End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_Stacker : WebPage
{
        public ToolKit_Stacker()
        {
        }
  public override void OnInitialized()
  {
    ToolKit.MenuList mnu = new ToolKit.MenuList();
    mnu.Align = ToolKit.Alignment.Vertical;
    mnu.Title.InnerText = "Column1";
    mnu.Add("Menu Item Column1-1");
    mnu.Add("Menu Item Column1-2");
    mnu.Add("Menu Item Column1-3");

    ToolKit.MenuList mnu1 = new ToolKit.MenuList();
    mnu1.Align = ToolKit.Alignment.Vertical;
    mnu1.Title.InnerText = "Column2";
    mnu1.Add("Menu Item Column2-1");
    mnu1.Add("Menu Item Column2-2");
    mnu1.Add("Menu Item Column2-3");

    ToolKit.Stacker _Stacker = new ToolKit.Stacker();
    _Stacker.AddColumn(mnu.HtmlText());
```

```
    _Stacker.AddColumn(mnu1.HtmlText());


    HtmlDoc.HtmlBodyText = _Stacker.HtmlText();
  }
}
```

Output



WebControl - TitleLabel

- The TitleLabel Control creates a Text label on webpage.

Code Example

Fig1. Visual Basic

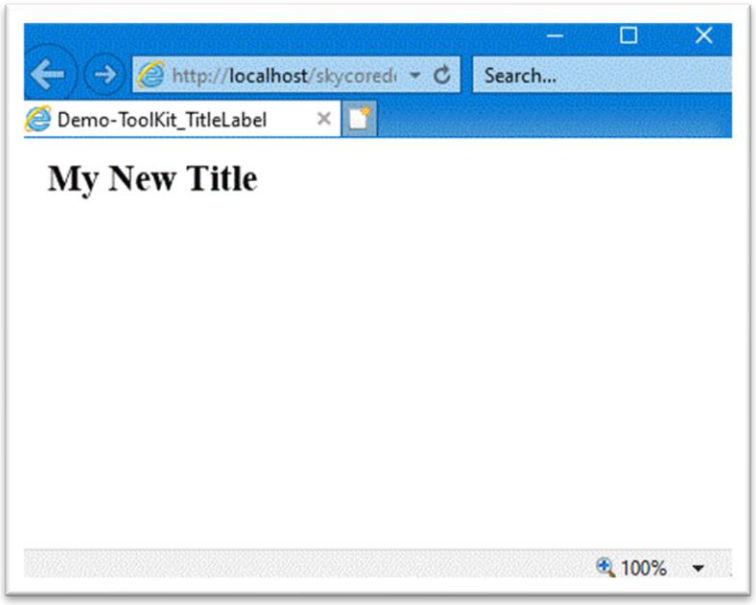
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_TitleLabel
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _TitleLabel As New ToolKit.TitleLabel
      _TitleLabel.InnerText = "My New Title"

      HtmlDoc.HtmlBodyText = _TitleLabel.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using skycore;

public class ToolKit_TitleLabel : WebPage
{
        public ToolKit_TitleLabel()
        {
        }
   public override void OnInitialized()
```

```
  {
    ToolKit.TitleLabel _TitleLabel = new ToolKit.TitleLabel();
    _TitleLabel.InnerText = "My New Title";

    HtmlDoc.HtmlBodyText = _TitleLabel.HtmlText();
  }
}
```

Output



WebControl - TreeView

- The TreeView Control displays a hierarchical list with expanding/collapsing nodes that contain nested items.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore
Imports System.Data

Public Class ToolKit_TreeView
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.AddJsFile(WebEnv.HeaderScripts.TreeScript)

      Dim dt As New DataTable
      dt.Columns.Add("Code")
      dt.Columns.Add("Name")
      dt.Columns.Add("Top")

      dt.Rows.Add({"1000", "SKYLINK", ""})
      dt.Rows.Add({"1100", "Development Platforms", "1000"})
      dt.Rows.Add({"1200", "Downloads", "1000"})
      dt.Rows.Add({"1300", "Prerequisite", "1000"})
      dt.Rows.Add({"1400", "AJAX Fundamental", "1000"})
      dt.Rows.Add({"1410", "ApiRequest", "1400"})
      dt.Rows.Add({"1420", "ApiResponse", "1400"})

      Dim TreeItems As New List(Of ToolKit.TreeView.TreeItem)
      If dt IsNot Nothing AndAlso dt.Rows.Count <> 0 Then
```

```
        For i As Integer = 0 To dt.Rows.Count - 1
            Dim titm As New ToolKit.TreeView.TreeItem With {.Id = dt.Rows(i)(0).ToString, _
                                        .Name = dt.Rows(i)(1).ToString, _
                                        .ParentId = dt.Rows(i)(2).ToString, _
                                        .Status = 0}

            titm.Item.SetAttribute("onclick", "ItemSelected(this, '" + dt.Rows(i)(0).ToString + "')")
            TreeItems.Add(titm)
        Next
    End If

    Dim TreeView As New ToolKit.TreeView With {.StartingId = "1000", _
                            .ImageOpen = ImageAliasPath + "tvopn.jpg", _
                            .ImageClose = ImageAliasPath + "tvhid.jpg", _
                            .ImageLast = ImageAliasPath + "tvlast.jpg", _
                            .TreeItems = TreeItems}

    HtmlDoc.HtmlBodyText = TreeView.HtmlText
  End Sub
End Class
```

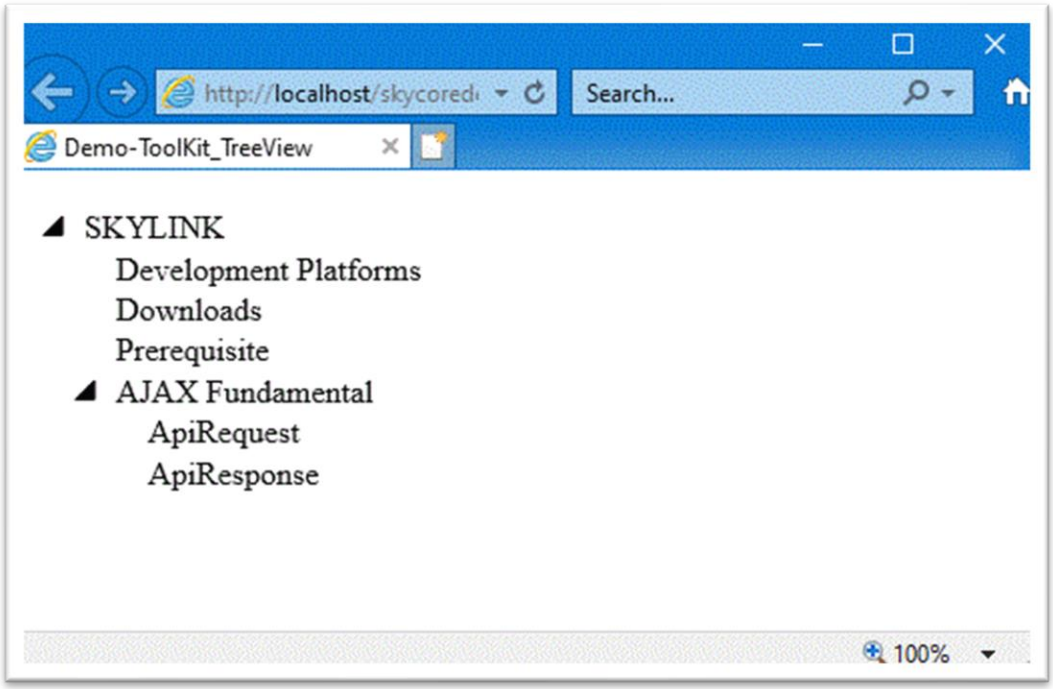
Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;
using System.Data;

public class ToolKit_TreeView : WebPage
{
        public ToolKit_TreeView()
        {
```

```
        }
public override void OnInitialized()
{
                    HtmlDoc.AddJsFile(WebEnv.HeaderScripts.TreeScript);

   DataTable dt = new DataTable();
   dt.Columns.Add("Code");
   dt.Columns.Add("Name");
   dt.Columns.Add("Top");

   dt.Rows.Add("1000", "SKYLINK", "");
   dt.Rows.Add("1100", "Development Platforms", "1000");
   dt.Rows.Add("1200", "Downloads", "1000");
   dt.Rows.Add("1300", "Prerequisite", "1000");
   dt.Rows.Add("1400", "AJAX Fundamental", "1000");
   dt.Rows.Add("1410", "ApiRequest", "1400");
   dt.Rows.Add("1420", "ApiResponse", "1400");

   List<ToolKit.TreeView.TreeItem> TreeItems = new List<ToolKit.TreeView.TreeItem>();
   if(dt != null && dt.Rows.Count != 0){
      for (int i = 0; i < dt.Rows.Count; i++) {
         ToolKit.TreeView.TreeItem titm = new ToolKit.TreeView.TreeItem() {
            Id = dt.Rows[i][0].ToString(),
            Name = dt.Rows[i][1].ToString(),
            ParentId = dt.Rows[i][2].ToString(),
            Status = 0 };

         titm.Item.SetAttribute("onclick", "ItemSelected(this, '" + dt.Rows[i][0].ToString() + "')");
         TreeItems.Add(titm);
      }
   }
```

```
    ToolKit.TreeView TreeView = new ToolKit.TreeView() {
            StartingId = "1000",
            ImageOpen = ImageAliasPath + "tvopn.jpg",
            ImageClose = ImageAliasPath + "tvhid.jpg",
            ImageLast = ImageAliasPath + "tvlast.jpg",
            TreeItems = TreeItems};

    HtmlDoc.HtmlBodyText = TreeView.HtmlText();
  }
}
```

Output



WebControl - Wrap

- The Wrap Control displays all the html elements together in a Wrap control as a section on the web page.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_DataList
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim filter As New ToolKit.TextSearch
      filter.Label.InnerText = "Search Term"
      filter.Text.SetAttribute(HtmlAttributes.list, "StudentList")

      Dim _NameValues As New List(Of NameValue)
      _NameValues.Add(New NameValue With {.name = "Alex", .value = "90"})
      _NameValues.Add(New NameValue With {.name = "David", .value = "87"})
      _NameValues.Add(New NameValue With {.name = "Julie", .value = "91"})
      _NameValues.Add(New NameValue With {.name = "Rose", .value = "75"})

      Dim _datalist As New ToolKit.DataList
      _datalist.SetAttribute(HtmlAttributes.id, "StudentList")
      _datalist.SetOption(_NameValues)

      Dim _wrap As New ToolKit.Wrap
      _wrap.AddItem(filter.HtmlText)
      _wrap.AddItem(_datalist.HtmlText)
```

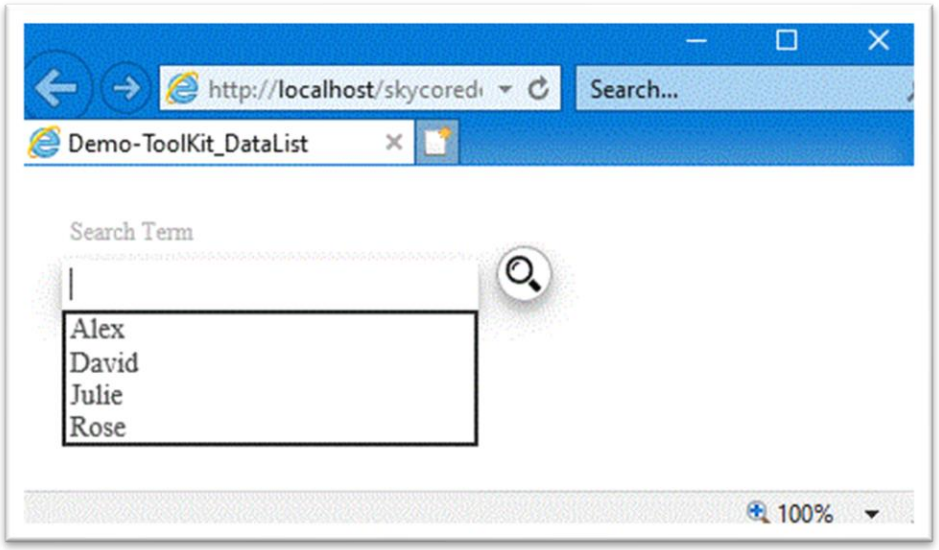
```
    HtmlDoc.HtmlBodyText = _wrap.HtmlText
  End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_DataList : WebPage
{
        public ToolKit_DataList()
        {
        }
  public override void OnInitialized()
  {
    ToolKit.TextSearch filter = new ToolKit.TextSearch();
    filter.Label.InnerText = "Search Term";
    filter.Text.SetAttribute(HtmlAttributes.list, "StudentList");

    List<NameValue> _NameValues = new List<NameValue>();
    _NameValues.Add(new NameValue() {name = "Alex", value = "90"});
    _NameValues.Add(new NameValue() {name = "David", value = "87"});
    _NameValues.Add(new NameValue() {name = "Julie", value = "91"});
    _NameValues.Add(new NameValue() {name = "Rose", value = "75"});

    ToolKit.DataList _datalist = new ToolKit.DataList();
     _datalist.SetAttribute(HtmlAttributes.id, "StudentList");
     _datalist.SetOption(_NameValues);
    ToolKit.Wrap _wrap = new ToolKit.Wrap();
    _wrap.AddItem(filter.HtmlText());
```

```
    _wrap.AddItem(_datalist.HtmlText());


    HtmlDoc.HtmlBodyText = _wrap.HtmlText();
  }
}
```

Output



WebControl - Page.Home

- The Page.Home Control displays all the html elements together in a Wrap control as a section on the web page.
- A logo, title, menu and so on are customizable.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore
Imports System.Data

Public Class ToolKit_PageHome
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.AddCSSFile("xStyle.css")
      HtmlDoc.SetTitle("Home")

      Dim mnuitm1000 As New ToolKit.Pages.Home.MenuItem With {.Id = "1000", .Name = "Menu1000"}
      Dim mnuitm1010 As New ToolKit.Pages.Home.MenuItem With {.Id = "1010", .Name = "Menu1010"}
      Dim mnuitm1020 As New ToolKit.Pages.Home.MenuItem With {.Id = "1020", .Name = "Menu1020"}
      Dim mnuitm1030 As New ToolKit.Pages.Home.MenuItem With {.Id = "1030", .Name = "Menu1030"}

      Dim mnuitm2000 As New ToolKit.Pages.Home.MenuItem With {.Id = "2000", .Name = "Menu2000"}
      Dim mnuitm2010 As New ToolKit.Pages.Home.MenuItem With {.Id = "2010", .Name = "Menu2010"}
      Dim mnuitm2020 As New ToolKit.Pages.Home.MenuItem With {.Id = "2020", .Name = "Menu2020"}
      Dim mnuitm2030 As New ToolKit.Pages.Home.MenuItem With {.Id = "2030", .Name = "Menu2030"}

      Dim mnu As New List(Of ToolKit.Pages.Home.MenuItem)
      mnu.Add(mnuitm1000)
      mnu.Add(mnuitm1010)
```

```
        mnu.Add(mnuitm1020)
        mnu.Add(mnuitm1030)

        Dim mnu1 As New List(Of ToolKit.Pages.Home.MenuItem)
        mnu1.Add(mnuitm2000)
        mnu1.Add(mnuitm2010)
        mnu1.Add(mnuitm2020)
        mnu1.Add(mnuitm2030)

        Dim _Home As New ToolKit.Pages.Home()
        _Home.AddUserIconButton("Profile", "UserProfile()")
        _Home.AddUserIconButton("Change Password", "UserChgPwd()")
        _Home.AddUserIconButton("Sign Out", "SignOut()")

        _Home.Functions.Buttons.Add("Manage Notice", "onclick:alert('1')")
        _Home.AddMenu("Menu 1", mnu)
        _Home.AddMenu("Menu 2", mnu1)
        HtmlDoc.HtmlBodyText = _Home.HtmlText
    End Sub
End Class
```

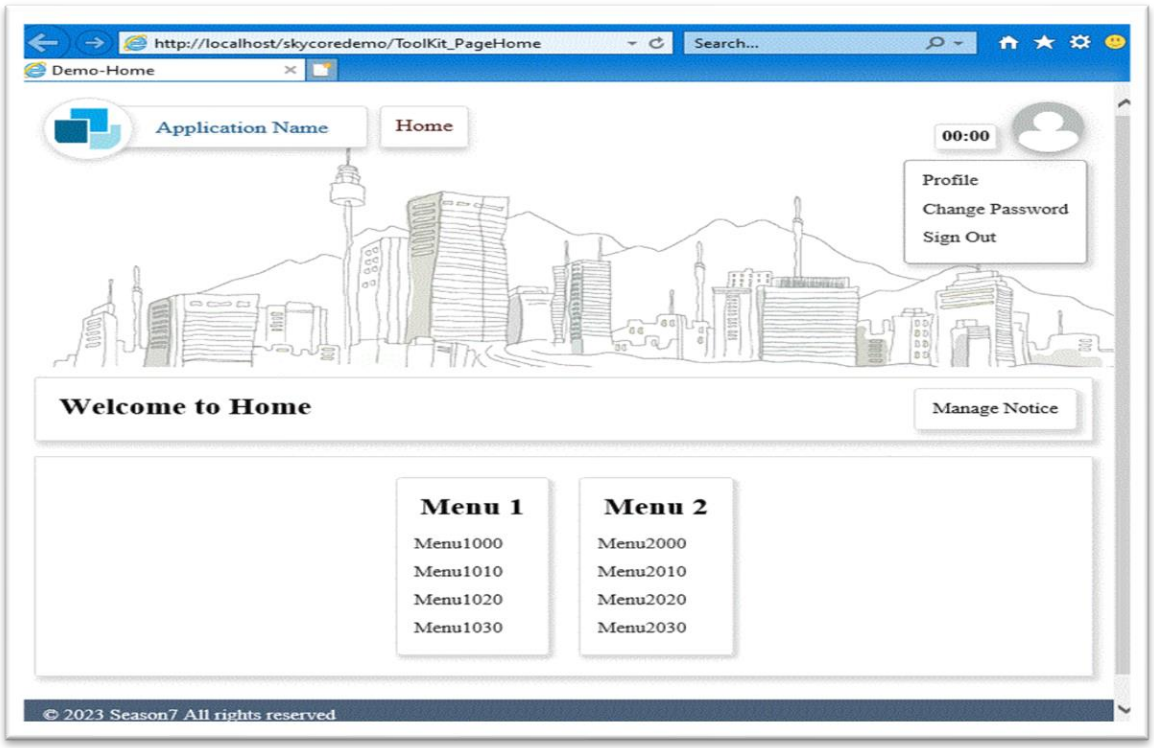
Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;
using System.Data;

public class ToolKit_PageHome : WebPage
{
        public ToolKit_PageHome()
        {
```

```
        }
    public override void OnInitialized()
    {
        HtmlDoc.SetTitle("Home");

        ToolKit.Pages.Home.MenuItem mnuitm1000 = new ToolKit.Pages.Home.MenuItem(){Id = "1000", Name =
"Menu1000"};
        ToolKit.Pages.Home.MenuItem mnuitm1010 = new ToolKit.Pages.Home.MenuItem(){Id = "1010", Name =
"Menu1010"};
        ToolKit.Pages.Home.MenuItem mnuitm1020 = new ToolKit.Pages.Home.MenuItem(){Id = "1020", Name =
"Menu1020"};
        ToolKit.Pages.Home.MenuItem mnuitm1030 = new ToolKit.Pages.Home.MenuItem(){Id = "1030", Name =
"Menu1030"};

        ToolKit.Pages.Home.MenuItem mnuitm2000 = new ToolKit.Pages.Home.MenuItem(){Id = "2000", Name =
"Menu2000"};
        ToolKit.Pages.Home.MenuItem mnuitm2010 = new ToolKit.Pages.Home.MenuItem(){Id = "2010", Name =
"Menu2010"};
        ToolKit.Pages.Home.MenuItem mnuitm2020 = new ToolKit.Pages.Home.MenuItem(){Id = "2020", Name =
"Menu2020"};
        ToolKit.Pages.Home.MenuItem mnuitm2030 = new ToolKit.Pages.Home.MenuItem(){Id = "2030", Name =
"Menu2030"};

        List<ToolKit.Pages.Home.MenuItem> mnu = new List<ToolKit.Pages.Home.MenuItem>();
        List<ToolKit.Pages.Home.MenuItem> mnu1 = new List<ToolKit.Pages.Home.MenuItem>();

        mnu.Add(mnuitm1000);
        mnu.Add(mnuitm1010);
        mnu.Add(mnuitm1020);
        mnu.Add(mnuitm1030);
        mnu1.Add(mnuitm2000);
        mnu1.Add(mnuitm2010);
        mnu1.Add(mnuitm2020);
        mnu1.Add(mnuitm2030);
```

```
    ToolKit.Pages.Home _Home = new ToolKit.Pages.Home();
    _Home.AddUserIconButton("Profile", "UserProfile()");
    _Home.AddUserIconButton("Change Password", "UserChgPwd()");
    _Home.AddUserIconButton("Sign Out", "SignOut()");

    _Home.Functions.Buttons.Add("Manage Notice", "onclick:alert('1')");
    _Home.AddMenu("Menu 1", mnu);
    _Home.AddMenu("Menu 2", mnu1);

    HtmlDoc.HtmlBodyText = _Home.HtmlText();
  }
}
```

Output



WebControl - Page.Custom

- The Page.Custom Control displays all the html elements together in a Wrap control as a section on the web page.
- A logo, title, menu and so on are customizable.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore
Imports System.Data

Public Class ToolKit_PageCustom
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.SetTitle("Home")

      Dim _Custom As New ToolKit.Pages.Custom()
      _Custom.AddUserIconButton("Profile", "UserProfile()")
      _Custom.AddUserIconButton("Change Password", "UserChgPwd()")
      _Custom.AddUserIconButton("Sign Out", "SignOut()")

      Dim titlbl As New ToolKit.TitleLabel With {.InnerText = "Custom Contents"}

      Dim contentTitle As New ToolKit.ContentsBox With {.InnerContents = titlbl.HtmlText, .Border = True}
      Dim contentFilter As New ToolKit.Sections.Filter.TextMonth
      contentFilter.Buttons.Add("Add New")
      contentFilter.Buttons.Add("Update")
      contentFilter.Buttons.Add("Close")

      _Custom.Contents.AddContents(contentTitle.HtmlText)
```

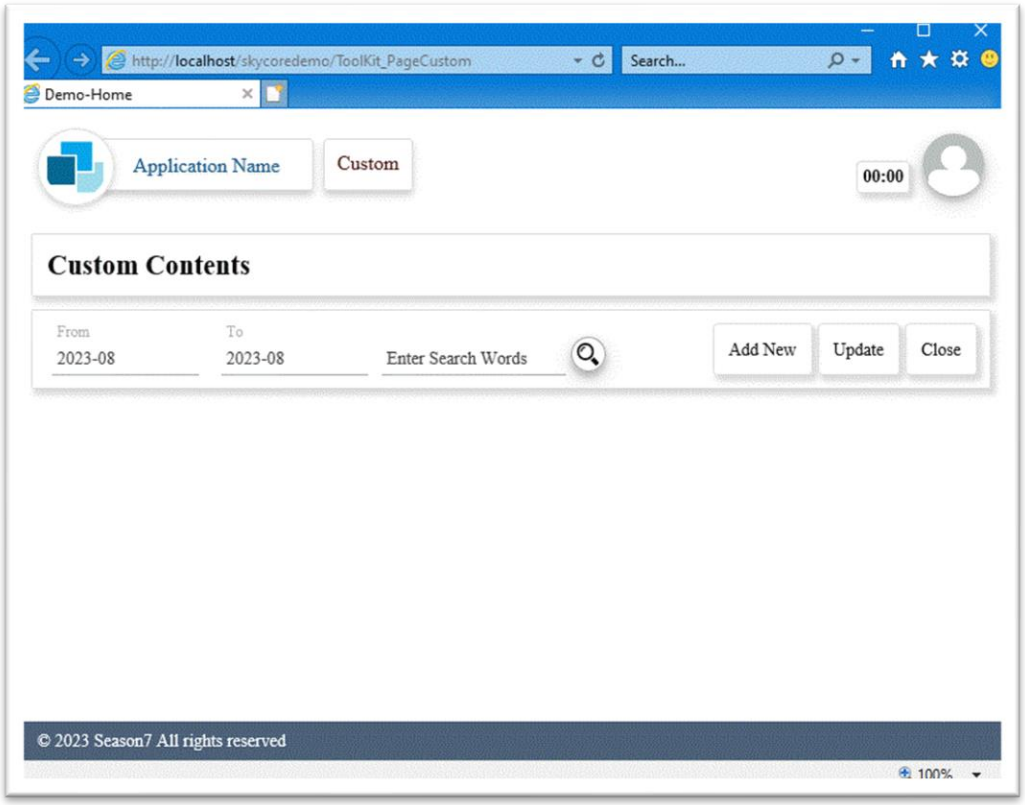
```
    _Custom.Contents.AddContents(contentFilter.HtmlText)

    HtmlDoc.HtmlBodyText = _Custom.HtmlText
  End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;
using System.Data;

public class ToolKit_PageCustom : WebPage
{
        public ToolKit_PageCustom()
        {
        }
  public override void OnInitialized()
  {
    HtmlDoc.SetTitle("Home");

    ToolKit.Pages.Custom _Custom = new ToolKit.Pages.Custom();

    _Custom.AddUserIconButton("Profile", "UserProfile()");
    _Custom.AddUserIconButton("Change Password", "UserChgPwd()");
    _Custom.AddUserIconButton("Sign Out", "SignOut()");

    ToolKit.TitleLabel titlbl = new ToolKit.TitleLabel() { InnerText = "Custom Contents" };
    ToolKit.ContentsBox contentTitle = new ToolKit.ContentsBox() {InnerContents = titlbl.HtmlText(), Border = true };
    ToolKit.Sections.Filter.TextMonth contentFilter = new ToolKit.Sections.Filter.TextMonth();
    contentFilter.Buttons.Add("Add New");
```

```
    contentFilter.Buttons.Add("Update");
    contentFilter.Buttons.Add("Close");

    _Custom.Contents.AddContents(contentTitle.HtmlText());
    _Custom.Contents.AddContents(contentFilter.HtmlText());

    HtmlDoc.HtmlBodyText = _Custom.HtmlText();
  }
}
```

Output



WebControl - Page.Generic

- The Page.Generic Control displays all the html elements together in a Wrap control as a section on the web page.
- A logo, title, menu and so on are customizable.

Code Example

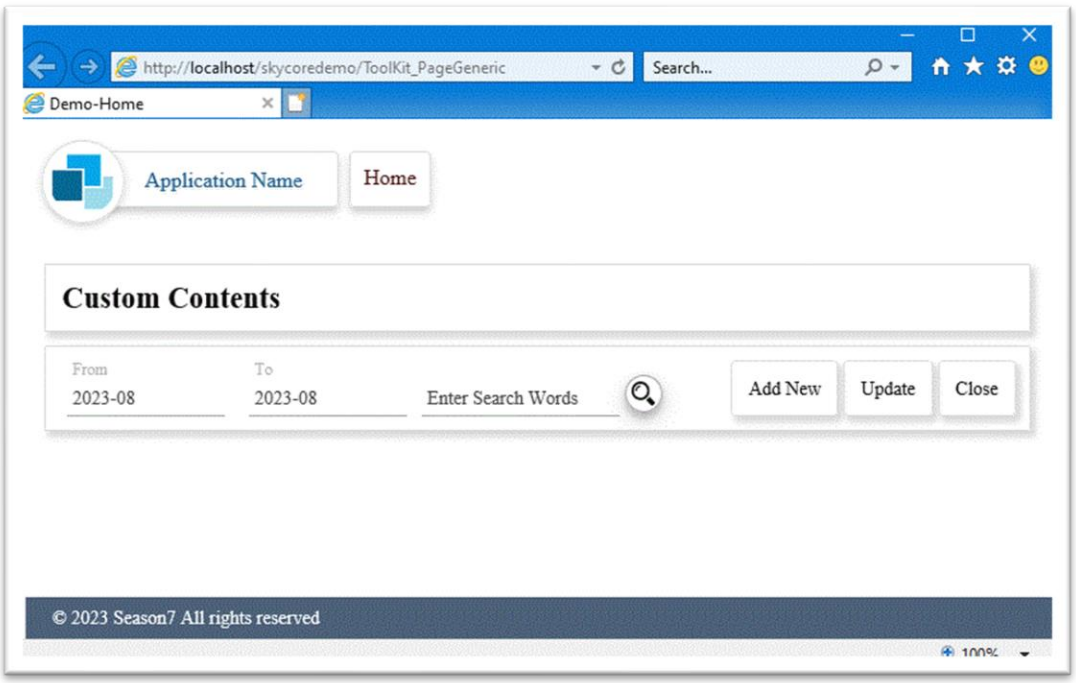
Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore
Imports System.Data
Public Class ToolKit_PageGeneric
    Inherits WebPage

    Public Overrides Sub OnInitialized()
        HtmlDoc.SetTitle("Home")

        Dim titlbl As New ToolKit.TitleLabel With {.InnerText = "Custom Contents"}
        Dim contentTitle As New ToolKit.ContentsBox With {.InnerContents = titlbl.HtmlText, .Border = True}
        Dim contentFilter As New ToolKit.Sections.Filter.TextMonth
        contentFilter.Buttons.Add("Add New")
        contentFilter.Buttons.Add("Update")
        contentFilter.Buttons.Add("Close")

        Dim _GenericPage As New ToolKit.Pages.Generic()
        _GenericPage.TitleBar.Title.InnerText = "Application Name"
        _GenericPage.TitleBar.Page.InnerText = "Home"
        _GenericPage.Contents.AddContents(contentTitle.HtmlText)
        _GenericPage.Contents.AddContents(contentFilter.HtmlText)

        HtmlDoc.HtmlBodyText = _GenericPage.HtmlText
    End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;
using System.Data;

public class ToolKit_PageGeneric : WebPage
{
        public ToolKit_PageGeneric()
        {
        }
   public override void OnInitialized()
   {
      HtmlDoc.SetTitle("Home");

      ToolKit.TitleLabel titlbl = new ToolKit.TitleLabel() { InnerText = "Custom Contents" };
      ToolKit.ContentsBox contentTitle = new ToolKit.ContentsBox() { InnerContents = titlbl.HtmlText(), Border = true };
      ToolKit.Sections.Filter.TextMonth contentFilter = new ToolKit.Sections.Filter.TextMonth();
      contentFilter.Buttons.Add("Add New");
      contentFilter.Buttons.Add("Update");
      contentFilter.Buttons.Add("Close");

      ToolKit.Pages.Generic _GenericPage = new ToolKit.Pages.Generic();
      _GenericPage.TitleBar.Title.InnerText = "Application Name";
      _GenericPage.TitleBar.Page.InnerText = "Home";
      _GenericPage.Contents.AddContents(contentTitle.HtmlText());
      _GenericPage.Contents.AddContents(contentFilter.HtmlText());

      HtmlDoc.HtmlBodyText = _GenericPage.HtmlText();
   }
}
```

Output


http://localhost/skycoredemo/ToolKit_PageGeneric
Search...
Demo-Home
Application Name
Home
Custom Contents
From
2023-08
To
2023-08
Enter Search Words
Add New
Update
Close
© 2023 Season7 All rights reserved

WebControl - Page.Login

- The Page.Login Control displays generic login webpage.
- A logo, title and so on are customizable.

Code Example

Fig1. Visual Basic

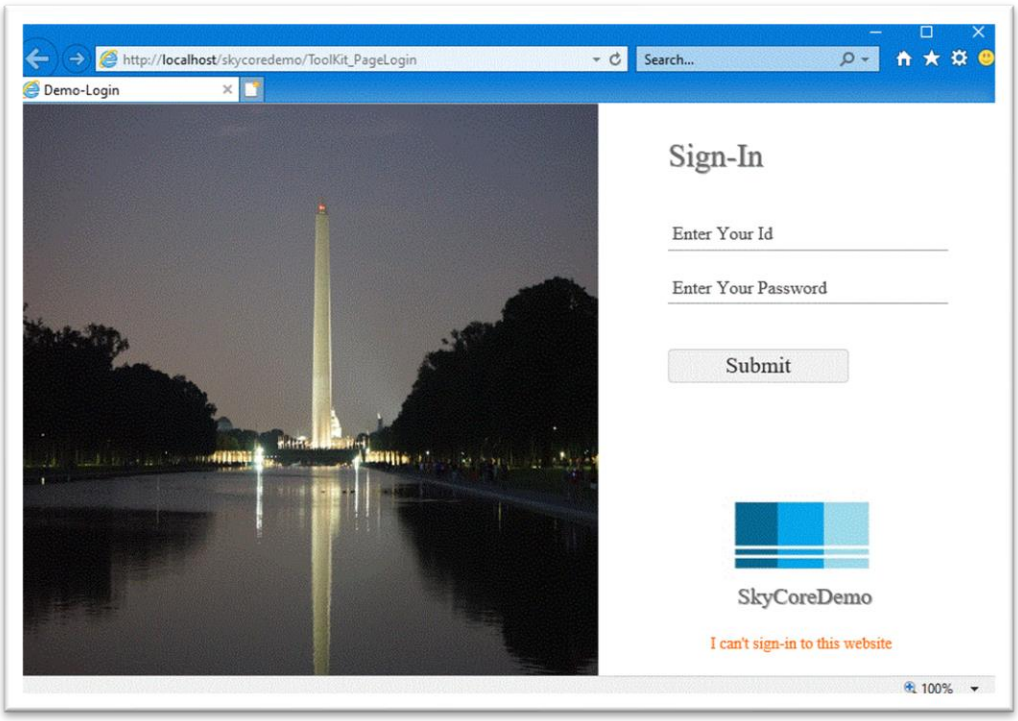
```
Imports Microsoft.VisualBasic
Imports skycore
Imports System.Data

Public Class ToolKit_PageLogin
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.SetTitle("Login")

      Dim LogIn As New ToolKit.Pages.Login
      HtmlDoc.HtmlBodyText = LogIn.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;
using System.Data;

public class ToolKit_PageLogin : WebPage
{
        public ToolKit_PageLogin()
        {
```

```
        }
  public override void OnInitialized()
  {
     HtmlDoc.SetTitle("Home");

     ToolKit.Pages.Login LogIn = new ToolKit.Pages.Login();
     HtmlDoc.HtmlBodyText = LogIn.HtmlText();
  }
}
```

Output



WebControl - Page.LoginIssues

- The Page.LoginIssues Control displays generic login troubleshooting webpage.
- A logo, title and so on are customizable.

Code Example

Fig1. Visual Basic

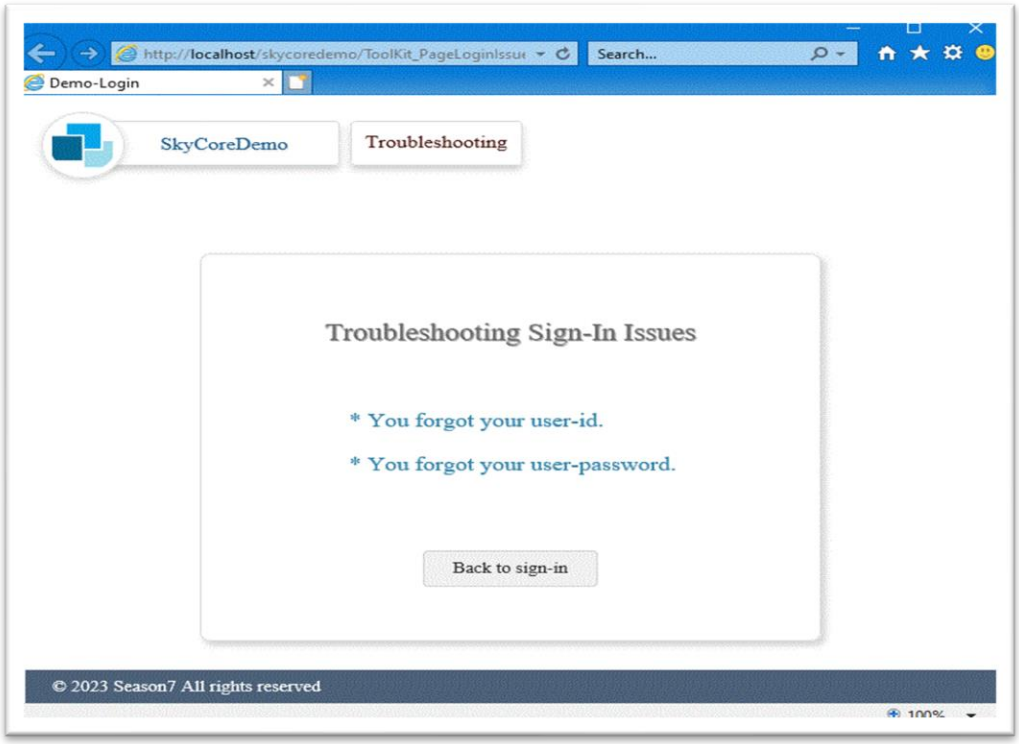
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_PageLoginIssues
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.SetTitle("Login")

      Dim LogIn As New ToolKit.Pages.LoginIssues
      HtmlDoc.HtmlBodyText = LogIn.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;
public class ToolKit_PageLoginIssues : WebPage
{
        public ToolKit_PageLoginIssues()
        {
        }
   public override void OnInitialized()
   {
```

```
    HtmlDoc.SetTitle("Home");


    ToolKit.Pages.LoginIssues LogIn = new ToolKit.Pages.LoginIssues();
    HtmlDoc.HtmlBodyText = LogIn.HtmlText();
  }
}
```

Output



WebControl - Page.LostId

- The Page.LostId Control displays Lost-Id troubleshooting webpage.
- A logo, title and so on are customizable.

Code Example

Fig1. Visual Basic

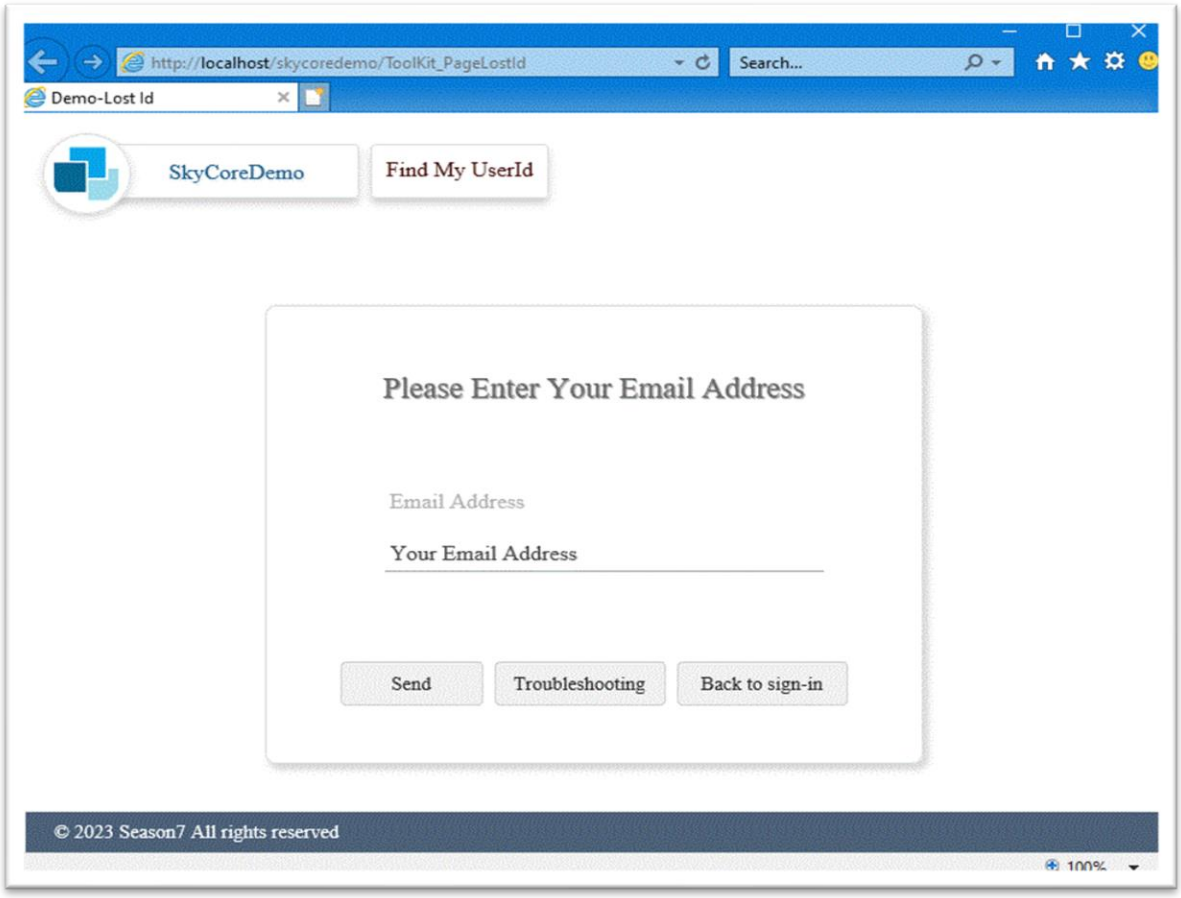
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_PageLostId
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.SetTitle("Lost Id")

      Dim LostId As New ToolKit.Pages.LostId
      HtmlDoc.HtmlBodyText = LostId.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_PageLostId : WebPage
{
        public ToolKit_PageLostId()
        {
        }
   public override void OnInitialized()
   {
```

```
    HtmlDoc.SetTitle("LostId");


    ToolKit.Pages.LostId LostId = new ToolKit.Pages.LostId();
    HtmlDoc.HtmlBodyText = LostId.HtmlText();
  }
}
```

Output



WebControl - Page.LostPassword

- The Page.LostPassword Control displays Lost-Password troubleshooting webpage.
- A logo, title and so on are customizable.

Code Example

Fig1. Visual Basic

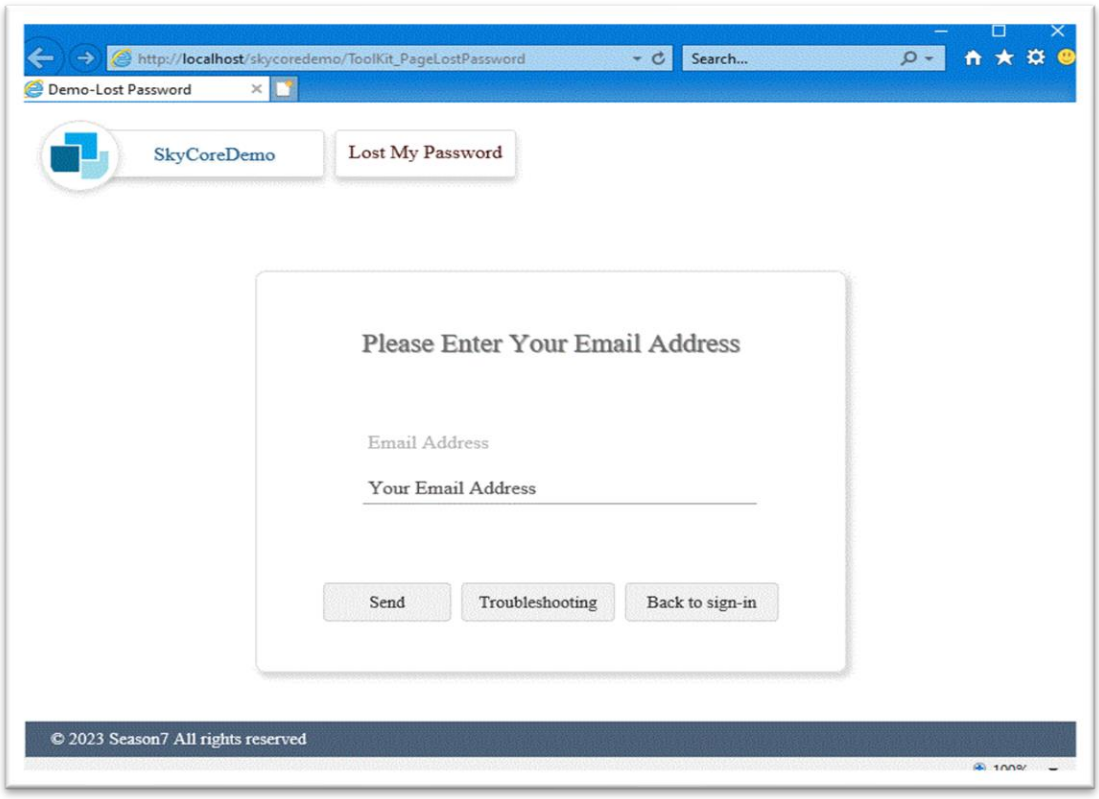
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_PageLostPassword
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.SetTitle("Lost Password")

      Dim Lost As New ToolKit.Pages.LostPassword
      HtmlDoc.HtmlBodyText = Lost.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_PageLostPassword : WebPage
{
        public ToolKit_PageLostPassword()
        {
        }
```

```
  public override void OnInitialized()
  {
     HtmlDoc.SetTitle("Lost Password");

     ToolKit.Pages.LostPassword Lost = new ToolKit.Pages.LostPassword();
     HtmlDoc.HtmlBodyText = Lost.HtmlText();
  }
}
```

Output



WebControl - Page.Message

- The Page.Message Control displays a system message on webpage.
- A logo, title and so on are customizable.

Code Example

Fig1. Visual Basic

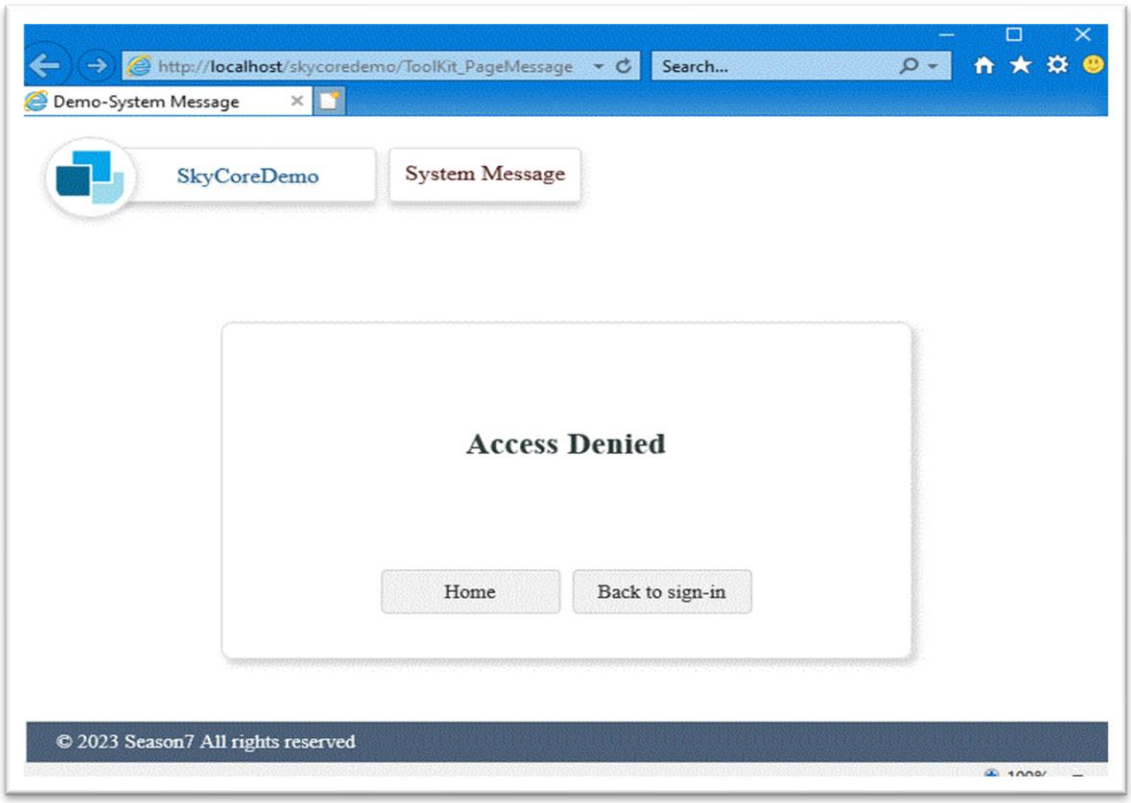
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_PageLostPassword
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.SetTitle("Lost Password")

      Dim Lost As New ToolKit.Pages.LostPassword
      HtmlDoc.HtmlBodyText = Lost.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_PageLostPassword : WebPage
{
        public ToolKit_PageLostPassword()
        {
        }
   public override void OnInitialized()
```

```
  {
    HtmlDoc.SetTitle("Lost Password");

    ToolKit.Pages.LostPassword Lost = new ToolKit.Pages.LostPassword();
    HtmlDoc.HtmlBodyText = Lost.HtmlText();
  }
}
```

Output



WebControl - Page.ResetPassword

- The Page.Message Control displays a change/reset password interface webpage.
- A logo, title and so on are customizable.

Code Example

Fig1. Visual Basic

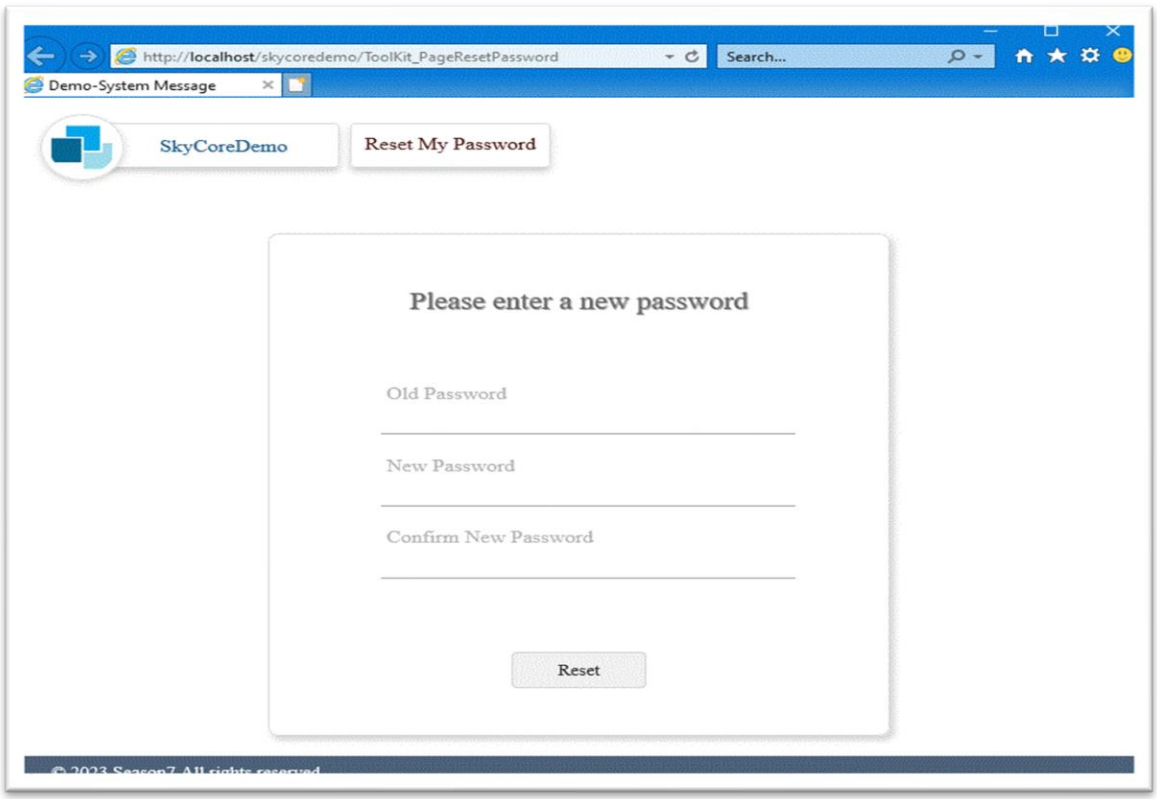
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_PageResetPassword
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.SetTitle("Reset Password")

      Dim page As New ToolKit.Pages.ResetPassword
      HtmlDoc.HtmlBodyText = page.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_PageResetPassword : WebPage
{
        public ToolKit_PageResetPassword()
        {
        }
```

```
  public override void OnInitialized()
  {
    HtmlDoc.SetTitle("Reset Password");

    ToolKit.Pages.ResetPassword page = new ToolKit.Pages.ResetPassword();
    HtmlDoc.HtmlBodyText = page.HtmlText();
  }
}
```

Output



WebControl - Page.ColumnStacker

- The Page.ColumnStacker Control displays horizontal stacked contents columns.
- A logo, title and so on are customizable.

Code Example

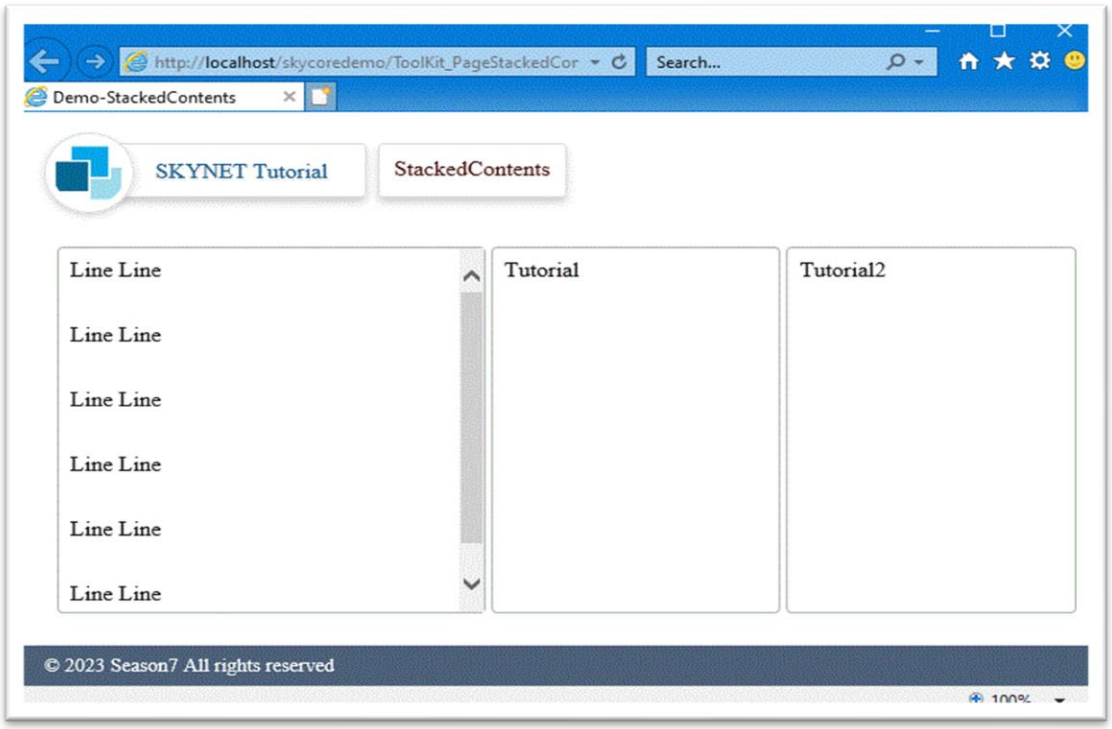
Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_PageStackedContents
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.SetTitle("ColumnStacker")

      Dim page As New ToolKit.Pages.ColumnStacker
      page.UserIcon.Visible = False
      page.TitleBar.Title.InnerText = "SKYNET Tutorial"
      page.TitleBar.Page.InnerText = "StackedContents"

      Dim contents As String = "<div style=""height:50px"">Line Line</div>" + "<div style=""height:50px"">Line Line</div>" + "<div style=""height:50px"">Line Line</div>" + "<div style=""height:50px"">Line Line</div>" + "<div style=""height:50px"">Line Line</div>" + "<div style=""height:50px"">Line Line</div>" + "<div style=""height:50px"">Line Line</div>"

      page.AddStackColumn(contents, "width:300px;")
      page.AddStackColumn("Tutorial", "width:calc(50% - 150px);")
      page.AddStackColumn("Tutorial2", "width:calc(50% - 150px);")

      HtmlDoc.HtmlBodyText = page.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_PageColumnStacker : WebPage
{
   public ToolKit_PageColumnStacker()
            {
            }
   public override void OnInitialized()
   {
      HtmlDoc.SetTitle("ColumnStacker");

      ToolKit.Pages.ColumnStacker page = new ToolKit.Pages.ColumnStacker();
      page.UserIcon.Visible = false ;
      page.TitleBar.Title.InnerText = "SKYNET Tutorial";
      page.TitleBar.Page.InnerText = "StackedContents";

      string contents = @"<div style=""height:50px"">Line Line</div>" + @"<div style=""height:50px"">Line Line</div>" +
@"<div style=""height:50px"">Line Line</div>";
      page.AddStackColumn(contents, "width:300px;");
      page.AddStackColumn("Tutorial", "width:calc(50% - 150px);");
      page.AddStackColumn("Tutorial2", "width:calc(50% - 150px);");

      HtmlDoc.HtmlBodyText = page.HtmlText();
   }
}
```

Output


http://localhost/skycoredemo/ToolKit_PageStackedCor
Search...
Demo-StackedContents
SKYNET Tutorial
StackedContents
Line Line
Line Line
Line Line
Line Line
Line Line
Line Line
Tutorial
Tutorial2
© 2023 Season7 All rights reserved

WebControl - Page.UserProfile

- The Page.UserProfile Control displays user's profile interface webpage.
- A logo, title and so on are customizable.

Code Example

Fig1. Visual Basic

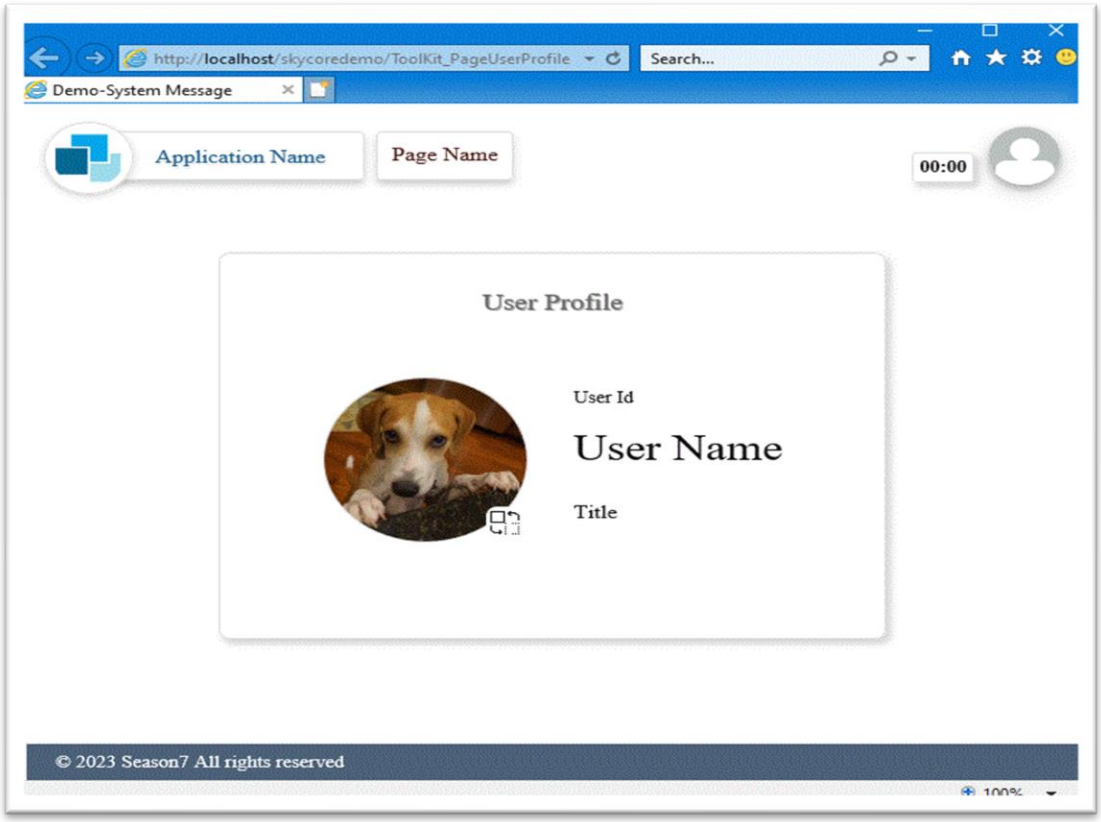
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_PageUserProfile
   Inherits WebPage
   Public Overrides Sub OnInitialized()
      HtmlDoc.SetTitle("User Profile")

      Dim page As New ToolKit.Pages.UserProfile
      HtmlDoc.HtmlBodyText = page.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_PageUserProfile : WebPage
{
        public ToolKit_PageUserProfile()
        {
        }
   public override void OnInitialized()
```

```
  {
    HtmlDoc.SetTitle("User Profile");

    ToolKit.Pages.UserProfile page = new ToolKit.Pages.UserProfile();
    HtmlDoc.HtmlBodyText = page.HtmlText();
  }
}
```

Output



WebControl - Page.Verification

- The Page.Verification Control displays Two Factor Authentication interface webpage.
- A logo, title and so on are customizable.

Code Example

Fig1. Visual Basic

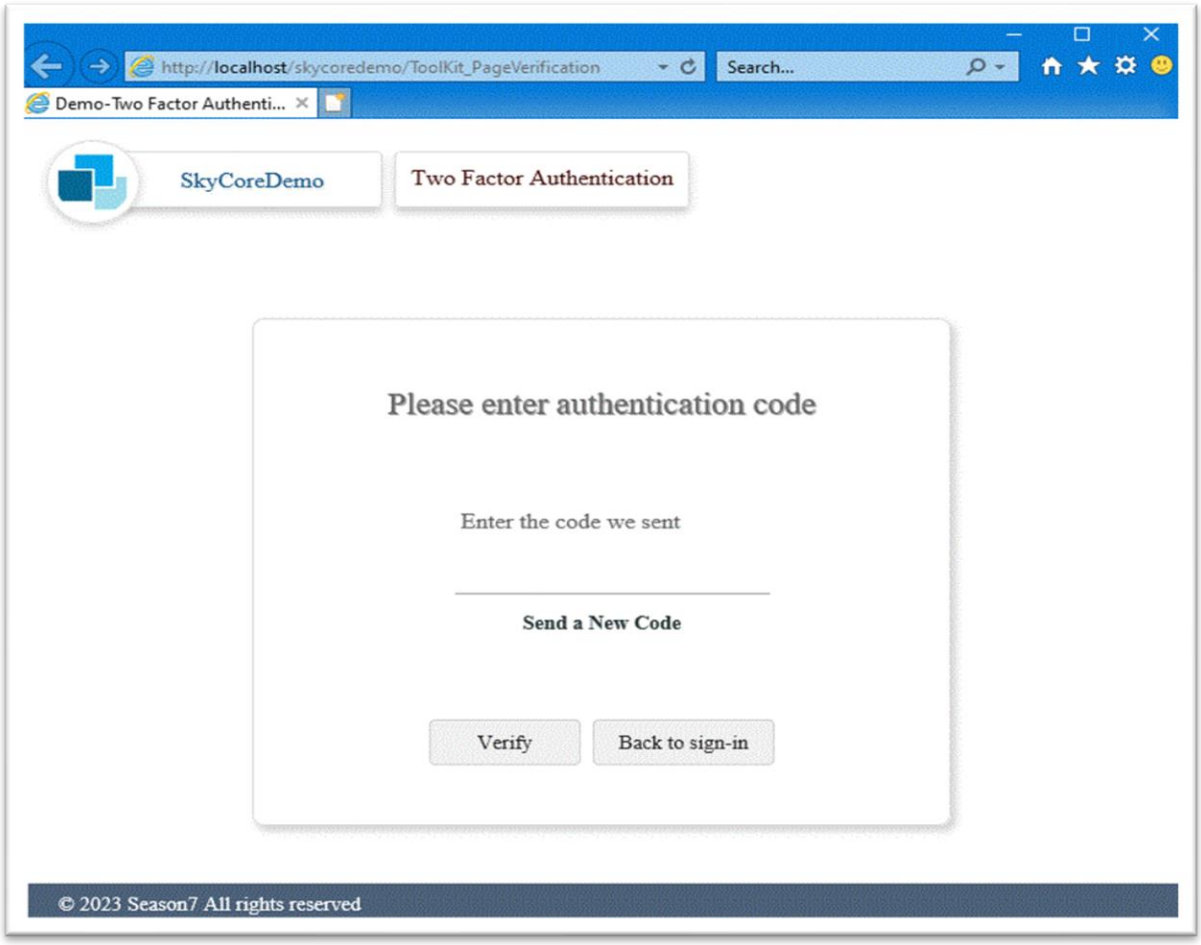
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_PageVerification
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      HtmlDoc.SetTitle("Two Factor Authentication")

      Dim page As New ToolKit.Pages.Verification
      HtmlDoc.HtmlBodyText = page.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_PageVerification : WebPage
{
        public ToolKit_PageVerification()
        {
        }
```

```
  public override void OnInitialized()
  {
    HtmlDoc.SetTitle("Two Factor Authentication");

    ToolKit.Pages.Verification page = new ToolKit.Pages.Verification();
    HtmlDoc.HtmlBodyText = page.HtmlText();
  }
}
```

Output



WebControl - Section.Authentication

- The Section.Authentication Control displays a Two Factor Authentication section on webpage.
- A logo, title and so on are customizable.

Code Example

Fig1. Visual Basic

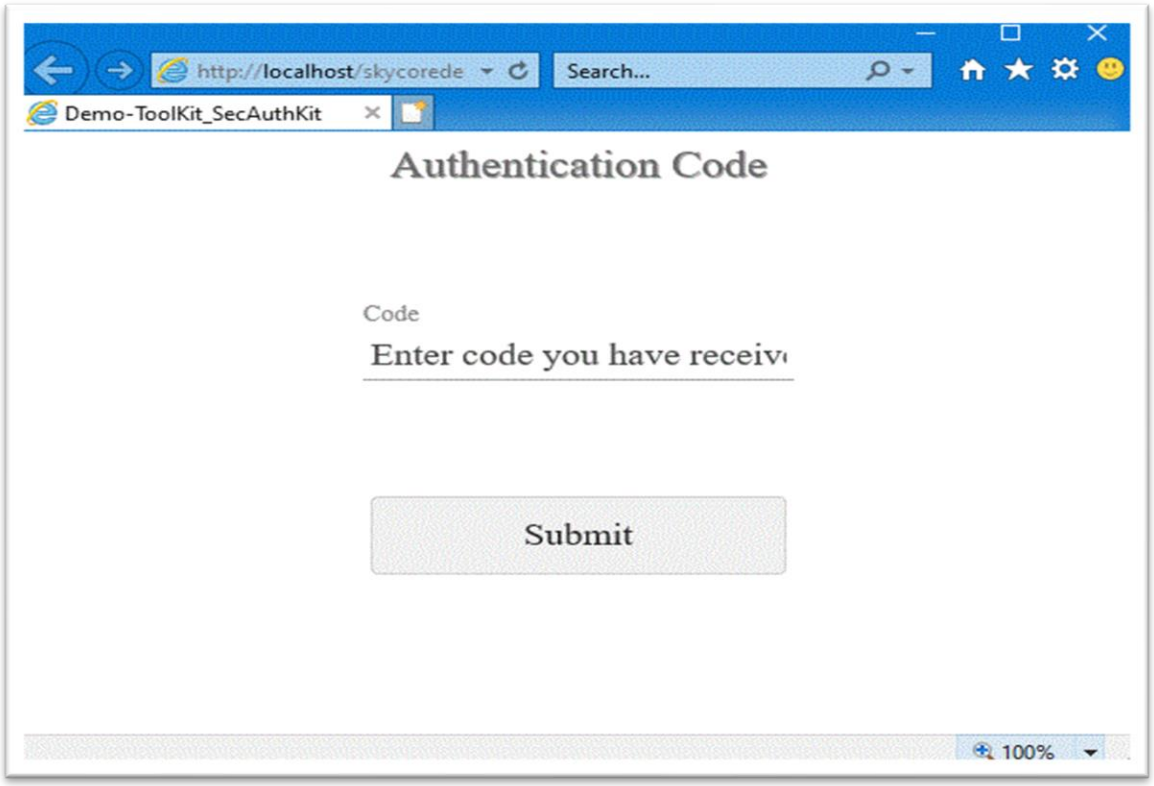
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecAuthKit
   Inherits WebPage

   Public Overrides Sub OnInitialized()

      Dim section As New ToolKit.Sections.Authentication

      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecAuthKit : WebPage
{
        public ToolKit_SecAuthKit()
        {
        }
   public override void OnInitialized()
```

```
  {
    ToolKit.Sections.Authentication section = new ToolKit.Sections.Authentication();

    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Footer

- The Section.Footer Control displays a footer of webpage.
- A title and menu are customizable.

Code Example

Fig1. Visual Basic

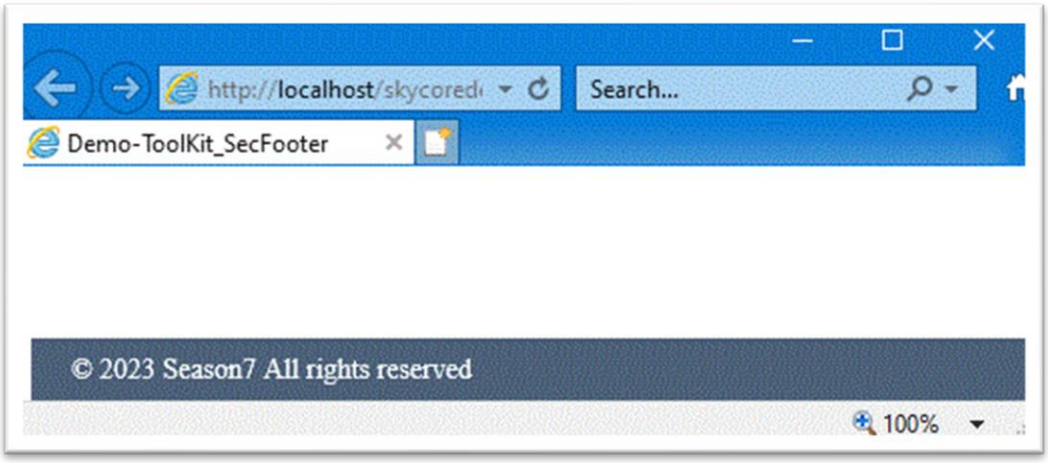
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecFooter
   Inherits WebPage

   Public Overrides Sub OnInitialized()

      Dim section As New ToolKit.Sections.Footer

      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecFooter : WebPage
{
        public ToolKit_SecFooter()
        {
        }
   public override void OnInitialized()
```

```
  {
    ToolKit.Sections.Footer section = new ToolKit.Sections.Footer();

    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Login

- The Section.Login Control displays a Login section.
- A title, label and button properties are customizable.

Code Example

Fig1. Visual Basic

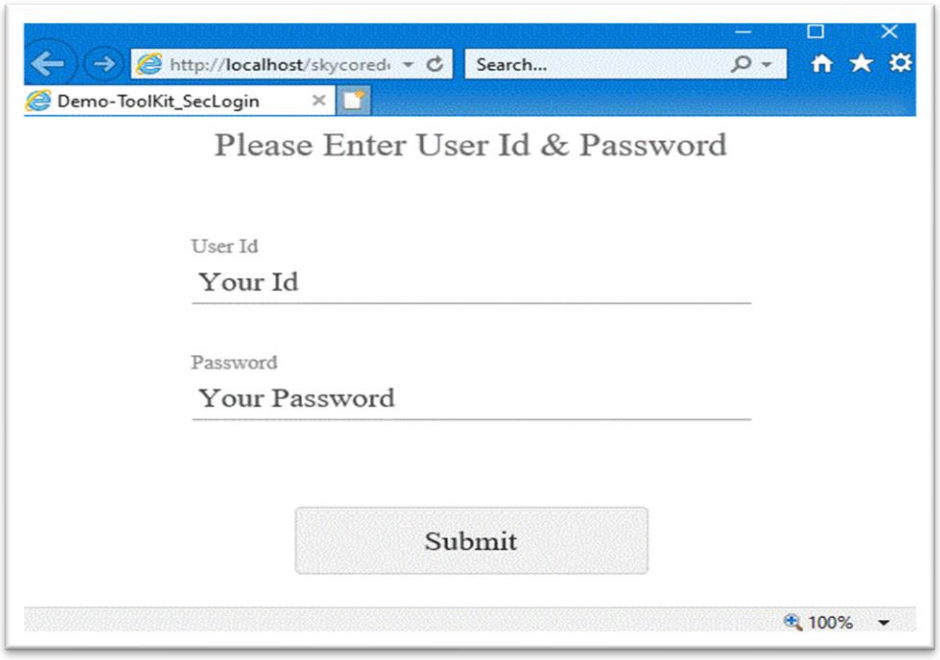
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecLogin
   Inherits WebPage

   Public Overrides Sub OnInitialized()

      Dim section As New ToolKit.Sections.Login

      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecLogin : WebPage
{
        public ToolKit_SecLogin()
        {
        }
```

```
  public override void OnInitialized()
  {
     ToolKit.Sections.Login section = new ToolKit.Sections.Login();

     HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Login2

- The Section.Login2 Control displays a Login section.
- A title, logo, label and button properties are customizable.

Code Example

Fig1. Visual Basic

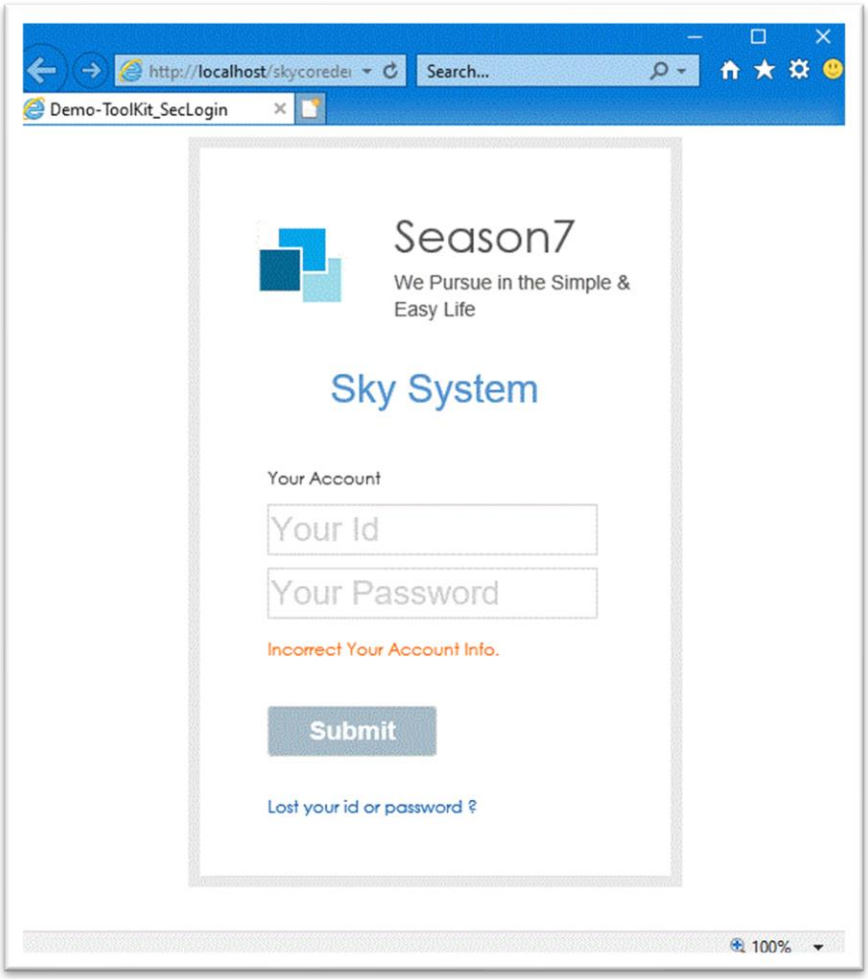
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecLogin
   Inherits WebPage

   Public Overrides Sub OnInitialized()

      Dim section As New ToolKit.Sections.Login2

      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecLogin : WebPage
{
        public ToolKit_SecLogin()
        {
        }
   public override void OnInitialized()
```

```
  {
    ToolKit.Sections.Login2 section = new ToolKit.Sections.Login2();

    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.MenuBox

- The Section.MenuBox Control displays memu list section.
- The menu items are customizable.

Code Example

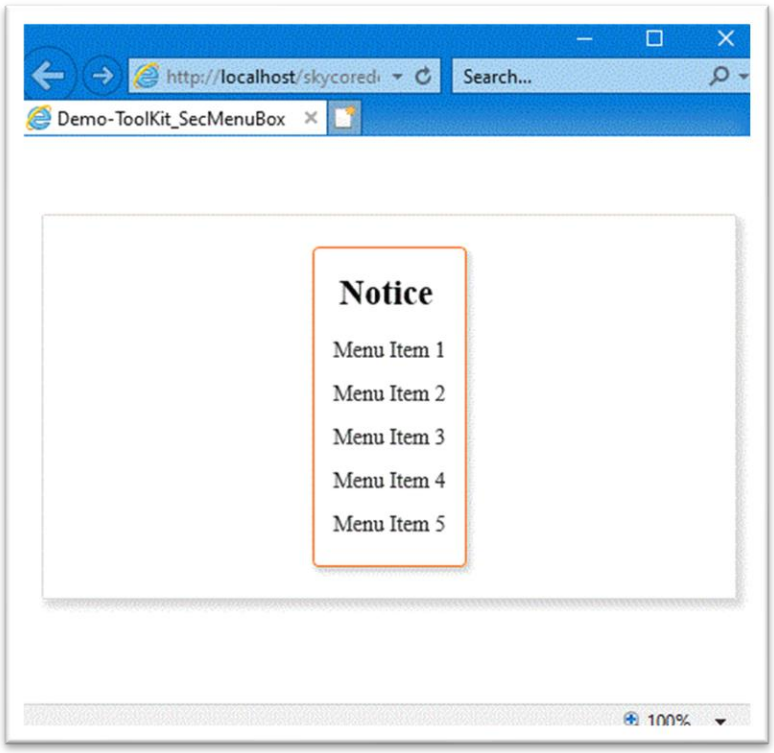
Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore
Public Class ToolKit_SecMenuBox
   Inherits WebPage
   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.MenuBox

      section.Wrap.SetStyle(HtmlStyles.marginTop, "50px")
      section.Wrap.SetStyle(HtmlStyles.marginBottom, "50px")

      Dim mnu As New ToolKit.MenuList
      mnu.SetStyle(HtmlStyles.border, "1px solid #ff6600")
      mnu.Title.InnerText = "Notice"
      mnu.Add("Menu Item 1")
      mnu.Add("Menu Item 2")
      mnu.Add("Menu Item 3")
      mnu.Add("Menu Item 4")
      mnu.Add("Menu Item 5")
      section.MenuBoxes.Add(mnu)

      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecMenuBox : WebPage
{
        public ToolKit_SecMenuBox()
        {
        }
   public override void OnInitialized()
   {
      ToolKit.Sections.MenuBox section = new ToolKit.Sections.MenuBox();
      section.Wrap.SetStyle(HtmlStyles.marginTop, "50px");
      section.Wrap.SetStyle(HtmlStyles.marginBottom, "50px");

      ToolKit.MenuList mnu = new ToolKit.MenuList();
      mnu.SetStyle(HtmlStyles.border, "1px solid #ff6600");
      mnu.Title.InnerText = "Notice";
      mnu.Add("Menu Item 1");
      mnu.Add("Menu Item 2");
      mnu.Add("Menu Item 3");
      mnu.Add("Menu Item 4");
      mnu.Add("Menu Item 5");
      section.MenuBoxes.Add(mnu);

      HtmlDoc.HtmlBodyText = section.HtmlText();
   }
}
```

Output


http://localhost/skycored
Search...
Demo-ToolKit_SecMenuBox
Notice
Menu Item 1
Menu Item 2
Menu Item 3
Menu Item 4
Menu Item 5
100%

WebControl - Section.Message

- The Section.Message Control displays a system message in a section.
- A message is customizable.

Code Example

Fig1. Visual Basic

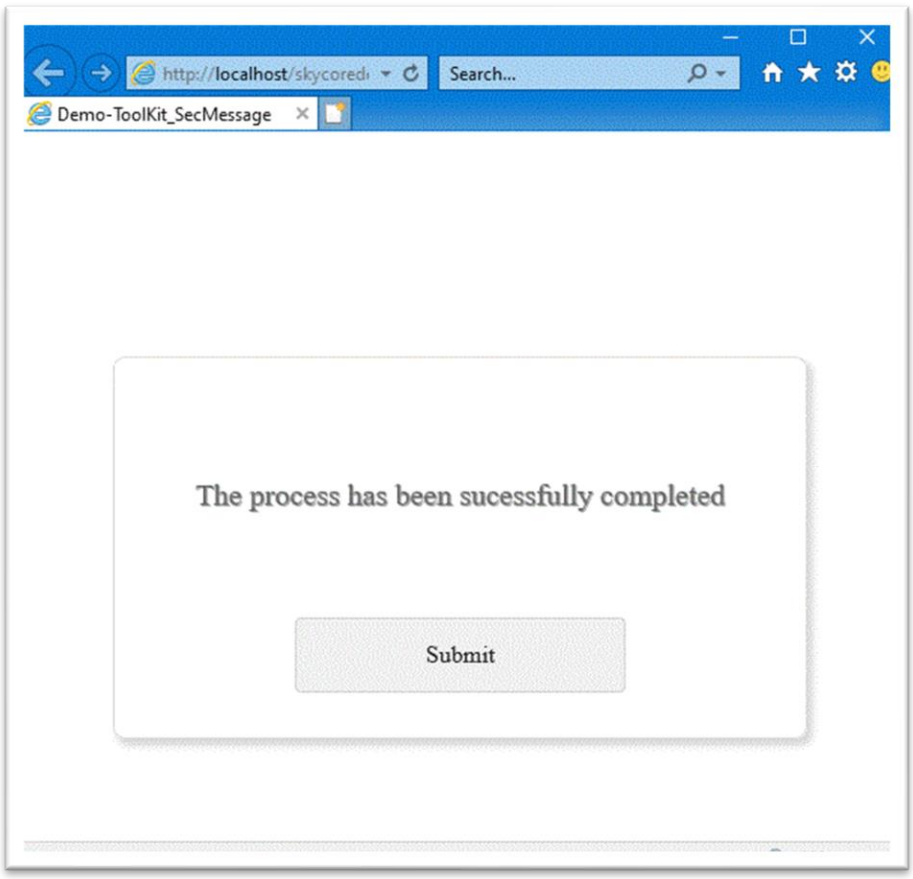
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecMessage
   Inherits WebPage

   Public Overrides Sub OnInitialized()

      Dim section As New ToolKit.Sections.Message
      section.Message.InnerText = "The process has been sucessfully completed"

      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecMessage : WebPage
{
        public ToolKit_SecMessage()
        {
        }
```

```
  public override void OnInitialized()
  {
     ToolKit.Sections.Message section = new ToolKit.Sections.Message();
     section.Message.InnerText = "The process has been sucessfully completed";

     HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.TextPanel

- The Section.TextPanel Control displays text items with a title in a section.
- Textitem and buttons are customizable.

Code Example

Fig1. Visual Basic

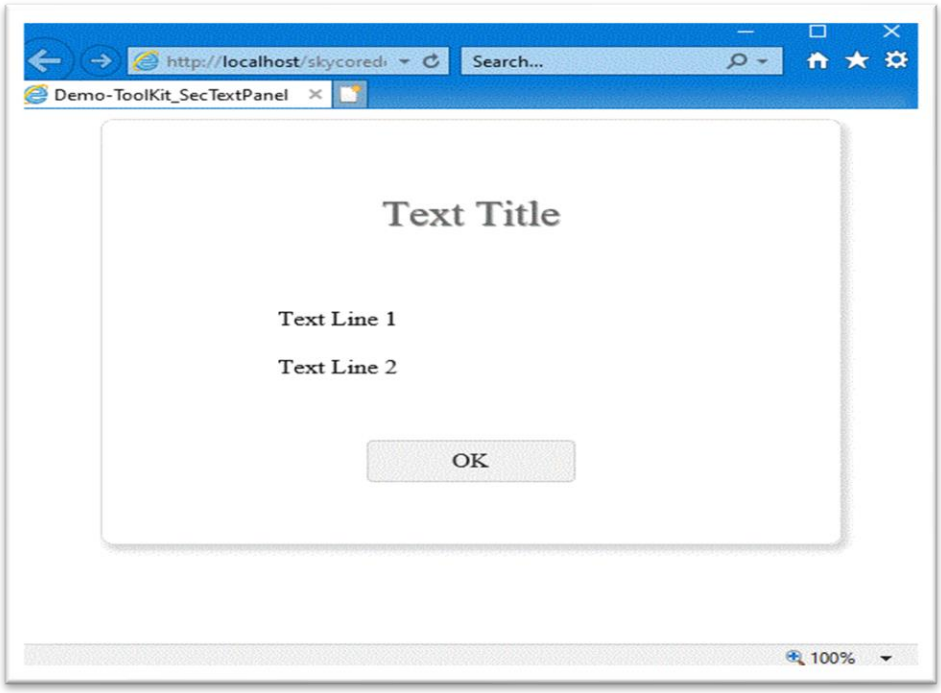
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecTextPanel
   Inherits WebPage

   Public Overrides Sub OnInitialized()

      Dim section As New ToolKit.Sections.TextPanel
      section.PanelTitle.InnerText = "Text Title"
      section.AddTextItem("Text Line 1")
      section.AddTextItem("Text Line 2")
      section.AddButton("OK")

      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecTextPanel : WebPage
```

```
{
        public ToolKit_SecTextPanel()
        {
        }
  public override void OnInitialized()
  {
    ToolKit.Sections.TextPanel section = new ToolKit.Sections.TextPanel();
    section.PanelTitle.InnerText = "Text Title";
    section.AddTextItem("Text Line 1");
    section.AddTextItem("Text Line 2");
    section.AddButton("OK");

    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.TextInputs

- The Section.TextInputs Control displays text inputs with a title in a section.
- Textitem and buttons are customizable.

Code Example

Fig1. Visual Basic

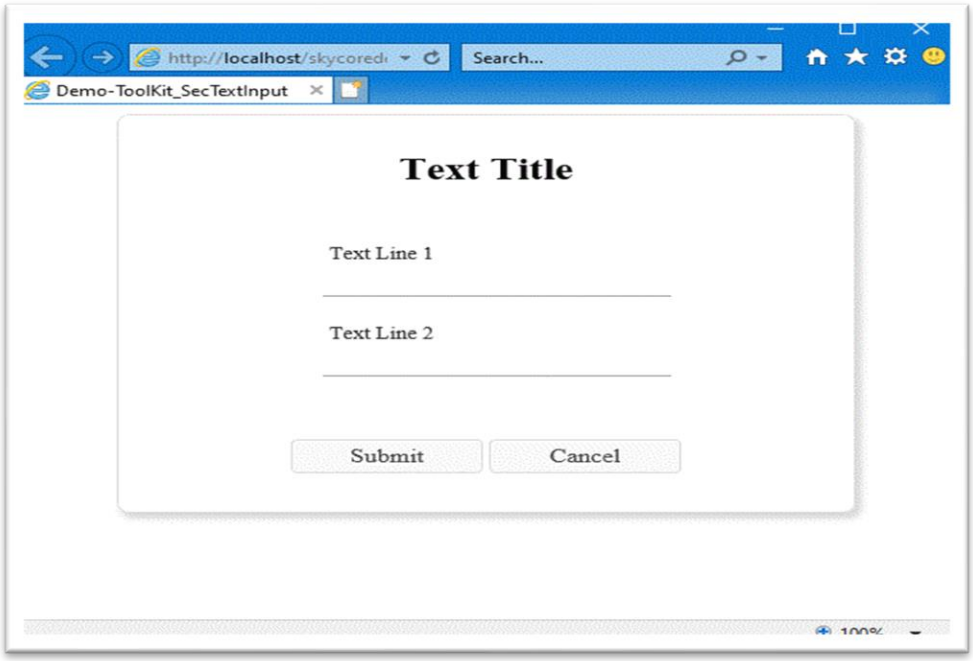
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecTextInput
   Inherits WebPage

   Public Overrides Sub OnInitialized()

      Dim section As New ToolKit.Sections.TextInputs
      section.Title.InnerText = "Text Title"
      section.AddTextBox("Text Line 1")
      section.AddTextBox("Text Line 2")
      section.AddButton("Submit")
      section.AddButton("Cancel")

      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;
```

```
public class ToolKit_SecTextInput : WebPage
{
        public ToolKit_SecTextInput()
        {
        }
  public override void OnInitialized()
  {
    ToolKit.Sections.TextInputs section = new ToolKit.Sections.TextInputs();
    section.Title.InnerText = "Text Title";
    section.AddTextBox("Text Line 1");
    section.AddTextBox("Text Line 2");
    section.AddButton("Submit");
    section.AddButton("Cancel");

    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.TitleBar

- The Section.TitleBar Control displays a title bar.
- Logo and menu are customizable.

Code Example

Fig1. Visual Basic

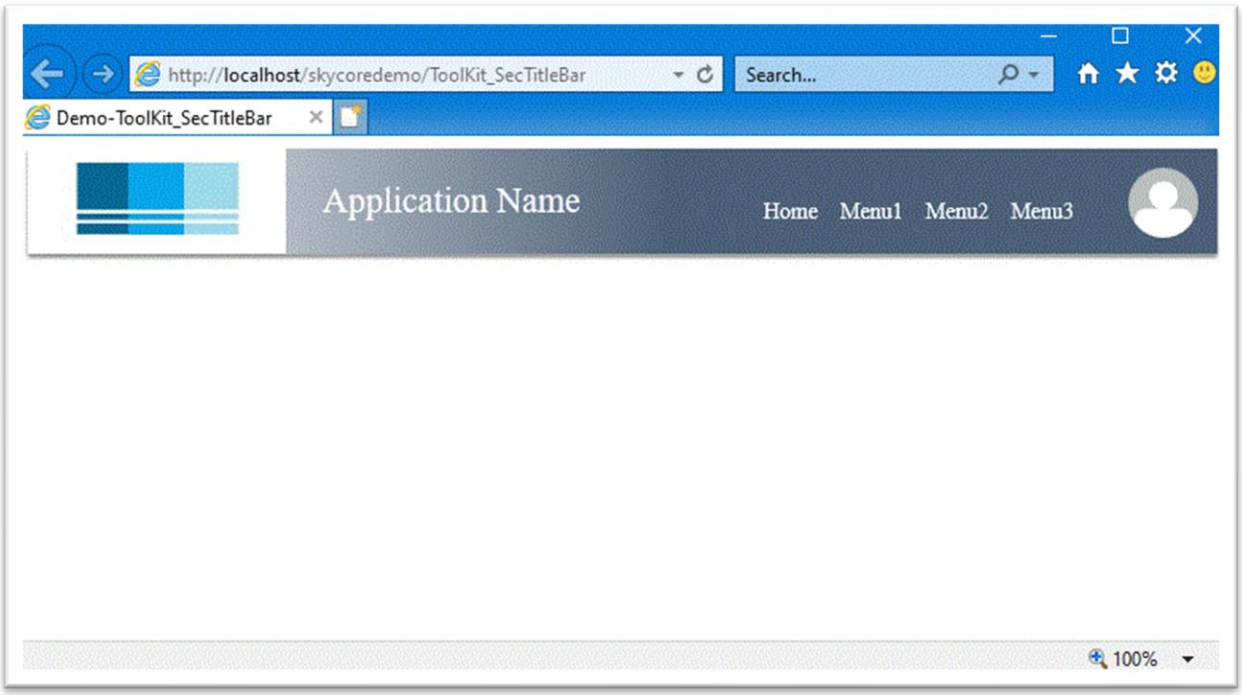
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecTitleBar
   Inherits WebPage

   Public Overrides Sub OnInitialized()

      Dim section As New ToolKit.Sections.TitleBar
      section.SetTitleMenu("Home")
      section.SetTitleMenu("Menu1")
      section.SetTitleMenu("Menu2")
      section.SetTitleMenu("Menu3")

      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecTitleBar : WebPage
```

```
{
        public ToolKit_SecTitleBar()
        {
        }
  public override void OnInitialized()
  {
    ToolKit.Sections.TitleBar section = new ToolKit.Sections.TitleBar();
    section.SetTitleMenu("Home");
    section.SetTitleMenu("Menu1");
    section.SetTitleMenu("Menu2");
    section.SetTitleMenu("Menu3");

    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.TitleBar2

- The Section.TitleBar2 Control displays a title bar.
- Logo and menu are customizable.

Code Example

Fig1. Visual Basic

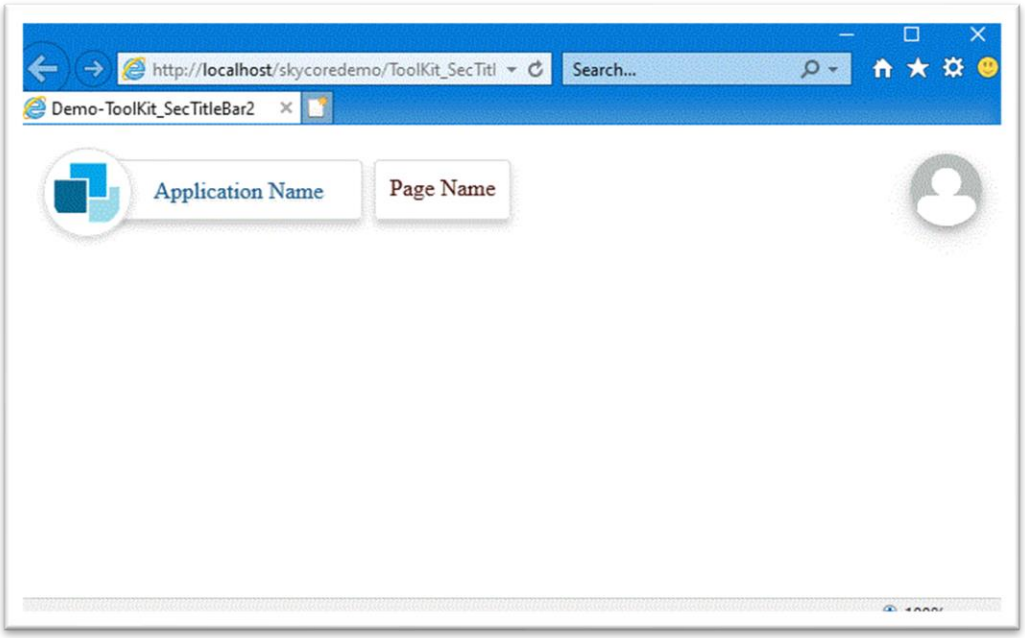
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecTitleBar2
   Inherits WebPage
   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.TitleBar2
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecTitleBar2 : WebPage
{
        public ToolKit_SecTitleBar2()
        {
        }
   public override void OnInitialized()
   {
      ToolKit.Sections.TitleBar2 section = new ToolKit.Sections.TitleBar2();
      HtmlDoc.HtmlBodyText = section.HtmlText();
```

```
    }
}
```

Output


http://localhost/skycoredemo/ToolKit_SecTitl
Search...
Demo-ToolKit_SecTitleBar2
Application Name
Page Name

WebControl - Section.TitleBar3

- The Section.TitleBar3 Control displays a title bar.
- Logo and menu are customizable.

Code Example

Fig1. Visual Basic

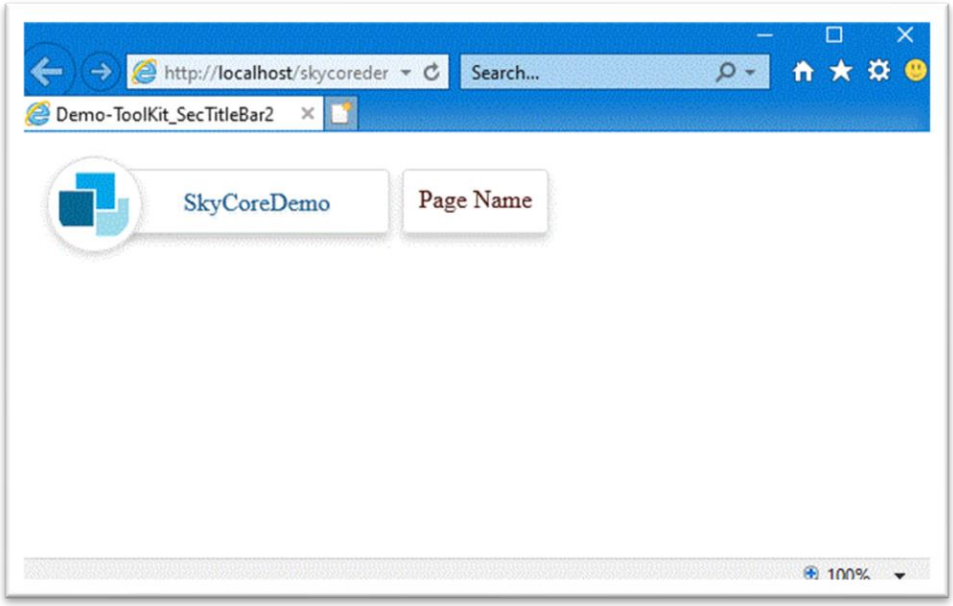
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecTitleBar3
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.TitleBar3
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecTitleBar3 : WebPage
{
        public ToolKit_SecTitleBar3()
        {
        }
   public override void OnInitialized()
   {
```

```
    ToolKit.Sections.TitleBar3 section = new ToolKit.Sections.TitleBar3();

    HtmlDoc.HtmlBodyText = section.HtmlText();

  }

}
```

Output



WebControl - Section.UserProfile

- The Section.UserProfile Control displays a user profile section.
- Photo and description items are customizable.

Code Example

Fig1. Visual Basic

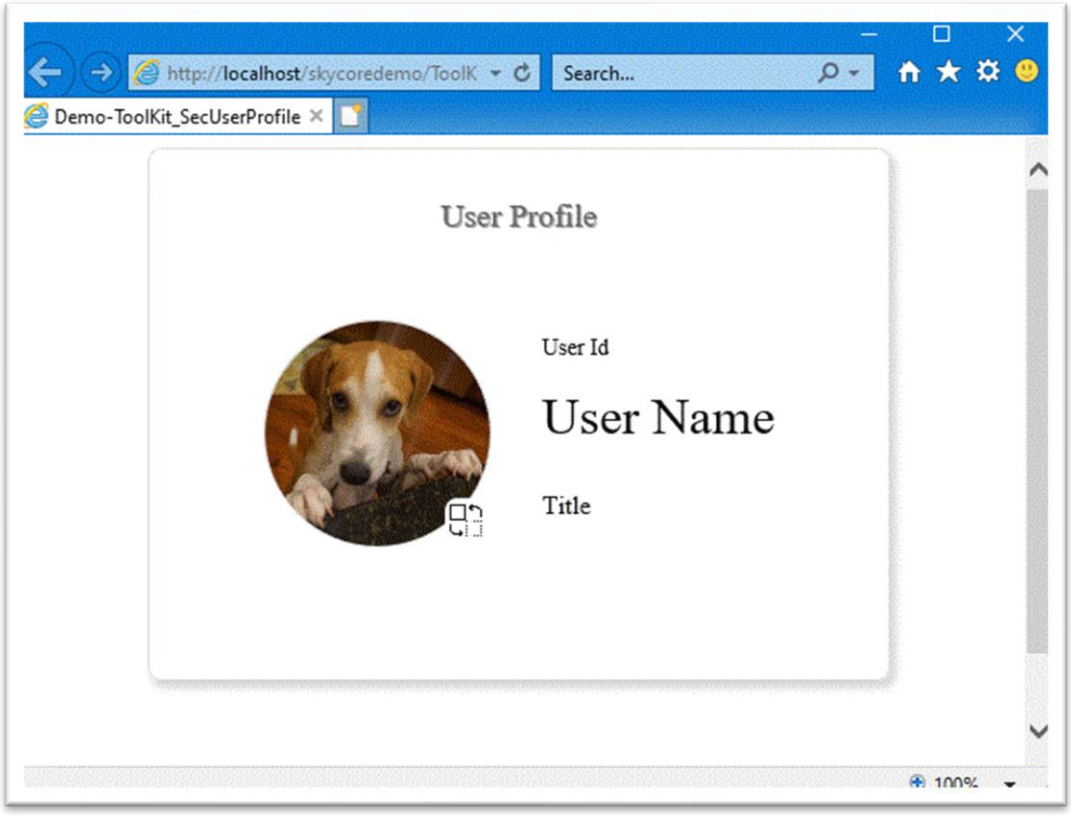
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecUserProfile
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.UserProfile
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecUserProfile : WebPage
{
        public ToolKit_SecUserProfile()
        {
        }
   public override void OnInitialized()
   {
```

```
    ToolKit.Sections.UserProfile section = new ToolKit.Sections.UserProfile();

    HtmlDoc.HtmlBodyText = section.HtmlText();

  }

}
```

Output



WebControl - Section.Filter.Date

- The Section.Filter.Date Control displays a date filter in the section.

Code Example

Fig1. Visual Basic

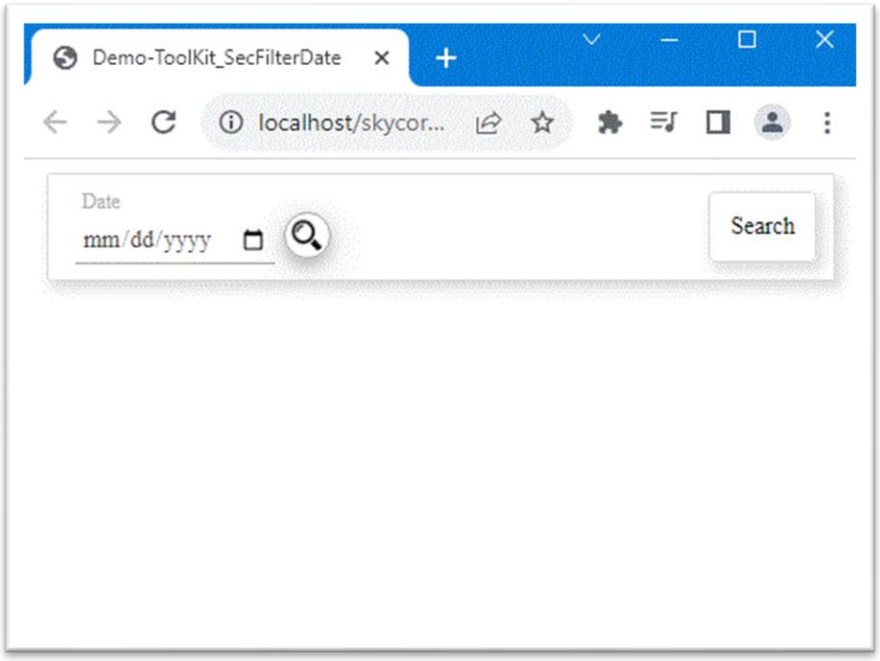
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecFilterDate
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.Date
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecFilterDate : WebPage
{
        public ToolKit_SecFilterDate()
        {
        }
   public override void OnInitialized()
   {
```

```
    ToolKit.Sections.Filter.Date section = new ToolKit.Sections.Filter.Date();
    section.Buttons.Add("Search");
    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Filter.DateToDate

- The Section.Filter.DateToDate Control displays a date filter in the section.

Code Example

Fig1. Visual Basic

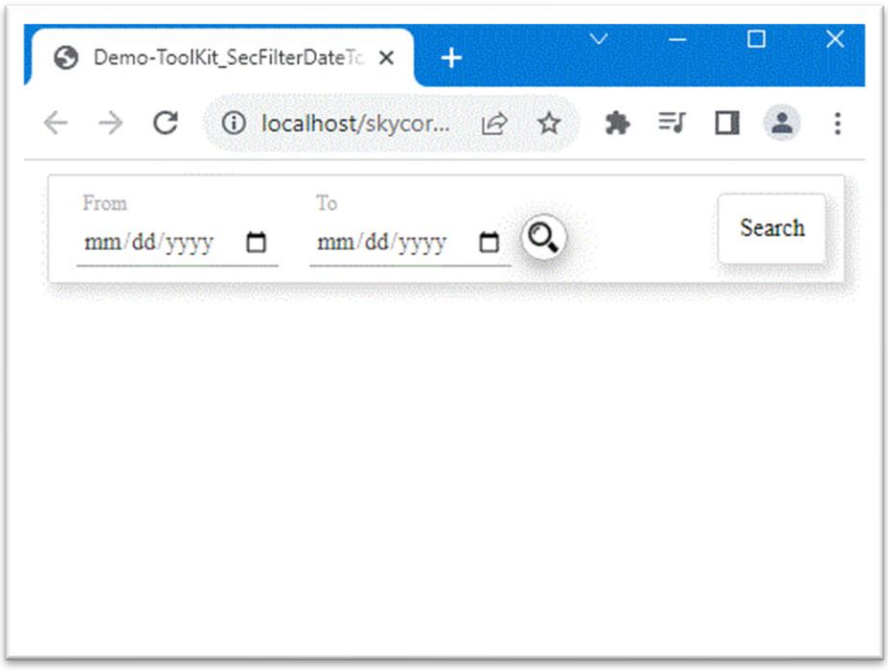
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecFilterDateToDate
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.DateToDate
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecFilterDateToDate : WebPage
{
        public ToolKit_SecFilterDateToDate()
        {
        }
   public override void OnInitialized()
```

```
  {
    ToolKit.Sections.Filter.DateToDate section = new ToolKit.Sections.Filter.DateToDate();
    section.Buttons.Add("Search");
    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Filter.DblDropDown

- The Section.Filter.DblDropDown Control displays 2 dropdown filters in the section.

Code Example

Fig1. Visual Basic

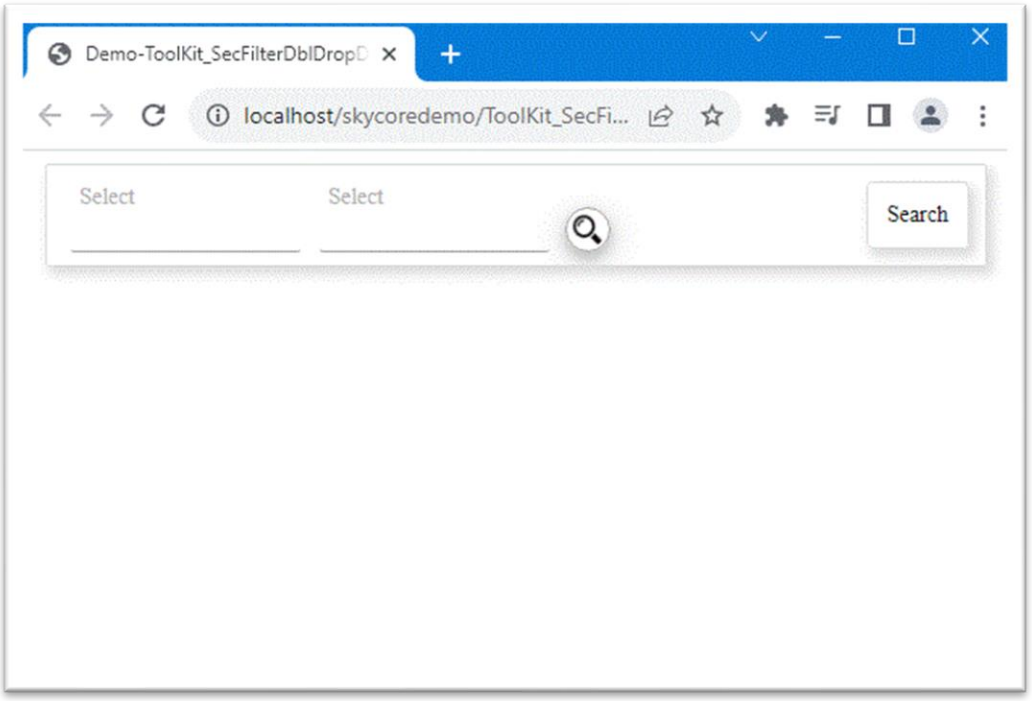
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecFilterDblDropDown
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.DblDropDown
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecFilterDblDropDown : WebPage
{
        public ToolKit_SecFilterDblDropDown()
        {
        }
   public override void OnInitialized()
   {
```

```
    ToolKit.Sections.Filter.DblDropDown section = new ToolKit.Sections.Filter.DblDropDown();
    section.Buttons.Add("Search");
    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Filter.DropDown

- The Section.Filter.DropDown Control displays a dropdown filter in the section.

Code Example

Fig1. Visual Basic

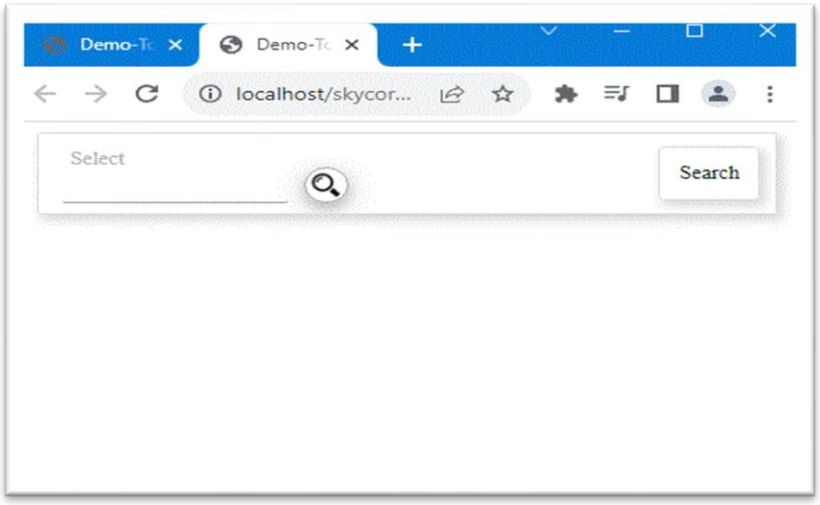
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecFilterDropDown
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.DropDown
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecFilterDropDown : WebPage
{
   public ToolKit_SecFilterDropDown()
   {
   }
   public override void OnInitialized()
```

```
  {
    ToolKit.Sections.Filter.DropDown section = new ToolKit.Sections.Filter.DropDown();
    section.Buttons.Add("Search");
    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Filter.Month

- The Section.Filter.Month Control displays a Month filter in the section.

Code Example

Fig1. Visual Basic

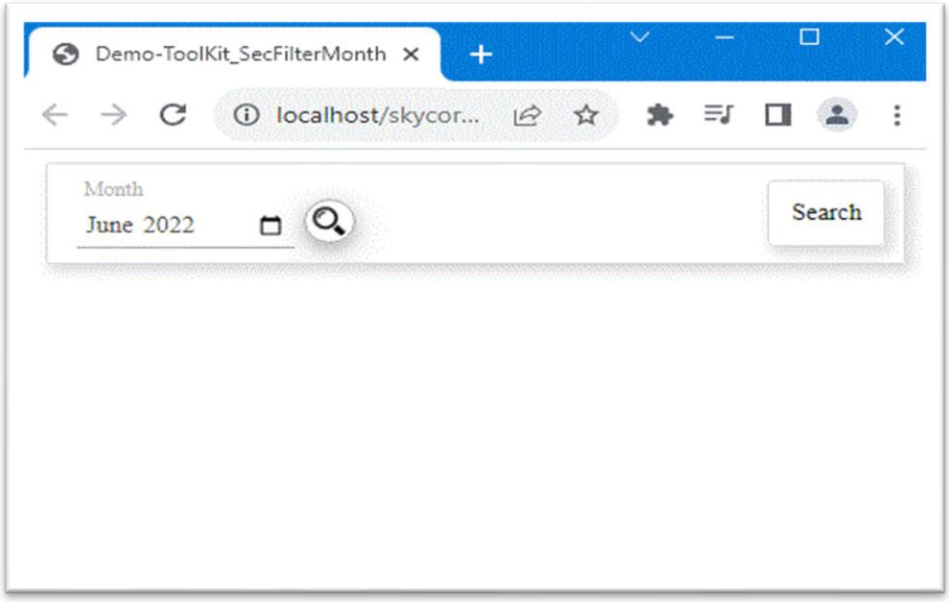
```
Imports Microsoft.VisualBasic
Imports skycore
Public Class ToolKit_SecFilterMonth
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.Month
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecFilterMonth : WebPage
{
            public ToolKit_SecFilterMonth()
            {
   }
   public override void OnInitialized()
   {
      ToolKit.Sections.Filter.Month section = new ToolKit.Sections.Filter.Month();
```

```
    section.Buttons.Add("Search");

    HtmlDoc.HtmlBodyText = section.HtmlText();

  }

}
```

Output



WebControl - Section.Filter.MonthToMonth

- The Section.Filter.MonthToMonth Control displays 2 Month filters in the section.

Code Example

Fig1. Visual Basic

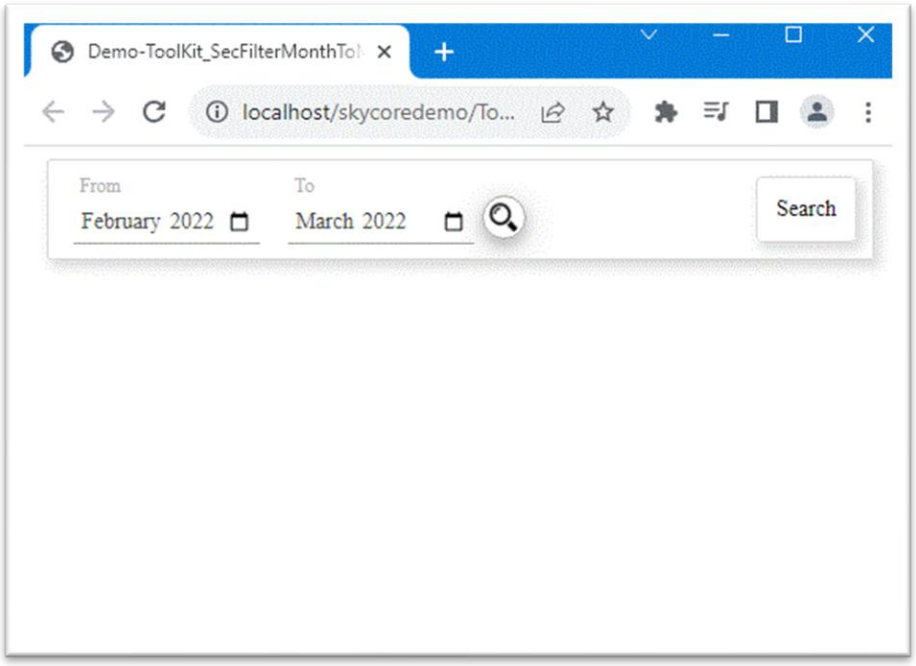
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecFilterMonthToMonth
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.MonthToMonth
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecFilterMonthToMonth : WebPage
{
        public ToolKit_SecFilterMonthToMonth()
        {
   }
   public override void OnInitialized()
```

```
  {
    ToolKit.Sections.Filter.MonthToMonth section = new ToolKit.Sections.Filter.MonthToMonth();
    section.Buttons.Add("Search");
    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Filter.Title

- The Section.Filter.Title Control displays a Title Label in the section.

Code Example

Fig1. Visual Basic

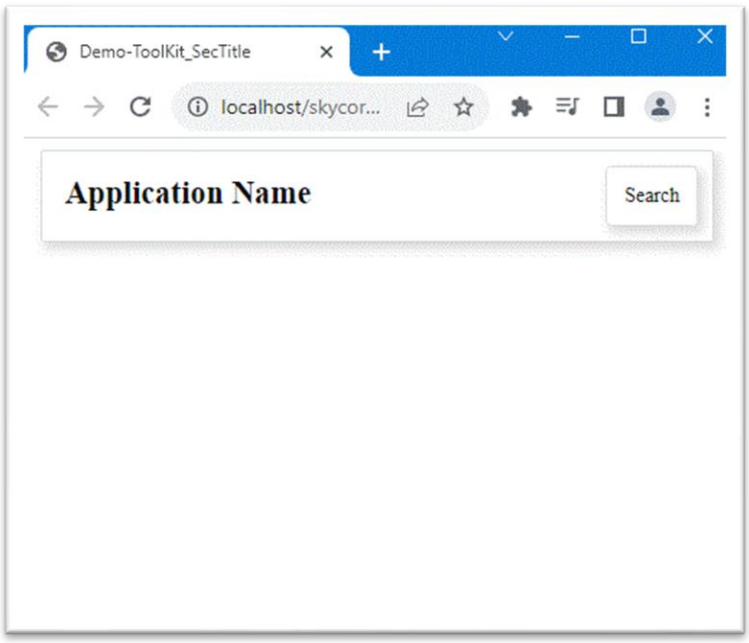
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecTitle
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.Title
      section.Filter.InnerText = "Application Name"
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecTitle : WebPage
{
        public ToolKit_SecTitle()
        {
   }
```

```
  public override void OnInitialized()
  {
     ToolKit.Sections.Filter.Title section = new ToolKit.Sections.Filter.Title();
     section.Filter.InnerText = "Application Name";
     section.Buttons.Add("Search");
     HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Filter.Text

- The Section.Filter.Text Control displays a Text Input in the section.

Code Example

Fig1. Visual Basic

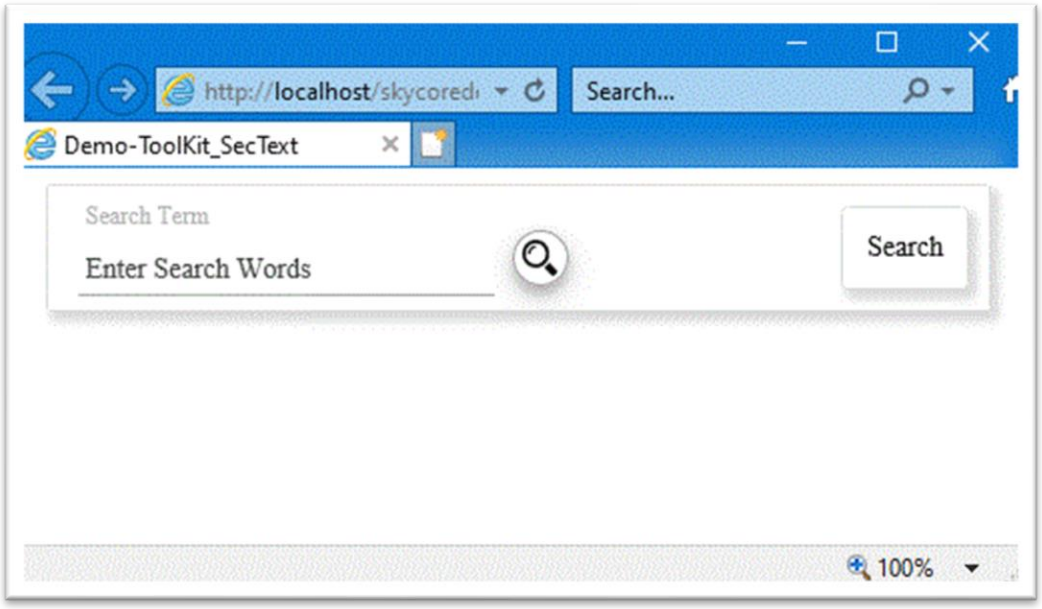
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecText
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.Text
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecText : WebPage
{
        public ToolKit_SecText()
        {
   }
   public override void OnInitialized()
```

```
  {
    ToolKit.Sections.Filter.Text section = new ToolKit.Sections.Filter.Text();
    section.Buttons.Add("Search");
    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Filter.TextCheck

- The Section.Filter.TextCheck Control displays a Text Input and checkbox in the section.

Code Example

Fig1. Visual Basic

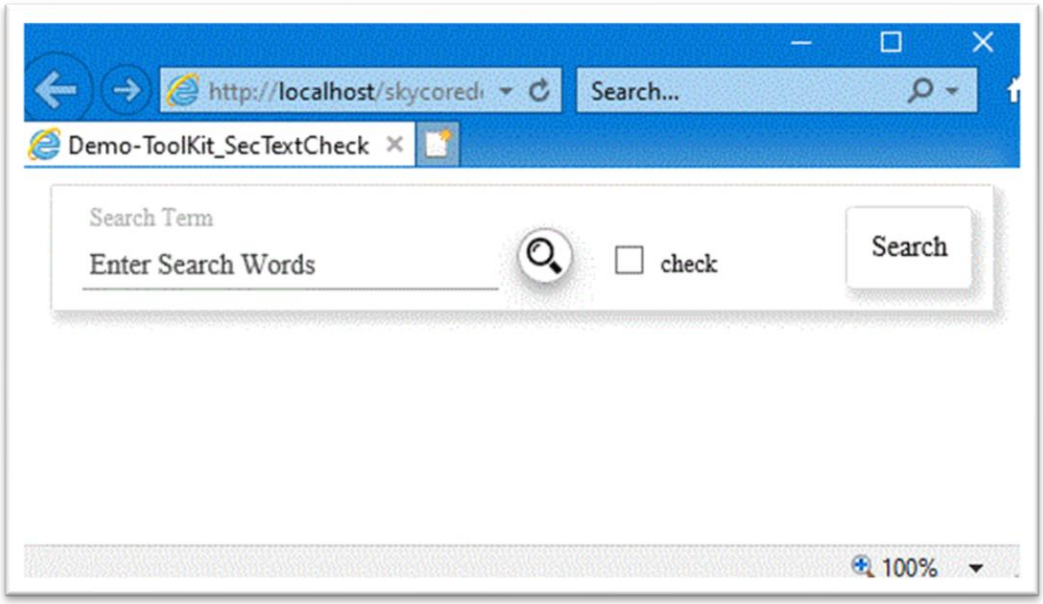
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecTextCheck
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.TextCheck
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecTextCheck : WebPage
{
        public ToolKit_SecTextCheck()
        {
   }
   public override void OnInitialized()
   {
```

```
    ToolKit.Sections.Filter.TextCheck section = new ToolKit.Sections.Filter.TextCheck();
    section.Buttons.Add("Search");
    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Filter.TextDate

- The Section.Filter.TextDate Control displays a Text Input and date in the section.

Code Example

Fig1. Visual Basic

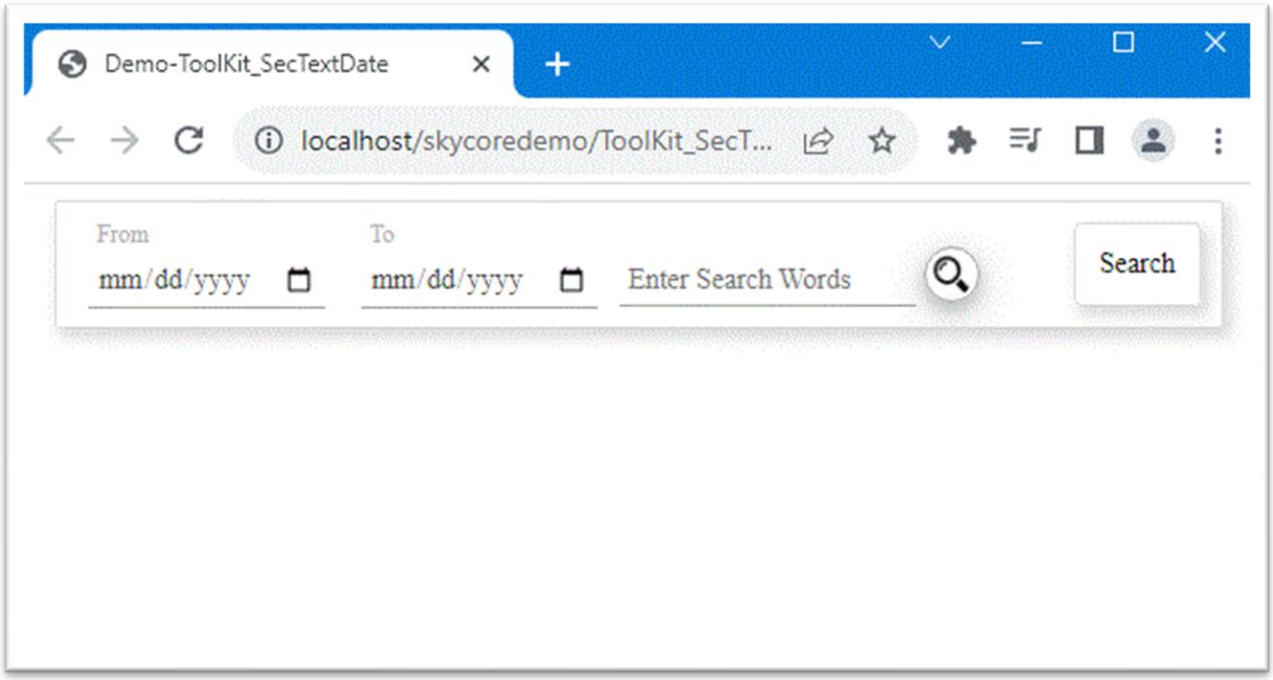
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecTextDate
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.TextDate
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecTextDate : WebPage
{
        public ToolKit_SecTextDate()
        {
   }
   public override void OnInitialized()
   {
```

```
    ToolKit.Sections.Filter.TextDate section = new ToolKit.Sections.Filter.TextDate();
    section.Buttons.Add("Search");
    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Section.Filter.TextMonth

- The Section.Filter.TextMonth Control displays a Text Input and month in the section.

Code Example

Fig1. Visual Basic

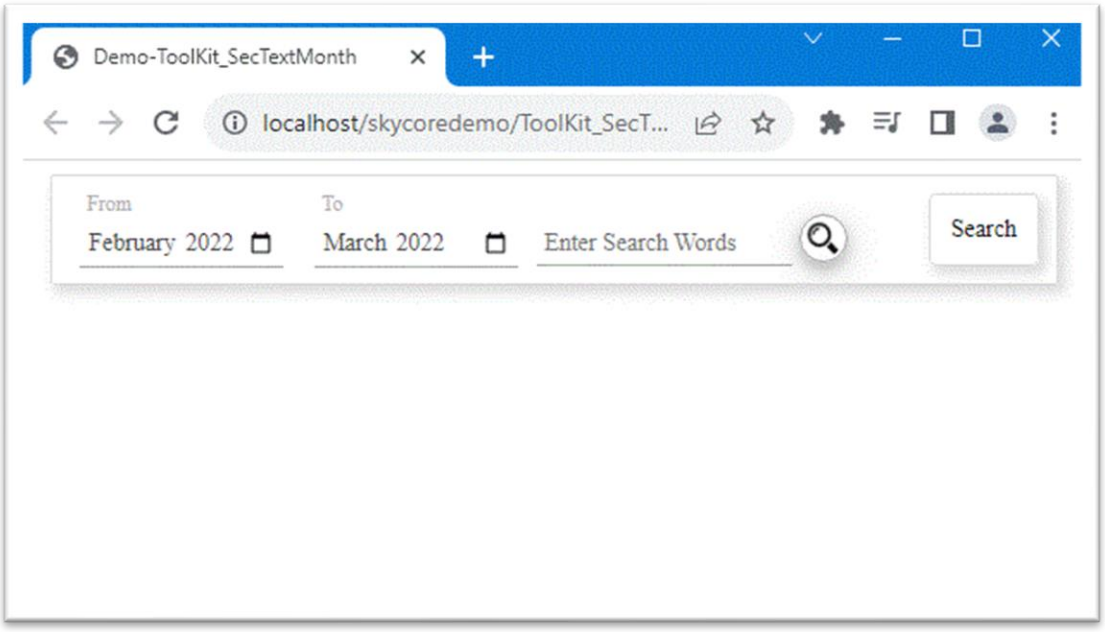
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_SecTextMonth
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim section As New ToolKit.Sections.Filter.TextMonth
      section.Buttons.Add("Search")
      HtmlDoc.HtmlBodyText = section.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_SecTextMonth : WebPage
{
        public ToolKit_SecTextMonth()
        {
   }
   public override void OnInitialized()
```

```
  {
    ToolKit.Sections.Filter.TextMonth section = new ToolKit.Sections.Filter.TextMonth();
    section.Buttons.Add("Search");
    HtmlDoc.HtmlBodyText = section.HtmlText();
  }
}
```

Output



WebControl - Inputs.Checkbox

- The Inputs.Checkbox Control displays a checkbox element.

Code Example

Fig1. Visual Basic

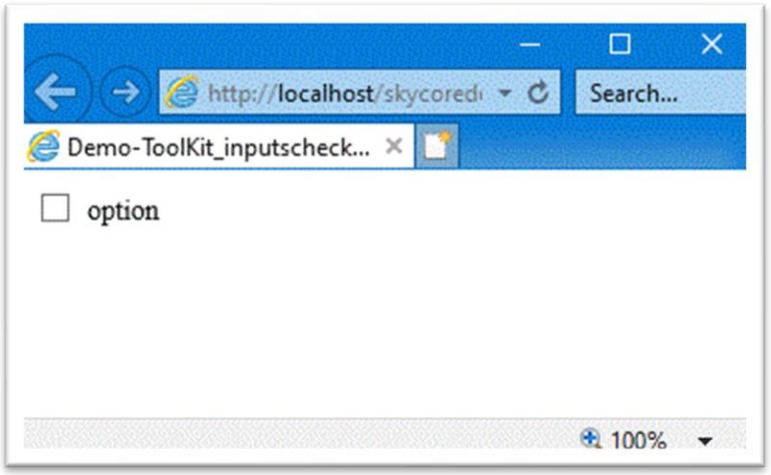
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_inputscheckbox
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Checkbox
      input.Label.InnerText = "option"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_inputscheckbox : WebPage
{
        public ToolKit_inputscheckbox()
        {
   }
   public override void OnInitialized()
```

```
  {
    ToolKit.Inputs.Checkbox input = new ToolKit.Inputs.Checkbox();
    input.Label.InnerText = "option";
    HtmlDoc.HtmlBodyText = input.HtmlText();
  }
}
```

Output



WebControl - Inputs.Color

- The Inputs.Color Control displays a color picker element.

Code Example

Fig1. Visual Basic

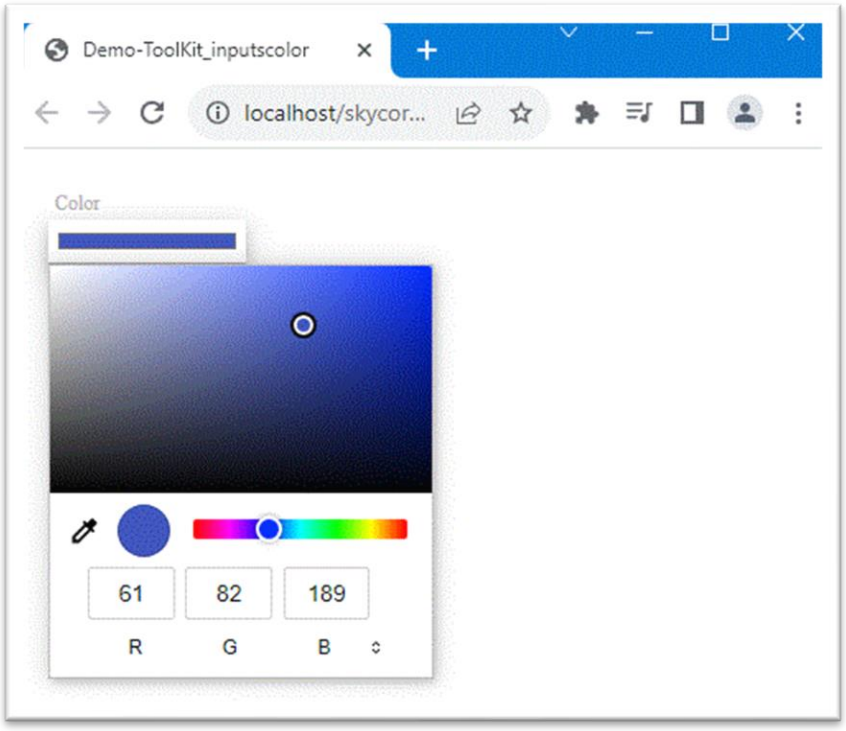
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_inputscolor
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Color
      input.Label.InnerText = "Color"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_inputscolor : WebPage
{
        public ToolKit_inputscolor()
        {
   }
   public override void OnInitialized()
```

```
  {
    ToolKit.Inputs.Checkbox input = new ToolKit.Inputs.Checkbox();
    input.Label.InnerText = "Color";
    HtmlDoc.HtmlBodyText = input.HtmlText();
  }
}
```

Output



WebControl - Inputs.Date

- The Inputs.Date Control displays a Date input element.

Code Example

Fig1. Visual Basic

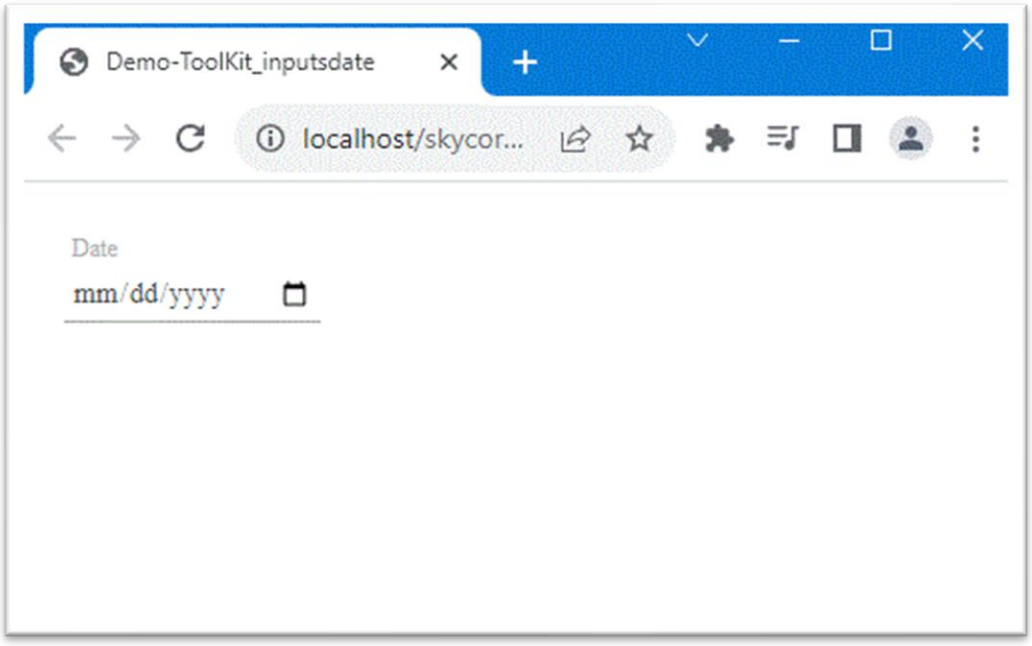
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_inputsdate
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Date
      input.Label.InnerText = "Date"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_inputsdate : WebPage
{
        public ToolKit_inputsdate()
        {
   }
   public override void OnInitialized()
```

```
  {
    ToolKit.Inputs.Date input = new ToolKit.Inputs.Date();
    input.Label.InnerText = "Date";
    HtmlDoc.HtmlBodyText = input.HtmlText();
  }
}
```

Output



WebControl - Inputs.Dropdown

- The Inputs.Dropdown Control displays a dropdown element.

Code Example

Fig1. Visual Basic

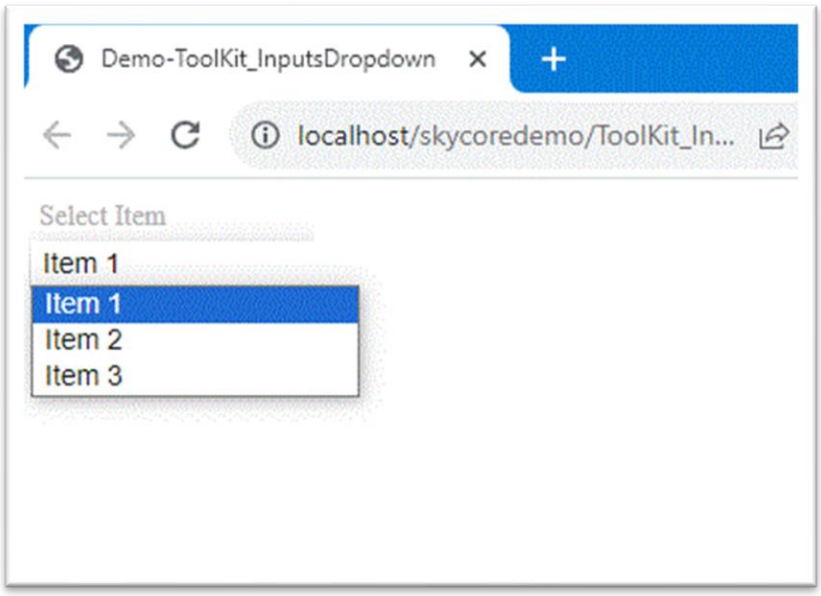
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsDropdown
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Dropdown
      input.Label.InnerText = "Select Item"
      input.SelOptions.Add(New ToolKit.Inputs.Dropdown.SelOption With {.InnerText = "Item 1"})
      input.SelOptions.Add(New ToolKit.Inputs.Dropdown.SelOption With {.InnerText = "Item 2"})
      input.SelOptions.Add(New ToolKit.Inputs.Dropdown.SelOption With {.InnerText = "Item 3"})
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsDropdown : WebPage
{
        public ToolKit_InputsDropdown()
```

```
        {
    }
    public override void OnInitialized()
    {
        ToolKit.Inputs.Dropdown input = new ToolKit.Inputs.Dropdown();
        input.Label.InnerText = "Select Item";
        input.SelOptions.Add(new ToolKit.Inputs.Dropdown.SelOption() { InnerText = "Item 1" });
        input.SelOptions.Add(new ToolKit.Inputs.Dropdown.SelOption() { InnerText = "Item 2" });
        input.SelOptions.Add(new ToolKit.Inputs.Dropdown.SelOption() { InnerText = "Item 3" });
        HtmlDoc.HtmlBodyText = input.HtmlText();
    }
}
```

Output



WebControl - Inputs.Hidden

- The Inputs.Hidden Control creates a hidden input element.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsHidden
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Hidden
      input.SetAttribute(HtmlAttributes.id, "userpwd")
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsHidden : WebPage
{
        public ToolKit_InputsHidden()
```

```
        {
    }
    public override void OnInitialized()
    {
        ToolKit.Inputs.Hidden input = new ToolKit.Inputs.Hidden();
        input.SetAttribute(HtmlAttributes.id, "userpwd");
        HtmlDoc.HtmlBodyText = input.HtmlText();
    }
}
```

Output

WebControl - Inputs.Month

- The Inputs.Month Control creates a Month input element.

Code Example

Fig1. Visual Basic

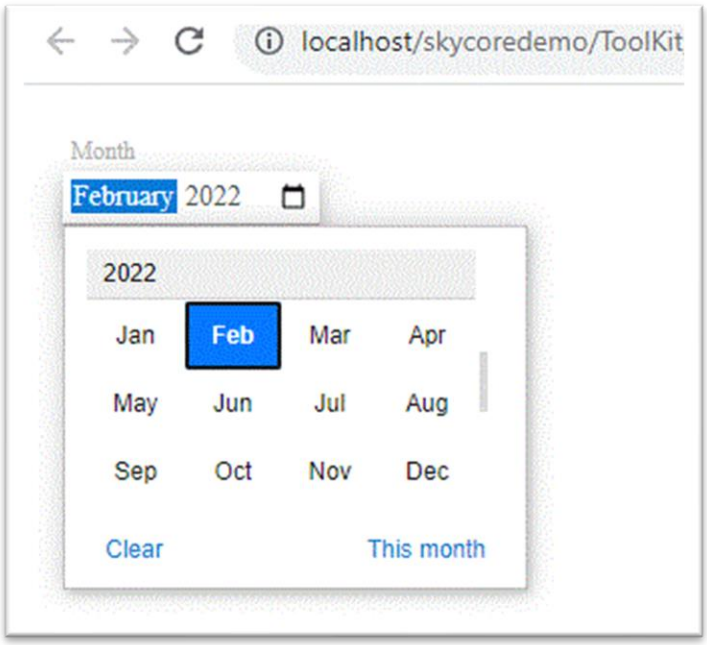
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsMonth
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Month
      input.Label.InnerText = "Month"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsMonth : WebPage
{
        public ToolKit_InputsMonth()
```

```
        {
    }
    public override void OnInitialized()
    {
        ToolKit.Inputs.Month input = new ToolKit.Inputs.Month();
        input.Label.InnerText = "Month";
        HtmlDoc.HtmlBodyText = input.HtmlText();
    }
}
```

Output



WebControl - Inputs.Number

- The Inputs.Number Control creates a Number input element.

Code Example

Fig1. Visual Basic

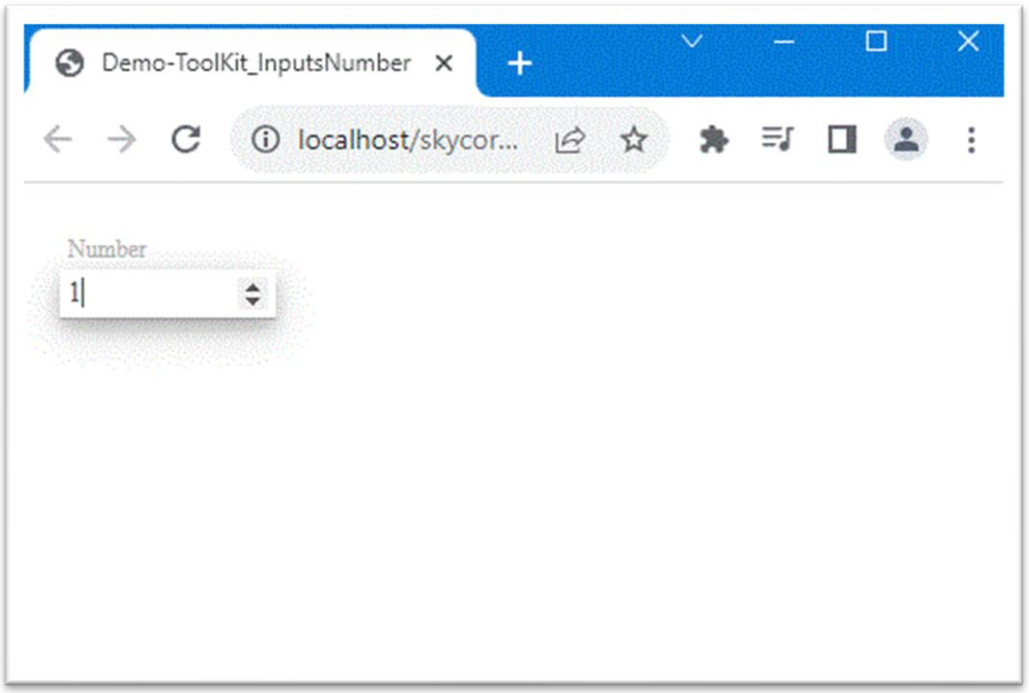
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsNumber
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Number
      input.Label.InnerText = "Number"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsNumber : WebPage
{
        public ToolKit_InputsNumber()
```

```
        {
    }
    public override void OnInitialized()
    {
        ToolKit.Inputs.Number input = new ToolKit.Inputs.Number();
        input.Label.InnerText = "Number";
        HtmlDoc.HtmlBodyText = input.HtmlText();
    }
}
```

Output



WebControl - Inputs.Password

- The Inputs.Password Control creates a password input element.

Code Example

Fig1. Visual Basic

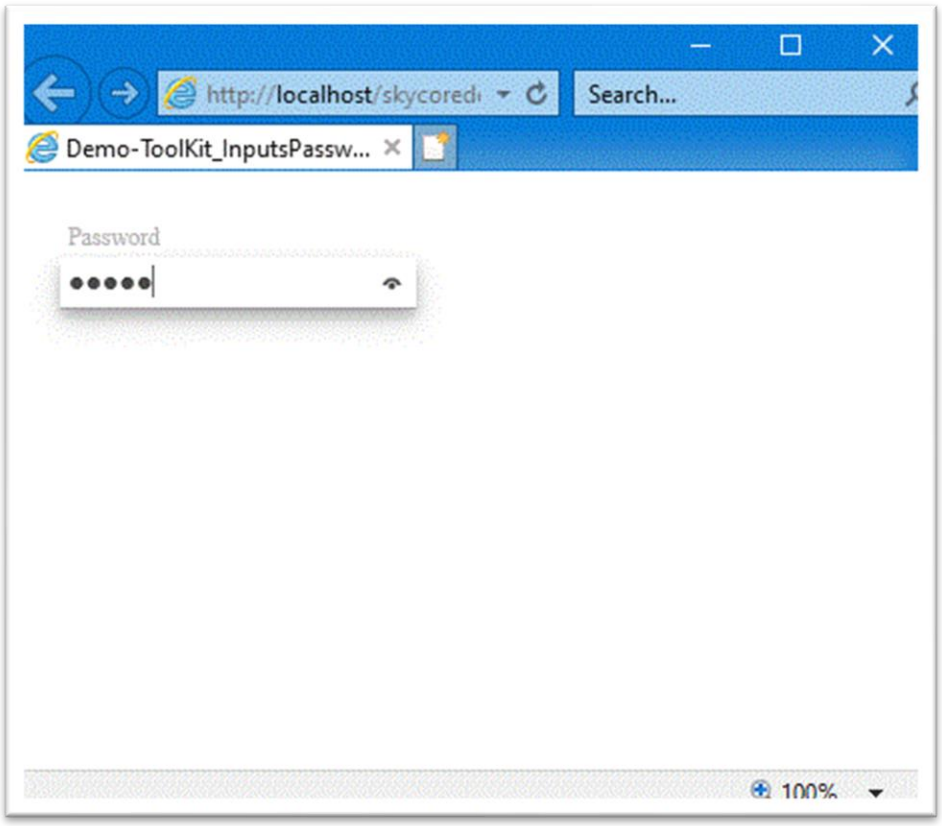
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsPassword
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Password
      input.Label.InnerText = "Password"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsPassword : WebPage
{
        public ToolKit_InputsPassword()
```

```
        {
    }
    public override void OnInitialized()
    {
        ToolKit.Inputs.Password input = new ToolKit.Inputs.Password();
        input.Label.InnerText = "Password";
        HtmlDoc.HtmlBodyText = input.HtmlText();
    }
}
```

Output



WebControl - Inputs.Radio

- The Inputs.Radio Control creates a Radio input element.

Code Example

Fig1. Visual Basic

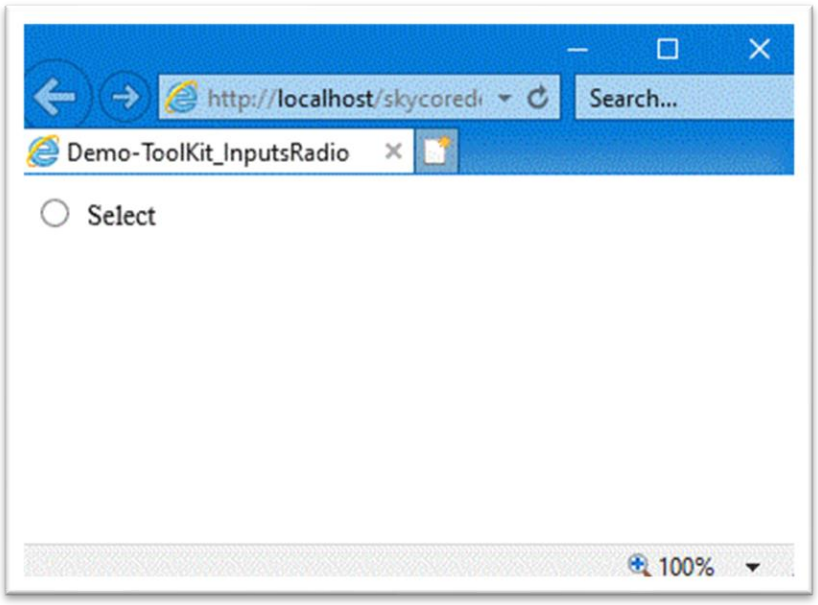
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsRadio
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Radio
      input.Label.InnerText = "Select"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsRadio : WebPage
{
        public ToolKit_InputsRadio()
```

```
        {
    }
    public override void OnInitialized()
    {
        ToolKit.Inputs.Radio input = new ToolKit.Inputs.Radio();
        input.Label.InnerText = "Select";
        HtmlDoc.HtmlBodyText = input.HtmlText();
    }
}
```

Output



WebControl - Inputs.Switch

- The Inputs.Switch Control creates a Switch toggle button element.

Code Example

Fig1. Visual Basic

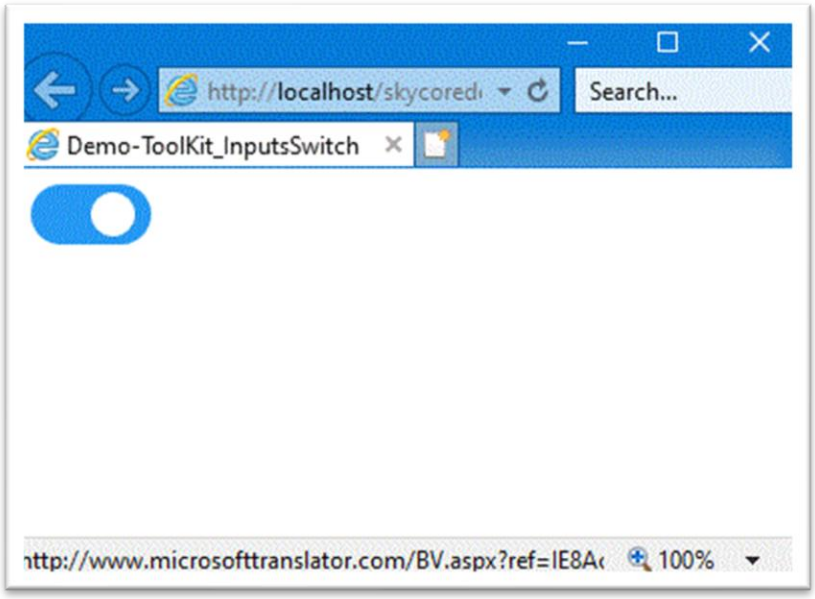
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsSwitch
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Switch
      input.checked = True
      input.id = "sw"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsSwitch : WebPage
{
        public ToolKit_InputsSwitch()
```

```
        {
    }
    public override void OnInitialized()
    {
        ToolKit.Inputs.Switch input = new ToolKit.Inputs.Switch();
        input.@checked = true;
        input.id = "sw";
        HtmlDoc.HtmlBodyText = input.HtmlText();
    }
}
```

Output



WebControl - Inputs.Text

- The Inputs.Text Control creates a Text input element.

Code Example

Fig1. Visual Basic

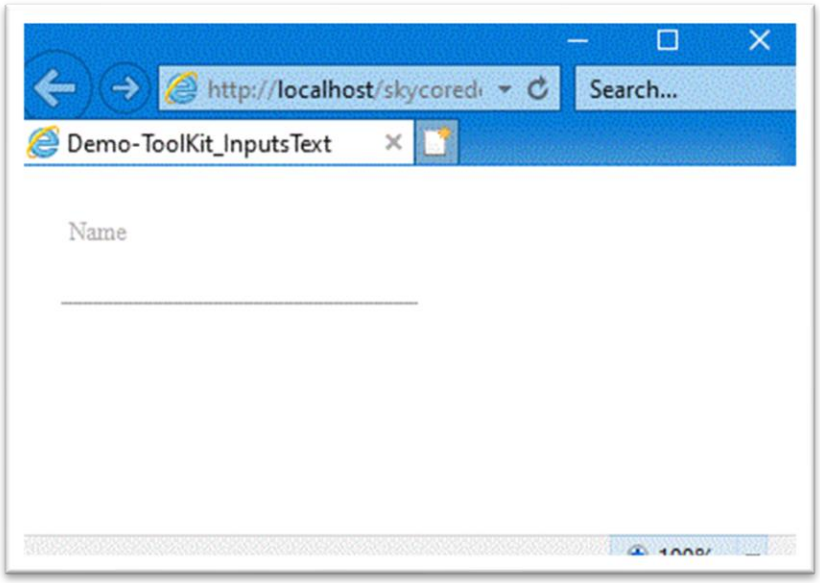
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsText
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Text
      input.Label.InnerText = "Name"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsText : WebPage
{
        public ToolKit_InputsText()
        {
```

```
  }
  public override void OnInitialized()
  {
     ToolKit.Inputs.Text input = new ToolKit.Inputs.Text();
     input.Label.InnerText = "Name";
     HtmlDoc.HtmlBodyText = input.HtmlText();
  }
}
```

Output



WebControl - Inputs.Time

- The Inputs.Time Control creates a Time input element.

Code Example

Fig1. Visual Basic

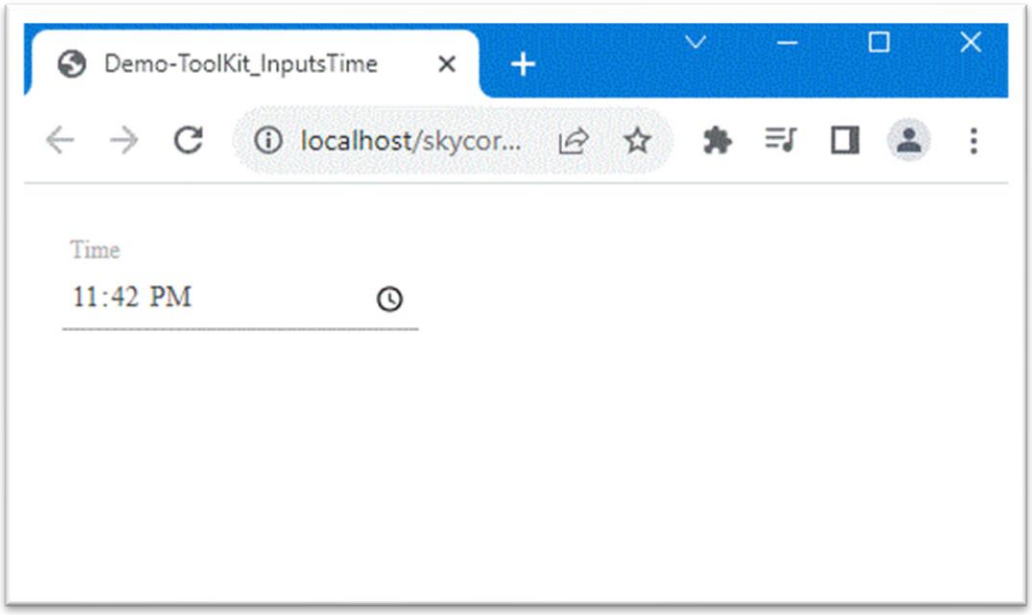
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsTime
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Time
      input.Label.InnerText = "Time"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsTime : WebPage
{
        public ToolKit_InputsTime()
        {
```

```
  }
  public override void OnInitialized()
  {
     ToolKit.Inputs.Time input = new ToolKit.Inputs.Time();
     input.Label.InnerText = "Time";
     HtmlDoc.HtmlBodyText = input.HtmlText();
  }
}
```

Output



WebControl - Inputs.Range

- The Inputs.Range Control creates a Range input element.

Code Example

Fig1. Visual Basic

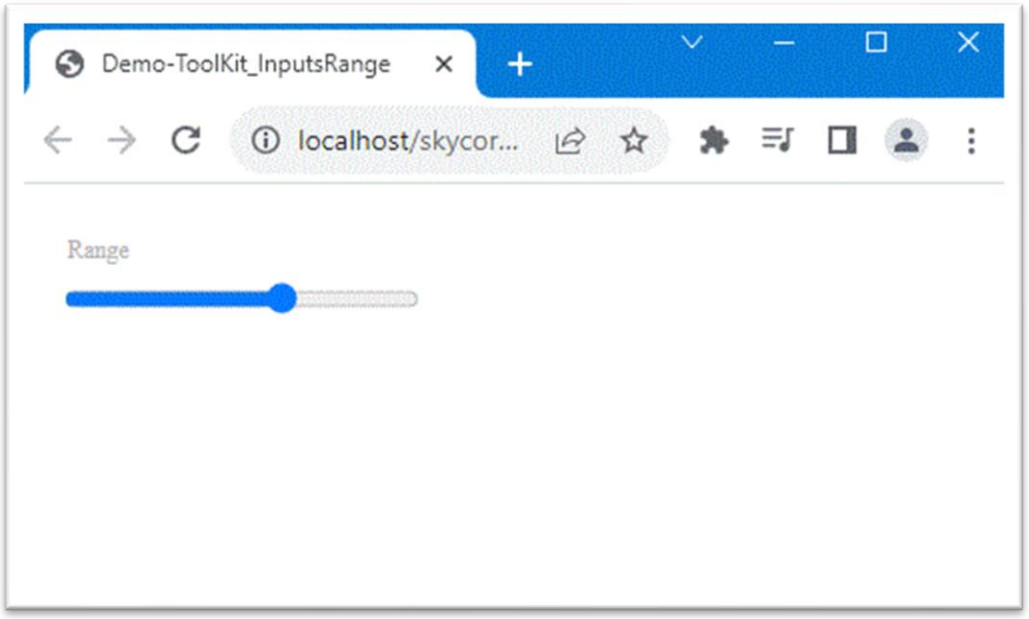
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsRange
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Range
      input.Label.InnerText = "Range"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsRange : WebPage
{
        public ToolKit_InputsRange()
        {
```

```
    }
    public override void OnInitialized()
    {
        ToolKit.Inputs.Range input = new ToolKit.Inputs.Range();
        input.Label.InnerText = "Range";
        HtmlDoc.HtmlBodyText = input.HtmlText();
    }
}
```

Output



WebControl - Inputs.TextArea

- The Inputs.TextArea Control creates a text input element.

Code Example

Fig1. Visual Basic

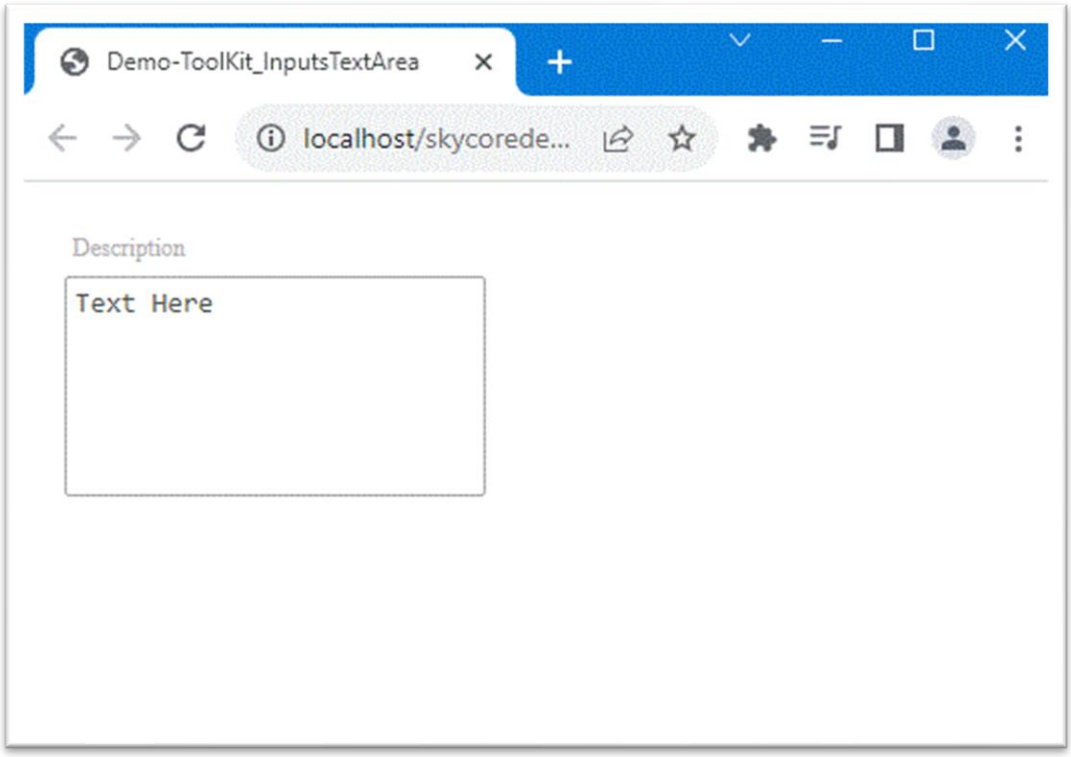
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsTextArea
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.TextArea
      input.Label.InnerText = "Description"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsTextArea : WebPage
{
        public ToolKit_InputsTextArea()
        {
```

```
  }
  public override void OnInitialized()
  {
     ToolKit.Inputs.TextArea input = new ToolKit.Inputs.TextArea();
     input.Label.InnerText = "Description";
     HtmlDoc.HtmlBodyText = input.HtmlText();
  }
}
```

Output



WebControl - Inputs.UIImage

- The Inputs.UIImage Control creates a image input element.

Code Example

Fig1. Visual Basic

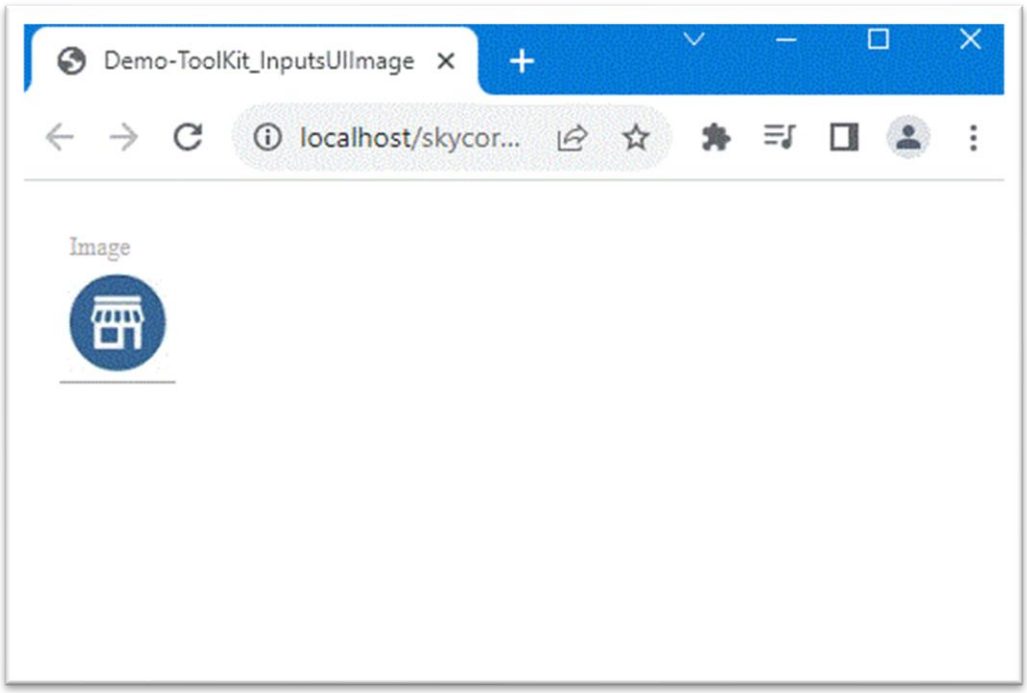
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsUIImage
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.UIImage
      input.Label.InnerText = "Image"
      input.Text.SetAttribute(HtmlAttributes.src, WebCore.ImageAliasPath + "home.jpg")
      input.Text.SetStyle(HtmlStyles.width, "50px")
      input.Text.SetStyle(HtmlStyles.height, "50px")
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsUIImage : WebPage
```

```
{
        public ToolKit_InputsUIImage()
        {
  }
  public override void OnInitialized()
  {
    ToolKit.Inputs.UIImage input = new ToolKit.Inputs.UIImage();
    input.Label.InnerText = "Image";
    input.Text.SetAttribute(HtmlAttributes.src, WebCore.ImageAliasPath + "home.jpg");
    input.Text.SetStyle(HtmlStyles.width, "50px");
    input.Text.SetStyle(HtmlStyles.height, "50px");
    HtmlDoc.HtmlBodyText = input.HtmlText();
  }
}
```

Output



WebControl - Inputs.Week

- The Inputs.Week Control creates a Week input element.

Code Example

Fig1. Visual Basic

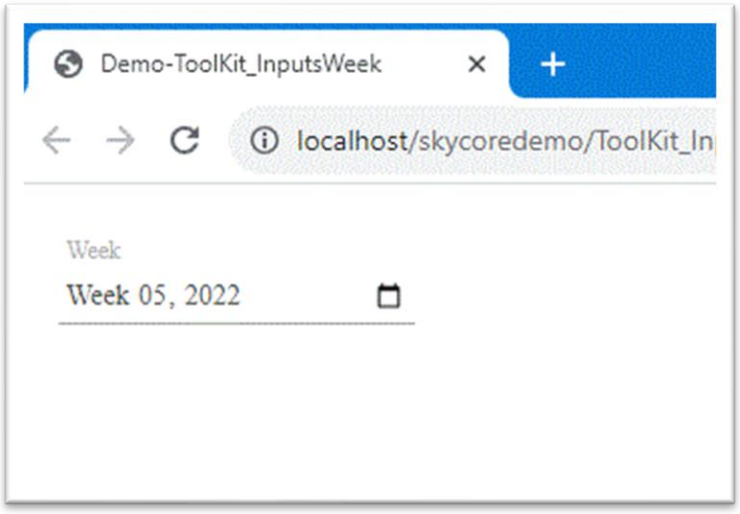
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsWeek
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Week
      input.Label.InnerText = "Week"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsWeek : WebPage
{
        public ToolKit_InputsWeek()
        {
```

```
  }
  public override void OnInitialized()
  {
     ToolKit.Inputs.Week input = new ToolKit.Inputs.Week();
     input.Label.InnerText = "Week";
     HtmlDoc.HtmlBodyText = input.HtmlText();
  }
}
```

Output



WebControl - Inputs.Year

- The Inputs.Year Control creates a Year input element.

Code Example

Fig1. Visual Basic

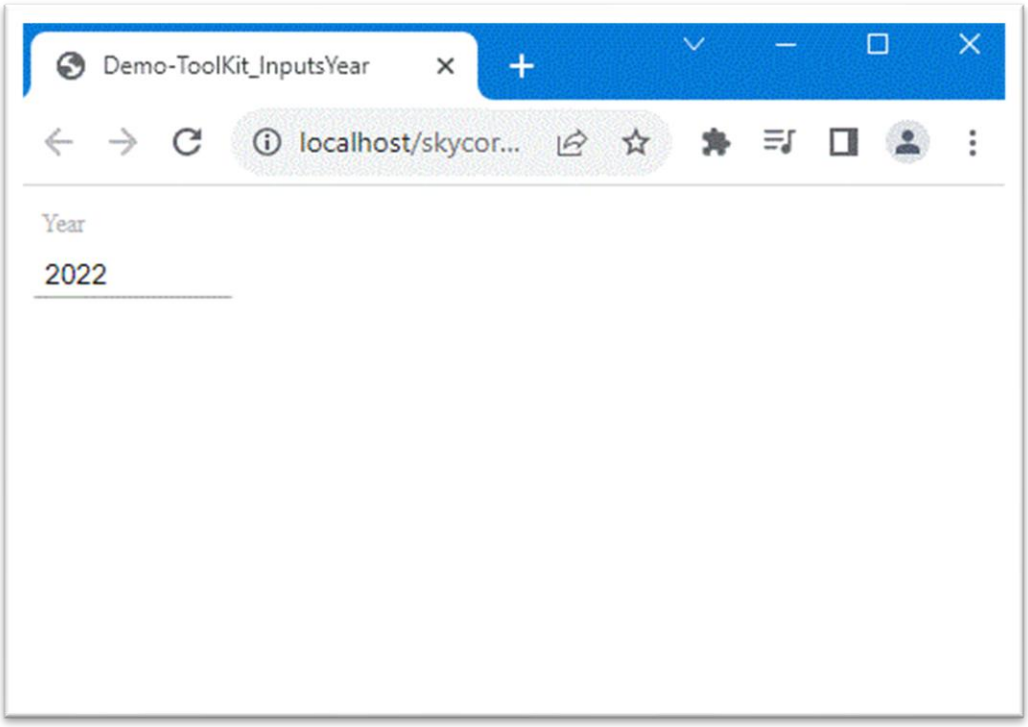
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_InputsYear
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim input As New ToolKit.Inputs.Year
      input.Label.InnerText = "Year"
      HtmlDoc.HtmlBodyText = input.HtmlText
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_InputsYear : WebPage
{
        public ToolKit_InputsYear()
        {
```

```
  }
  public override void OnInitialized()
  {
     ToolKit.Inputs.Year input = new ToolKit.Inputs.Year();
     input.Label.InnerText = "Year";
     HtmlDoc.HtmlBodyText = input.HtmlText();
  }
}
```

Output



WebControl - Charts.BarGraph

- The Charts.BarGraph Control creates a bar graph.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ChartBarGraph
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _BarData As New List(Of ToolKit.Charts.BarData) From {
        (New ToolKit.Charts.BarData With {.Label = "2010", .Item = "TV", .Value = 100, .Color = "#FFA85F"}),
        (New ToolKit.Charts.BarData With {.Label = "2011", .Item = "TV", .Value = 120}),
        (New ToolKit.Charts.BarData With {.Label = "2012", .Item = "TV", .Value = 95}),
        (New ToolKit.Charts.BarData With {.Label = "2013", .Item = "TV", .Value = 130}),
        (New ToolKit.Charts.BarData With {.Label = "2014", .Item = "TV", .Value = 115}),
        (New ToolKit.Charts.BarData With {.Label = "2015", .Item = "TV", .Value = 98}),
        (New ToolKit.Charts.BarData With {.Label = "2013", .Item = "Phone", .Value = 100, .Color = "#149EFE"}),
        (New ToolKit.Charts.BarData With {.Label = "2014", .Item = "Phone", .Value = 150, .Color = "#FFA85F"})
        }
      Dim _BarObject As New ToolKit.Charts.BarObject With _
         {.xAxis = "Year", .yAxis = "Sales", .xFont = "font-family:Arial;font-size:12px;font-weight:bold;", .yFont = "font-
family:Arial;font-size:12px;font-weight:bold;", _
          .vFont = "font-family:Arial;font-size:10px;", .lFont = "font-family:Arial;font-size:10px;", _
          .BarDatas = _BarData}

      Dim chart As New ToolKit.Charts.BarGraph
```

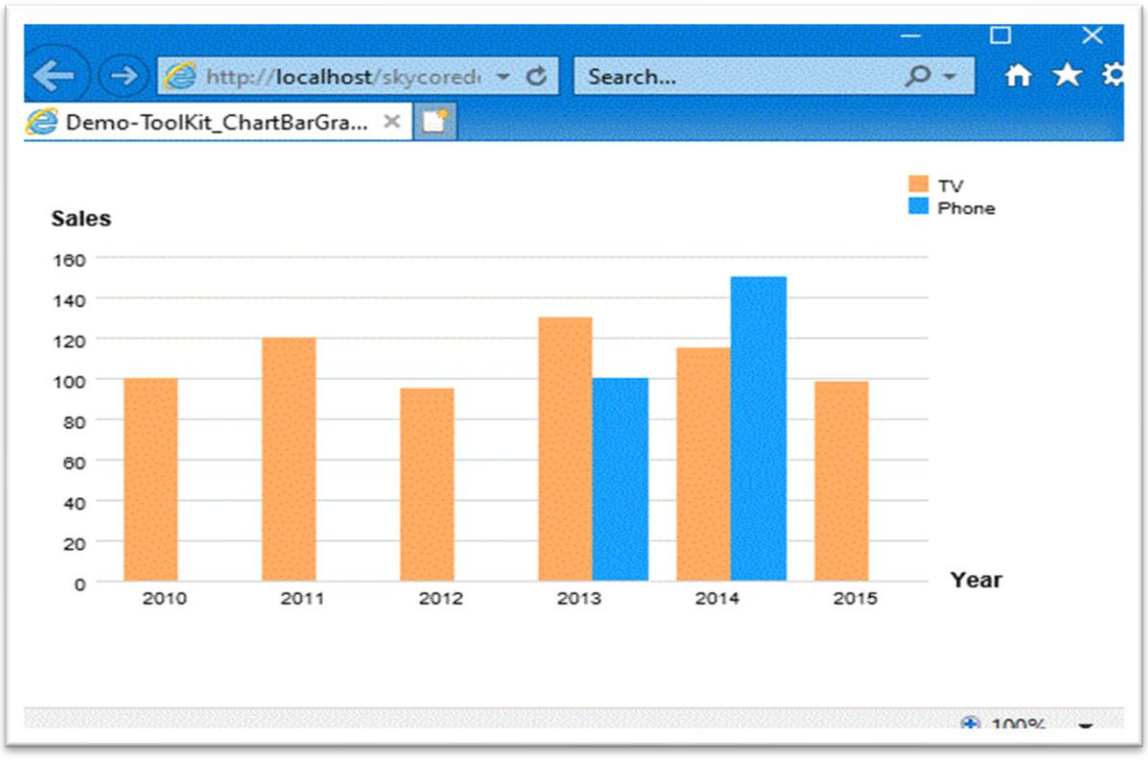
```
    HtmlDoc.HtmlBodyText = chart.Draw(500, 300, _BarObject)
  End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_ChartBarGraph : WebPage
{
        public ToolKit_ChartBarGraph()
        {
  }
  public override void OnInitialized()
  {
    List<ToolKit.Charts.BarData> _BarData = new List<ToolKit.Charts.BarData>();
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2010", Item = "TV", Value = 100, Color = "#FFA85F" });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2011", Item = "TV", Value = 120 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2012", Item = "TV", Value = 95 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2013", Item = "TV", Value = 130 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2014", Item = "TV", Value = 115 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2015", Item = "TV", Value = 98 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2013", Item = "Phone", Value = 100, Color = "#149EFE" });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2014", Item = "Phone", Value = 150, Color = "#FFA85F" });

    ToolKit.Charts.BarObject _BarObject = new ToolKit.Charts.BarObject()
    {
      xAxis = "Year",
      yAxis = "Sales",
```

```
        xFont = "font-family:Arial;font-size:12px;font-weight:bold;",
        yFont = "font-family:Arial;font-size:12px;font-weight:bold;",
        vFont = "font-family:Arial;font-size:10px;",
        lFont = "font-family:Arial;font-size:10px;",
        BarDatas = _BarData
    };

    ToolKit.Charts.BarGraph chart = new ToolKit.Charts.BarGraph();
    HtmlDoc.HtmlBodyText = chart.Draw(500, 300, _BarObject);
  }
}
```

Output



WebControl - Charts.LineGraph

- The Charts.LineGraph Control creates a line graph.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ChartLineGraph
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _BarData As New List(Of ToolKit.Charts.BarData) From {
        (New ToolKit.Charts.BarData With {.Label = "2010", .Item = "TV", .Value = 100, .Color = "#FFA85F"}),
        (New ToolKit.Charts.BarData With {.Label = "2011", .Item = "TV", .Value = 120}),
        (New ToolKit.Charts.BarData With {.Label = "2012", .Item = "TV", .Value = 95}),
        (New ToolKit.Charts.BarData With {.Label = "2013", .Item = "TV", .Value = 130}),
        (New ToolKit.Charts.BarData With {.Label = "2014", .Item = "TV", .Value = 115}),
        (New ToolKit.Charts.BarData With {.Label = "2015", .Item = "TV", .Value = 98}),
        (New ToolKit.Charts.BarData With {.Label = "2013", .Item = "Phone", .Value = 100, .Color = "#149EFE"}),
        (New ToolKit.Charts.BarData With {.Label = "2014", .Item = "Phone", .Value = 150, .Color = "#FFA85F"})
        }
      Dim _BarObject As New ToolKit.Charts.BarObject With _
         {.xAxis = "Year", .yAxis = "Sales", .xFont = "font-family:Arial;font-size:12px;font-weight:bold;", .yFont = "font-
family:Arial;font-size:12px;font-weight:bold;", _
          .vFont = "font-family:Arial;font-size:10px;", .lFont = "font-family:Arial;font-size:10px;", _
          .BarDatas = _BarData}

      Dim chart As New ToolKit.Charts.LineGraph
```

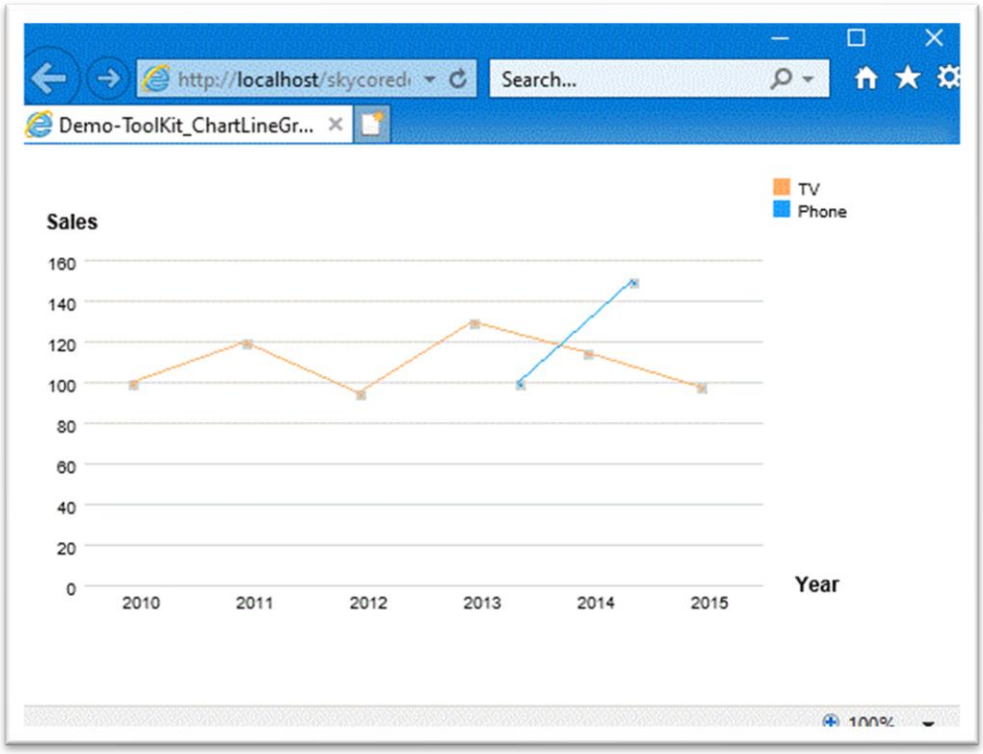
```
    HtmlDoc.HtmlBodyText = chart.Draw(500, 300, _BarObject)
  End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_ChartLineGraph : WebPage
{
        public ToolKit_ChartLineGraph()
        {
  }
  public override void OnInitialized()
  {
    List<ToolKit.Charts.BarData> _BarData = new List<ToolKit.Charts.BarData>();
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2010", Item = "TV", Value = 100, Color = "#FFA85F" });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2011", Item = "TV", Value = 120 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2012", Item = "TV", Value = 95 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2013", Item = "TV", Value = 130 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2014", Item = "TV", Value = 115 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2015", Item = "TV", Value = 98 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2013", Item = "Phone", Value = 100, Color = "#149EFE" });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2014", Item = "Phone", Value = 150, Color = "#FFA85F" });

    ToolKit.Charts.BarObject _BarObject = new ToolKit.Charts.BarObject()
    {
      xAxis = "Year",
      yAxis = "Sales",
```

```
        xFont = "font-family:Arial;font-size:12px;font-weight:bold;",
        yFont = "font-family:Arial;font-size:12px;font-weight:bold;",
        vFont = "font-family:Arial;font-size:10px;",
        lFont = "font-family:Arial;font-size:10px;",
        BarDatas = _BarData
    };

    ToolKit.Charts.LineGraph chart = new ToolKit.Charts.LineGraph();
    HtmlDoc.HtmlBodyText = chart.Draw(500, 300, _BarObject);
  }
}
```

Output



WebControl - Charts.LinebarGraph

- The Charts.LinebarGraph Control creates a line and bar graph.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ChartLinebarGraph
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _BarData As New List(Of ToolKit.Charts.BarData) From {
        (New ToolKit.Charts.BarData With {.Label = "2010", .Item = "TV", .Value = 100, .Color = "#FFA85F"}),
        (New ToolKit.Charts.BarData With {.Label = "2011", .Item = "TV", .Value = 120}),
        (New ToolKit.Charts.BarData With {.Label = "2012", .Item = "TV", .Value = 95}),
        (New ToolKit.Charts.BarData With {.Label = "2013", .Item = "TV", .Value = 130}),
        (New ToolKit.Charts.BarData With {.Label = "2014", .Item = "TV", .Value = 115}),
        (New ToolKit.Charts.BarData With {.Label = "2015", .Item = "TV", .Value = 98}),
        (New ToolKit.Charts.BarData With {.Label = "2013", .Item = "Phone", .Value = 100, .Color = "#149EFE"}),
        (New ToolKit.Charts.BarData With {.Label = "2014", .Item = "Phone", .Value = 150, .Color = "#FFA85F"})
        }
      Dim _BarObject As New ToolKit.Charts.BarObject With _
         {.xAxis = "Year", .yAxis = "Sales", .xFont = "font-family:Arial;font-size:12px;font-weight:bold;", .yFont = "font-
family:Arial;font-size:12px;font-weight:bold;", _
          .vFont = "font-family:Arial;font-size:10px;", .lFont = "font-family:Arial;font-size:10px;", _
          .BarDatas = _BarData}

      Dim chart As New ToolKit.Charts.LinebarGraph
```

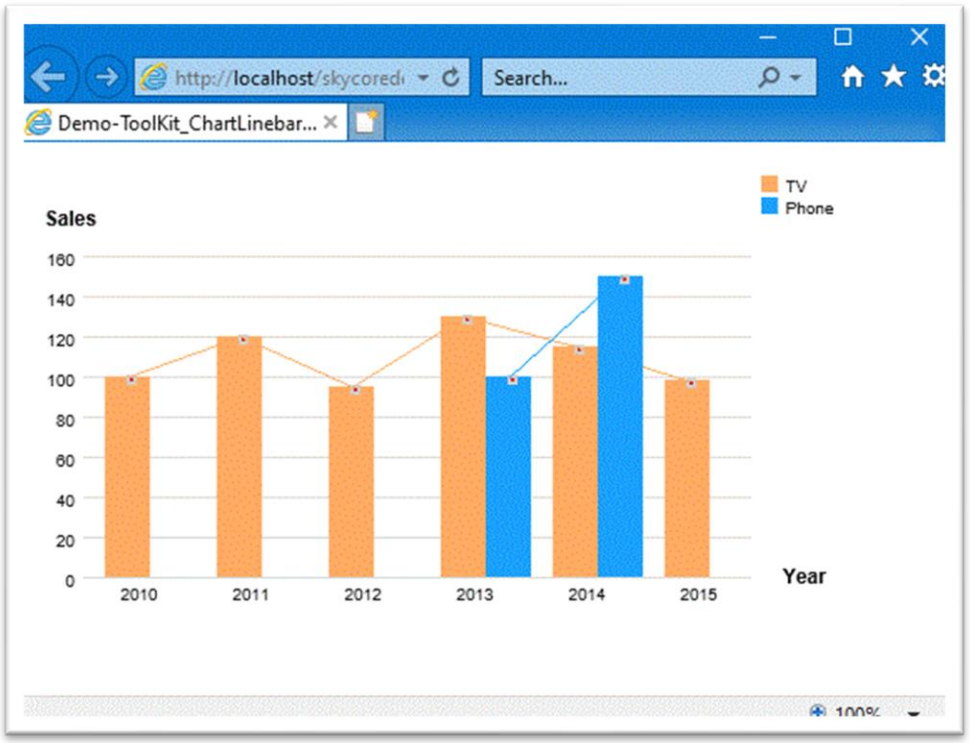
```
    HtmlDoc.HtmlBodyText = chart.Draw(500, 300, _BarObject)
  End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_ChartLinebarGraph : WebPage
{
        public ToolKit_ChartLinebarGraph()
        {
  }
  public override void OnInitialized()
  {
    List<ToolKit.Charts.BarData> _BarData = new List<ToolKit.Charts.BarData>();
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2010", Item = "TV", Value = 100, Color = "#FFA85F" });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2011", Item = "TV", Value = 120 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2012", Item = "TV", Value = 95 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2013", Item = "TV", Value = 130 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2014", Item = "TV", Value = 115 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2015", Item = "TV", Value = 98 });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2013", Item = "Phone", Value = 100, Color = "#149EFE" });
    _BarData.Add(new ToolKit.Charts.BarData { Label = "2014", Item = "Phone", Value = 150, Color = "#FFA85F" });

    ToolKit.Charts.BarObject _BarObject = new ToolKit.Charts.BarObject()
    {
      xAxis = "Year",
      yAxis = "Sales",
```

```
        xFont = "font-family:Arial;font-size:12px;font-weight:bold;",
        yFont = "font-family:Arial;font-size:12px;font-weight:bold;",
        vFont = "font-family:Arial;font-size:10px;",
        lFont = "font-family:Arial;font-size:10px;",
        BarDatas = _BarData
    };

    ToolKit.Charts.LinebarGraph chart = new ToolKit.Charts.LinebarGraph();
    HtmlDoc.HtmlBodyText = chart.Draw(500, 300, _BarObject);
  }
}
```

Output



WebControl - Charts.PieGraph

- The Charts.PieGraph Control creates a pie graph.

Code Example

Fig1. Visual Basic

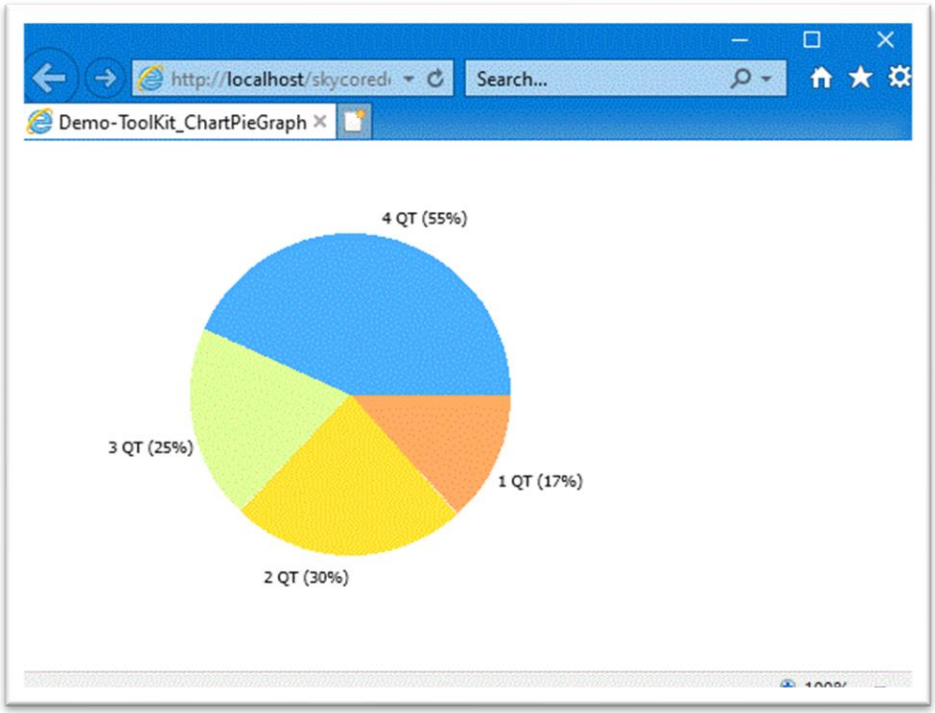
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ChartPieGraph
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _PieData As New List(Of ToolKit.Charts.PieData) From { _
         (New ToolKit.Charts.PieData With {.Name = "1 QT", .Unit = "%", .Value = "17", .Color = "#FFA85F", .Attr =
"id=""1"""}), _
         (New ToolKit.Charts.PieData With {.Name = "2 QT", .Unit = "%", .Value = "30", .Color = "#FFE32E"}), _
         (New ToolKit.Charts.PieData With {.Name = "3 QT", .Unit = "%", .Value = "25"}), _
         (New ToolKit.Charts.PieData With {.Name = "4 QT", .Unit = "%", .Value = "55", .Color = "#46AEFF"}) _
         }

      Dim chart As New ToolKit.Charts.PieGraph
      HtmlDoc.HtmlBodyText = chart.Draw(400, 300, _PieData)
   End Sub
End Class
```

Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_ChartPieGraph : WebPage
```

```
{
        public ToolKit_ChartPieGraph()
        {
  }
  public override void OnInitialized()
  {
    List<ToolKit.Charts.PieData> _PieData = new List<ToolKit.Charts.PieData>();
    _PieData.Add(new ToolKit.Charts.PieData() { Name = "1 QT", Unit = "%", Value = 17, Color = "#FFA85F", Attr =
@"id=""1""" });
    _PieData.Add(new ToolKit.Charts.PieData() { Name = "2 QT", Unit = "%", Value = 30, Color = "#FFE32E" });

    ToolKit.Charts.PieGraph chart = new ToolKit.Charts.PieGraph();
    HtmlDoc.HtmlBodyText = chart.Draw(400, 300, _PieData);
  }
}
```

Output



WebControl - Charts.Schedule

- The Charts.Schedule Control creates a simple monthly, weekly, daily schedule.

Code Example

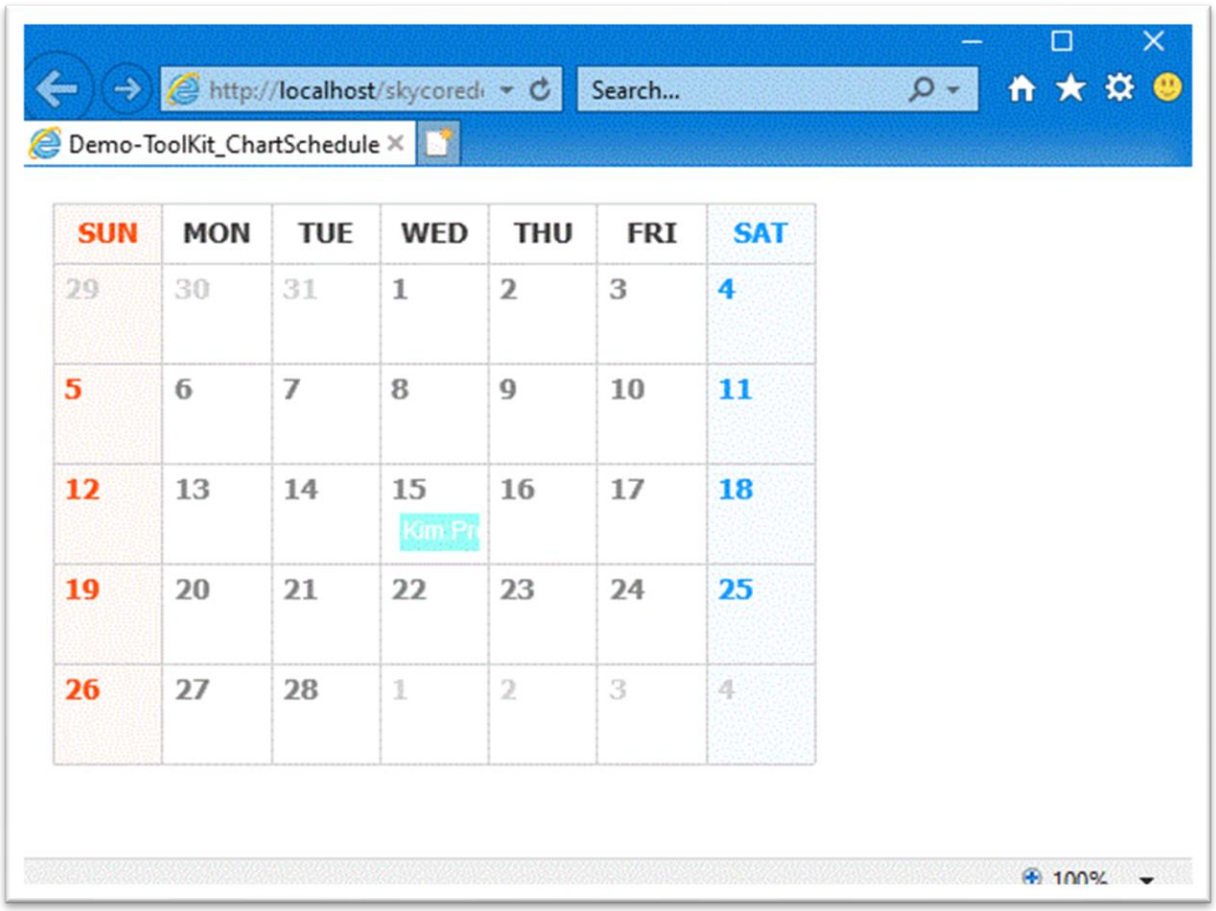
Fig1. Visual Basic

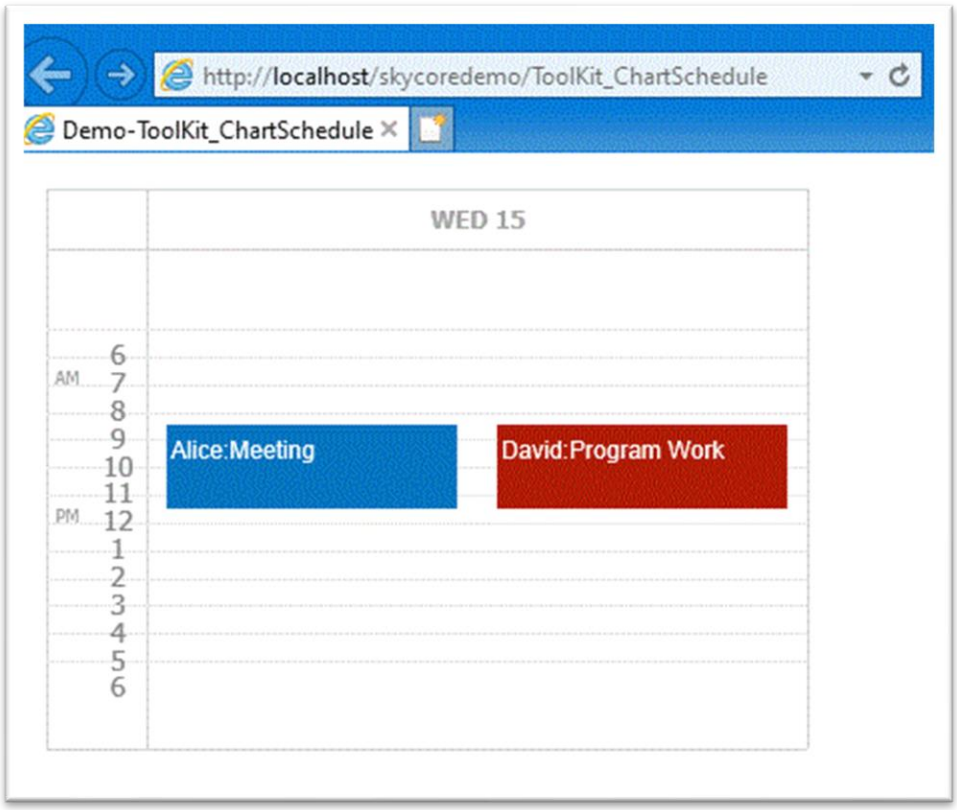
```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ChartSchedule
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _ScheduleHoliday As New List(Of ToolKit.Charts.ScheduleHoliday)
      Dim _ScheduleData As New List(Of ToolKit.Charts.ScheduleData) From {
         (New ToolKit.Charts.ScheduleData With {.Name = "David", .Task = "Program Work", .StdDte = "02/15/2023",
.StdTme = "09:30", .EndTme = "12:30", .Font = "font-family:Arial;font-size:12px;color:#fff;cursor:pointer;", .Attr =
"onclick=""alert('1')"""}),
         (New ToolKit.Charts.ScheduleData With {.Name = "Alice", .Task = "Meeting", .StdDte = "02/15/2023", .StdTme =
"09:30", .EndTme = "12:30", .Font = "font-family:Arial;font-size:12px;color:#fff;"})
         }
      Dim _ScheduleObject As New ToolKit.Charts.ScheduleObject With _
         {.Type = "m", .StdDte = "02/15/2023", .StdTme = "6", .EndTme = "18", _
          .hFont = "font-family:Tahoma;font-size:12px;color:#a0a0a0;font-weight:bold;", _
          .ScheduleDatas = _ScheduleData,
          .ScheduleHolidays = _ScheduleHoliday}
      Dim chart As New ToolKit.Charts.Schedule
      HtmlDoc.HtmlBodyText = chart.Draw(400, 300, _ScheduleObject)
   End Sub
End Class
```

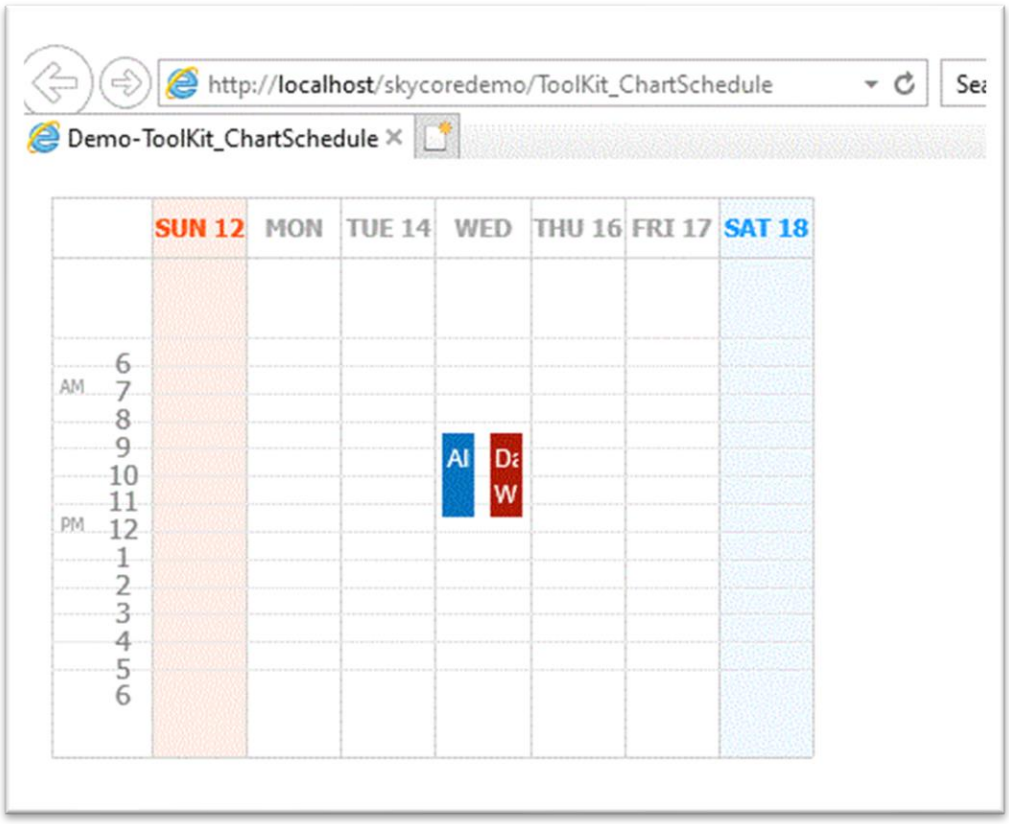
Fig2. CSharp

```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_ChartSchedule : WebPage
{
        public ToolKit_ChartSchedule()
        {
   }
   public override void OnInitialized()
   {
      List<ToolKit.Charts.ScheduleHoliday> _ScheduleHoliday = new List<ToolKit.Charts.ScheduleHoliday>();
      List<ToolKit.Charts.ScheduleData> _ScheduleData = new List<ToolKit.Charts.ScheduleData>();
      _ScheduleData.Add(new ToolKit.Charts.ScheduleData() { Name = "David", Task = "Program Work", StdDte = @"02/15/2023", StdTme = "09:30", EndTme = "12:30", Font = @"font-family:Arial;font-size:12px;color:#fff;cursor:pointer;", Attr = @"onclick=""alert('1')""" });
      _ScheduleData.Add(new ToolKit.Charts.ScheduleData() { Name = "Alice", Task = "Meeting", StdDte = "02/15/2023", StdTme = "09:30", EndTme = "12:30", Font = @"font-family:Arial;font-size:12px;color:#fff;"});

      ToolKit.Charts.ScheduleObject _ScheduleObject = new ToolKit.Charts.ScheduleObject() {
         Type = "m", StdDte = "02/15/2023", StdTme = "6", EndTme = "18",
         hFont = "font-family:Tahoma;font-size:12px;color:#a0a0a0;font-weight:bold;",
         ScheduleDatas = _ScheduleData,
         ScheduleHolidays = _ScheduleHoliday
      };

      ToolKit.Charts.Schedule chart = new ToolKit.Charts.Schedule();
      HtmlDoc.HtmlBodyText = chart.Draw(400, 300, _ScheduleObject);
   }
}
```

Output


http://localhost/skycored
Search...
Demo-ToolKit_ChartSchedule
SUN MON TUE WED THU FRI SAT
29 30 31 1 2 3 4
5 6 7 8 9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 1 2 3 4
100%


http://localhost/skycoredemo/ToolKit_ChartSchedule
Demo-ToolKit_ChartSchedule
WED 15
AM
PM
6 7 8 9 10 11 12 1 2 3 4 5 6
Alice:Meeting
David:Program Work


http://localhost/skycoredemo/ToolKit_ChartSchedule
Demo-ToolKit_ChartSchedule
SUN 12
MON
TUE 14
WED
THU 16
FRI 17
SAT 18
AM
PM
6
7
8
9
10
11
12
1
2
3
4
5
6
Al
Da
W

WebControl - Charts.Timeline

- The Charts.Timeline Control creates a simple timeline schedule.

Code Example

Fig1. Visual Basic

```
Imports Microsoft.VisualBasic
Imports skycore

Public Class ToolKit_ChartTimeline
   Inherits WebPage

   Public Overrides Sub OnInitialized()
      Dim _TimeLineData As New List(Of ToolKit.Charts.TimeLineData) From {
        (New ToolKit.Charts.TimeLineData With {.Label = "Planning", .StdDte = "2016-11-28", .EndDte = "2016-12-05",
.Color = "#FFA85F", .Attr = "onclick=""alert(1);""", .Font = "font-family:San-serif;font-size:12px;color:blue;"}),
        (New ToolKit.Charts.TimeLineData With {.Label = "Analysis", .StdDte = "2016-12-01", .EndDte = "2016-12-10"}),
        (New ToolKit.Charts.TimeLineData With {.Label = "Design", .StdDte = "2016-12-11", .EndDte = "2016-12-16"}),
        (New ToolKit.Charts.TimeLineData With {.Label = "Development", .StdDte = "2016-12-17", .EndDte = "2017-01-
10"}),
        (New ToolKit.Charts.TimeLineData With {.Label = "Test", .StdDte = "2017-01-11", .EndDte = "2017-01-15"}),
        (New ToolKit.Charts.TimeLineData With {.Label = "Implementation", .StdDte = "2017-01-13", .EndDte = "2017-01-
22"}),
        (New ToolKit.Charts.TimeLineData With {.Label = "Delivery", .StdDte = "2017-01-25", .EndDte = "2017-02-05"})
        }

      'type q,m,d
      Dim _TimeLineObject As New ToolKit.Charts.TimeLineObject With _
        {.Type = "d", .xAxis = "Activity", .xFont = "font-family:Arial;font-size:12px;font-weight:bold;", _
         .hFont = "font-family:Arial;font-size:10px;color:#282828;", _
         .hBackColor = "#CDE8FF", .lineColor = "#c0c0c0", _
         .TimeLineDatas = _TimeLineData}
```

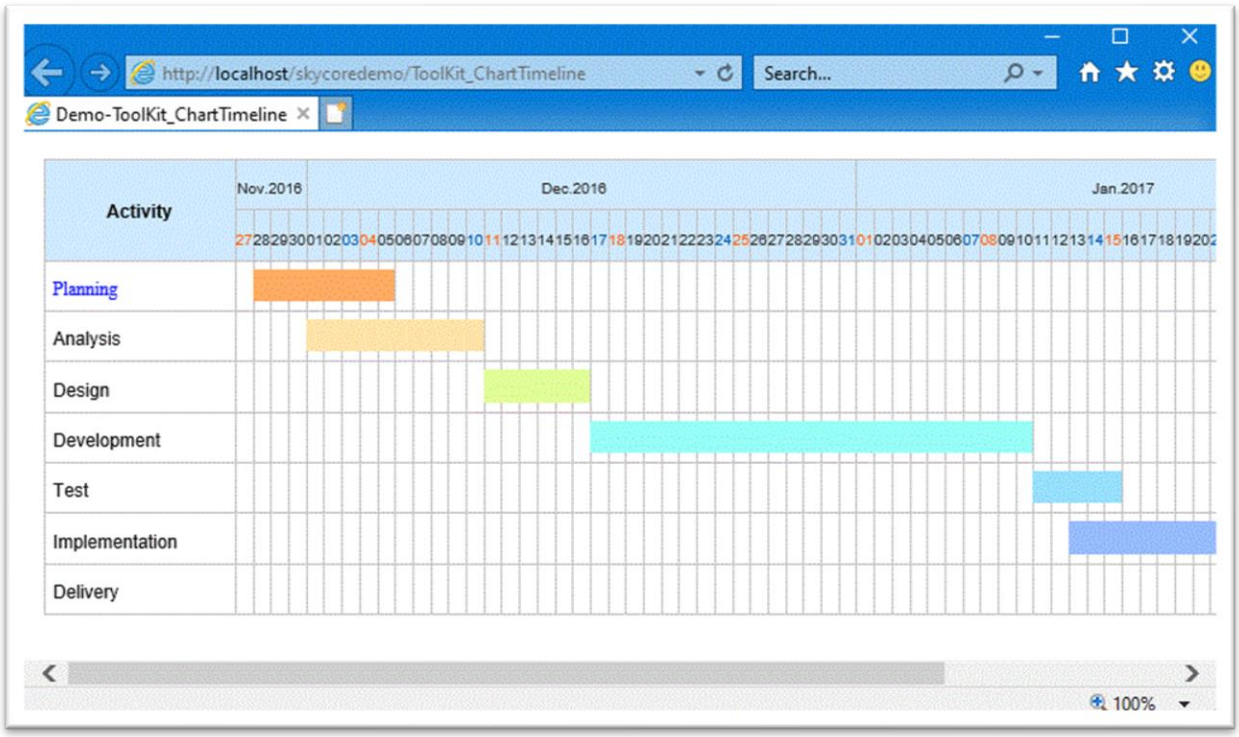
```
    Dim chart As New ToolKit.Charts.Timeline
    HtmlDoc.HtmlBodyText = chart.Draw(1000, 300, _TimeLineObject)
  End Sub
End Class
```

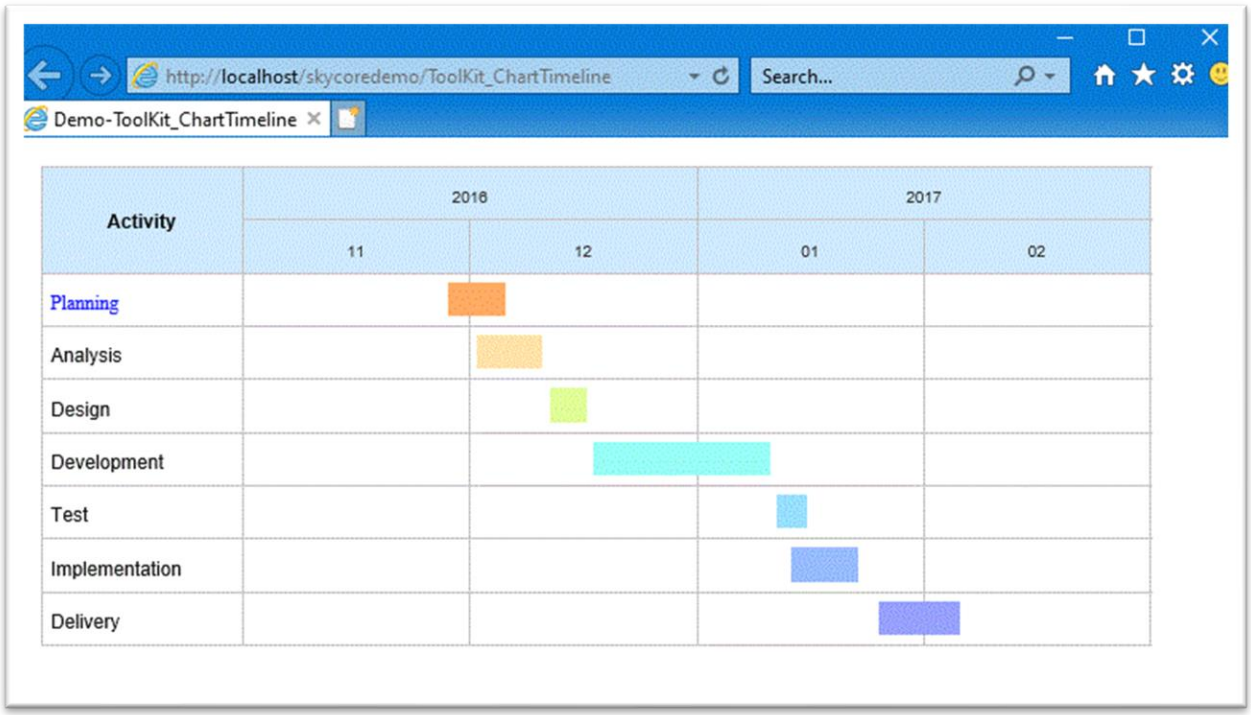
Fig2. CSharp

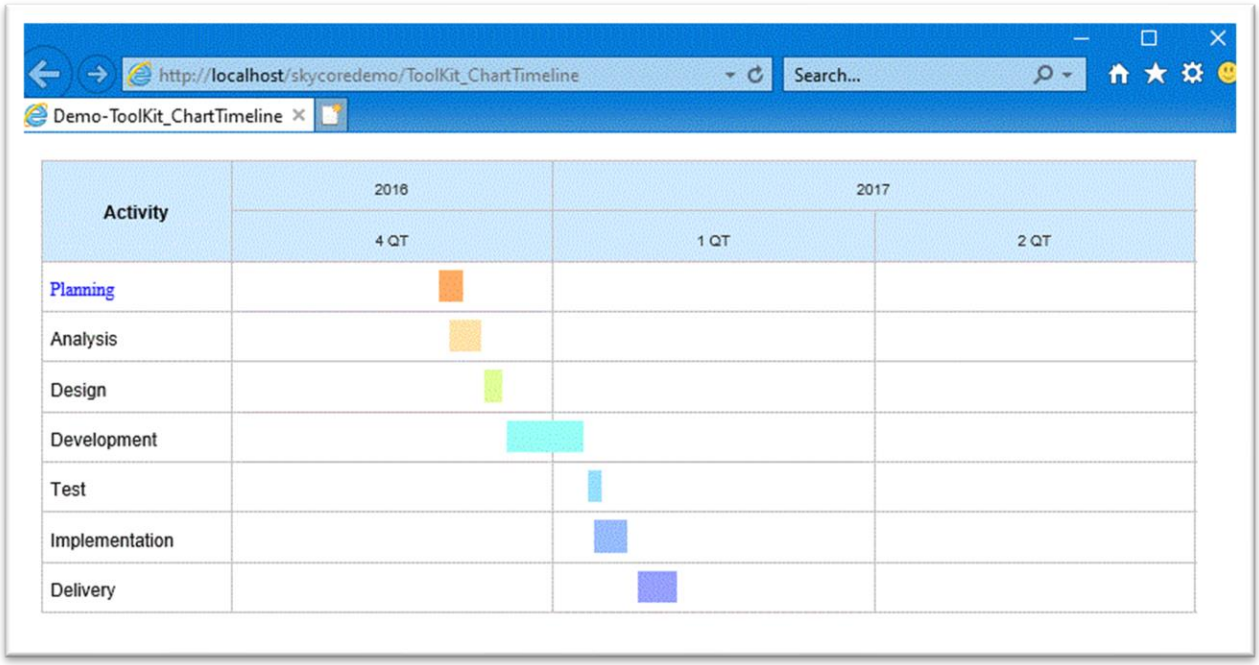
```
using System;
using System.Collections.Generic;
using skycore;

public class ToolKit_ChartTimeline : WebPage
{
        public ToolKit_ChartTimeline()
        {
  }
  public override void OnInitialized()
  {
    List<ToolKit.Charts.TimeLineData> _TimeLineData = new List<ToolKit.Charts.TimeLineData>();
    _TimeLineData.Add(new ToolKit.Charts.TimeLineData() {Label = "Planning", StdDte = "2016-11-28", EndDte = "2016-12-05", Color = "#FFA85F", Attr = @"onclick=""alert(1);""", Font = "font-family:San-serif;font-size:12px;color:blue;"});
    _TimeLineData.Add(new ToolKit.Charts.TimeLineData(){Label = "Analysis", StdDte = "2016-12-01", EndDte = "2016-12-10"});
    _TimeLineData.Add(new ToolKit.Charts.TimeLineData(){Label = "Design", StdDte = "2016-12-11", EndDte = "2016-12-16"});
    _TimeLineData.Add(new ToolKit.Charts.TimeLineData(){Label = "Development", StdDte = "2016-12-17", EndDte = "2017-01-10"});
    _TimeLineData.Add(new ToolKit.Charts.TimeLineData(){Label = "Test", StdDte = "2017-01-11", EndDte = "2017-01-15"});
    _TimeLineData.Add(new ToolKit.Charts.TimeLineData(){Label = "Implementation", StdDte = "2017-01-13", EndDte = "2017-01-22"});
    _TimeLineData.Add(new ToolKit.Charts.TimeLineData(){Label = "Delivery", StdDte = "2017-01-25", EndDte = "2017-02-05"});
```

```
    ToolKit.Charts.TimeLineObject _TimeLineObject = new ToolKit.Charts.TimeLineObject()
    {
      Type = "d", xAxis = "Activity", xFont = "font-family:Arial;font-size:12px;font-weight:bold;",
       hFont = "font-family:Arial;font-size:10px;color:#282828;",
       hBackColor = "#CDE8FF", lineColor = "#c0c0c0",
       TimeLineDatas = _TimeLineData
    };

    ToolKit.Charts.Timeline chart = new ToolKit.Charts.Timeline();
    HtmlDoc.HtmlBodyText = chart.Draw(1000, 300, _TimeLineObject);
  }
}
```

Output




http://localhost/skycoredemo/ToolKit_ChartTimeline
Search...
Demo-ToolKit_ChartTimeline
Activity
2016
2017
11
12
01
02
Planning
Analysis
Design
Development
Test
Implementation
Delivery


http://localhost/skycoredemo/ToolKit_ChartTimeline
Search...
Demo-ToolKit_ChartTimeline
Activity
2016
2017
4 QT
1 QT
2 QT
Planning
Analysis
Design
Development
Test
Implementation
Delivery

Database Classes

- SkyCore Framework provides easy tools to retrieve and save data into MS-SQL and Oracle databases.

Handling Database

Class	Description
SQLData	This class handles MS-SQL database.
OraData	This class handles Oracle databases. In order to use Oracle Database, OraOledb must be registerd in system. *** regsvr32.exe OraOledb12.dll
OleData	This class handles various type of data files.

SQLData Class

- This class handles MS-SQL database

SQLInfo Class

Property Name	Description
DataSource	Database Server.
DatabaseName	Database Name
UserId	Database connection user id
Password	Database connection user's password
TimeOut	Database command timeout.
ConnectionString	Database ConnectionString.

SQLData Methods

Method Name	Description
DataGetSet (SQL As String, ByRef SqlDataSet As System.Data.DataSet, ByRef eMsg As String)	Retrieve dataset from database - SQL: SQL Statement. - (ref) SqlDataSet: Return dataset - (ref) eMsg: Return error message
DataPut (SQL As List(Of String), ByRef eMsg As String)	Manipulate data - SQL: SQL Statement. - (ref) eMsg: Return error message
DataPutNoTran (SQL As List(Of String), ByRef eMsg As String)	Manipulate data without transaction - SQL: SQL Statement. - (ref) eMsg: Return error message
DataBulkInsert (TableName As String, dt As DataTable, ByRef eMsg As String)	Manipulate data without transaction - TableName: Database Table. - dt: Datatable to insert. - (ref) eMsg: Return error message

Example

Fig1. Using default SQL Database in application.cfg file

```
Dim ds As System.Data.DataSet = Nothing
Dim emsg As String = String.Empty
Dim sql As String = "select top 20 * from Person.Address;"

Dim _SQLData As New SQLData
Select Case _SQLData.DataGetSet(sql, ds, emsg)
   Case True
   Case False
End Select
```

Fig2. Custom SQL Database connection

```
Dim _SQLInfo As New SQLInfo
_SQLInfo.DataSource = "192.168.0.10"
_SQLInfo.DatabaseName = "AdvantureWorks"
_SQLInfo.UserId = "sa"
_SQLInfo.Password = "pass1234"

Dim ds As System.Data.DataSet = Nothing
Dim emsg As String = String.Empty
Dim sql As String = "select top 20 * from Person.Address;"

Dim _SQLData As New SQLData
Select Case _SQLData.DataGetSet(sql, ds, emsg)
   Case True
   Case False
End Select
```

OraData Class

- This class handles Oracle database

OraInfo Class

Property Name	Description
Host	Database Server.
Port	Database connection port number.
Service	Service Name
UserId	Database connection user id
Password	Database connection user's password
TimeOut	Database command timeout.
ConnectionString	Database ConnectionString.

OraData Methods

Method Name	Description
DataGetSet (SQL As String, ByRef SqlDataSet As System.Data.DataSet, ByRef eMsg As String)	Retrieve dataset from database - SQL: SQL Statement. - (ref) SqlDataSet: Return dataset - (ref) eMsg: Return error message
DataPut (SQL As List(Of String), ByRef eMsg As String)	Manipulate data - SQL: SQL Statement. - (ref) eMsg: Return error message

Example

Fig1. Using default Oracle Database in application.cfg file

```
Dim ds As System.Data.DataSet = Nothing
Dim emsg As String = String.Empty
Dim Sql As String = "select top 20 * from Person.Address;"

Dim _OraData As New OraData
Select Case _OraData.DataGetSet(Sql, ds, emsg)
   Case True
   Case False
End Select
```

OleData Class

- his class handles various type of data files.

OleData Methods

Method Name	Description
DataGetSet (SQL As String, ByRef SqlDataSet As System.Data.DataSet, ByRef eMsg As String)	Retrieve dataset from database - SQL: SQL Statement. - (ref) SqlDataSet: Return dataset - (ref) eMsg: Return error message
DataPut (SQL As List(Of String), ByRef eMsg As String)	Manipulate data - SQL: SQL Statement. - (ref) eMsg: Return error message
GetOleDataTable (SQL As List(Of String), ByRef eMsg As String)	Retrieve datatable - SQL: SQL Statement. - (ref) eMsg: Return error message