Fundusze Europejskie Wiedza Edukacja Rozwój
Rzeczpospolita Polska
Unia Europejska Europejski Fundusz Społeczny

Strony HTML i ASP.NET

Technologie Internetowe

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Serwer WWW

- Serwer WWW to program działający na serwerze internetowym. Obsługuje żądania HTTP i odsyła do klienta stronę HTML lub dane w dowolnym formacie np. XML, JSON, YAML.
- Na serwerze można umieszczać statyczne witryny złożone ze stron HTML oraz aplikacje i usługi sieciowe.
- Odpowiedź może być statycznym plikiem HTML zapisanym na dysku lub kodem wygenerowanym przez aplikację lub usługę sieciową umieszczoną na serwerze.
- Ranking serwerów WWW https://news.netcraft.com/archives/category/web-server-survey/

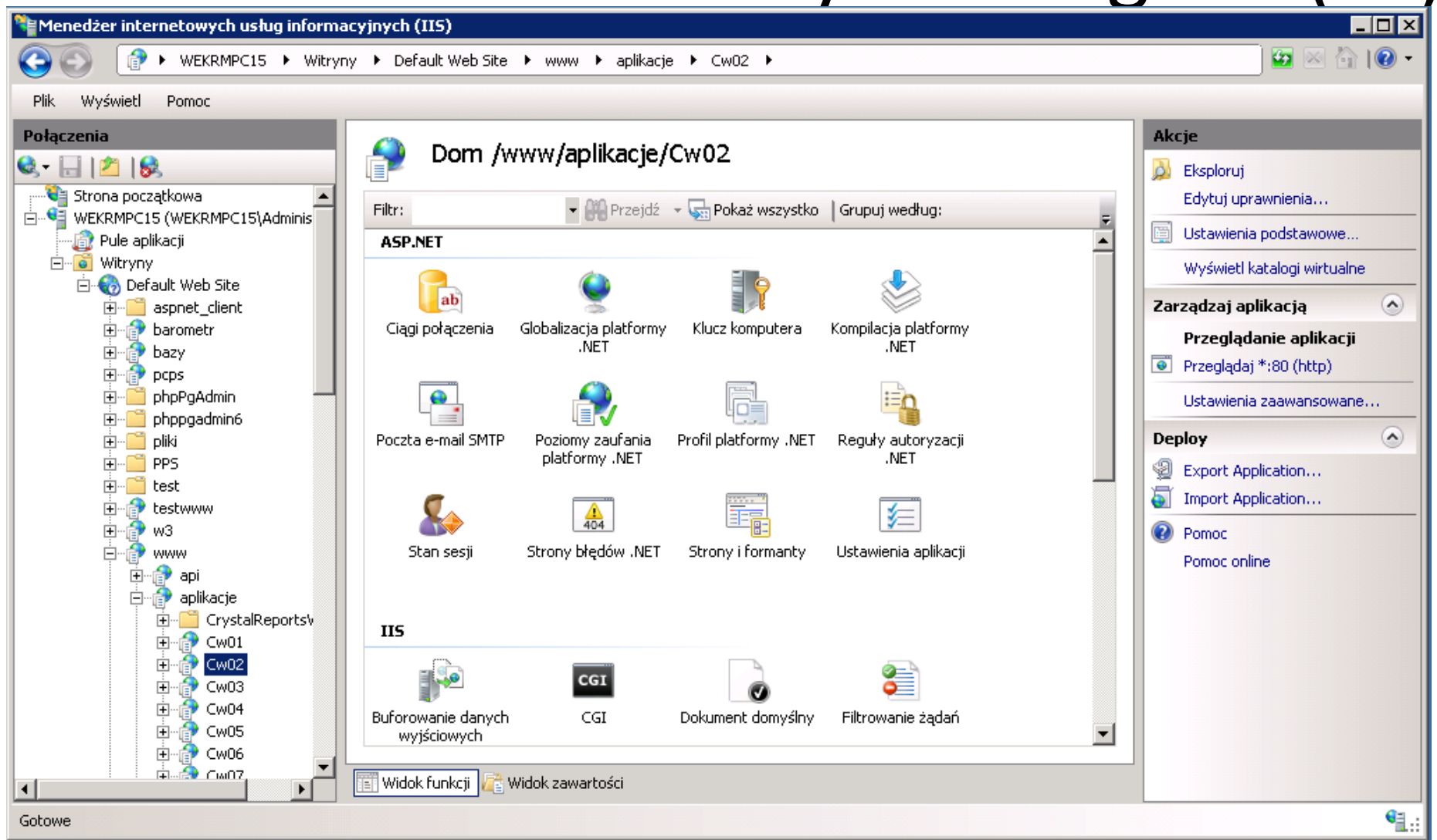
Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Własny serwer WWW można zrobić w NodeRED

- W przypadku dużej liczby obsługiwanych plików można stworzyć jeden punkt końcowy, który będzie obsługiwał wiele adresów URL zawierających parametr, np. nazwę pliku.
  - W tym celu można wykorzystać „*” w adresie URL bloczka.
  - Jeśli parametr URL bloczka http in będzie zawierał na końcu „*”, np. /titi/pliki/*, to punkt końcowy będzie obsługiwał wszystkie żądania o adresie URL zaczynającym się od /titi/pliki/, resztę adresu przekazując jako parametr msg.req.params.
  - Cały URL można odczytać jako msg.req.url.
  - Użycie „*” zastępuje wiele poziomów adresu oddzielonych znakiem /.
- Jeden poziom adresu URL można zastąpić dwukropkiem „:”, po którym następuje nazwa parametru.
  - Parametr o podanej nazwie pojawi się w kolekcji msg.req.params.
  - Punkt końcowy o adresie URL /path1/:param1/: param2/path4 będzie obsługiwał żądania o adresach czteropoziomowych, zaczynających się od /path1 i kończących /path4, z dowolnymi łańcuchami tekstu wpisanymi na poziomach 2. i 3.
  - Teksty wpisane na tych poziomach będą dostępne w kolekcji msg. req.params jako parametry o nazwach param1 i param2.

Projekt „SezAM wiedzy, kompetencji i umiejętności” jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Menedżer internetowych usług inf. (IIS)



Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

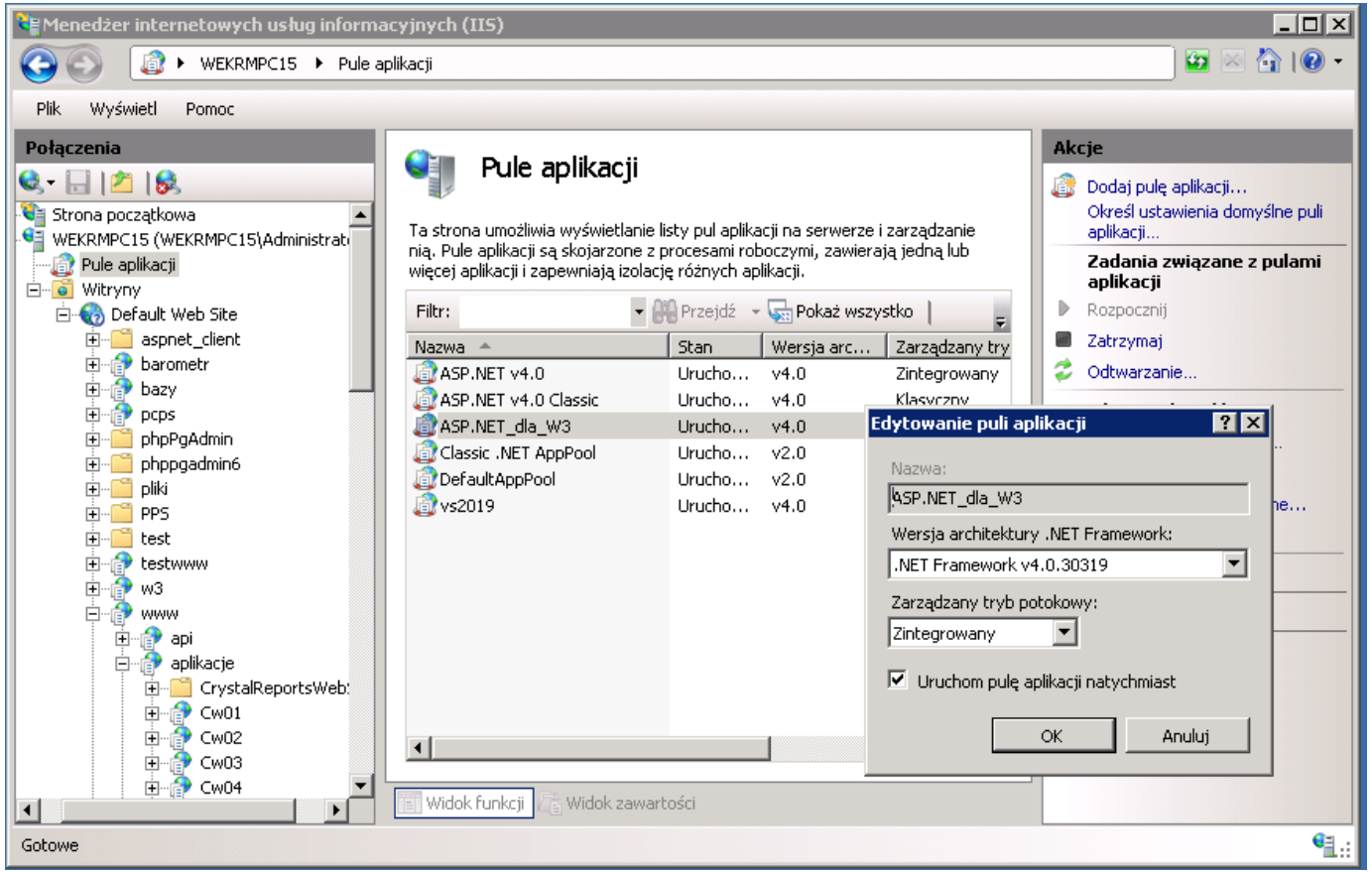
Katalog wirtualny i aplikacja IIS

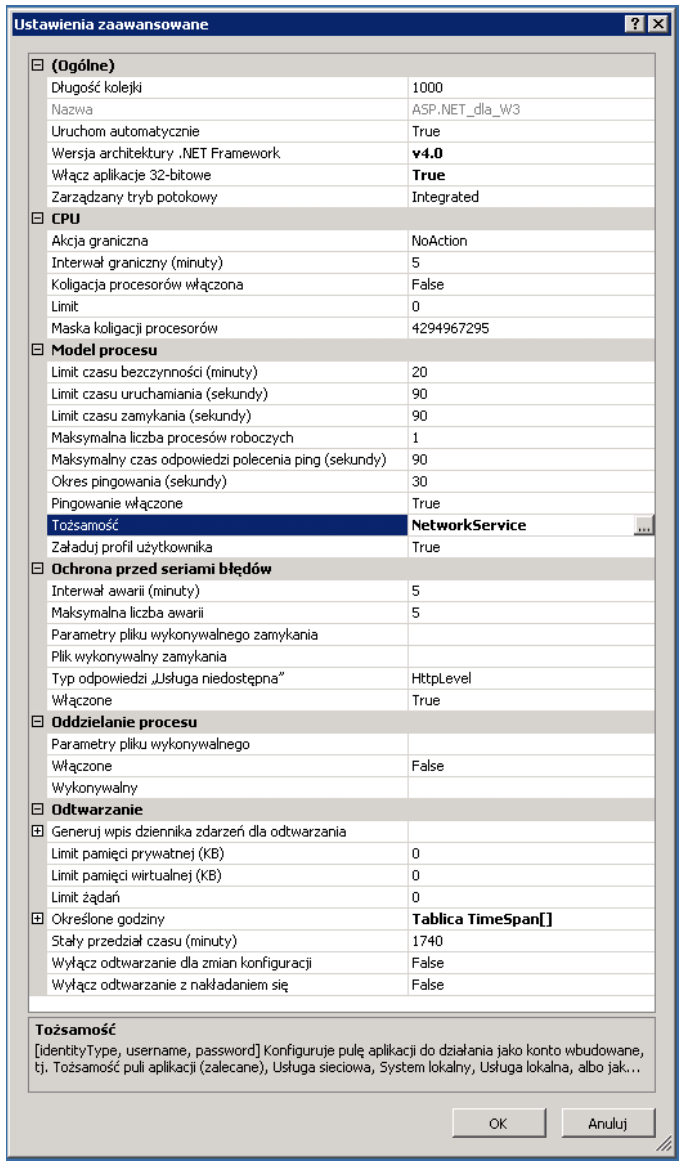
- Katalog wirtualny – dowolny katalog serwera, który jest udostępniony jako cel dla żądań napływających z sieci.
- Domyślny witryna sieci Web – `C:\inetpub\wwwroot\`.
- Aplikacja sieciowa składa się ze wszystkich stron internetowych, plików, kodu, obiektów, plików wykonywalnych, elementów graficznych i innych zasobów, które zostały umieszczone w katalogu wirtualnym serwera IIS lub podkatalogu tego katalogu wirtualnego (inaczej aplikacja sieciowa jest witryną internetową albo usługą sieciową).
- Każdy katalog wirtualny jest oddzielną aplikacją, każda aplikacja obsługiwana przez IIS musi posiadać wirtualny katalog główny.
- Opcje katalogu wirtualnego
  - strona domyślna,
  - przeglądanie katalogów.

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Administrowanie serwerem IIS

Pule aplikacji: tożsamość, uprawnienia





Projekt „SezAM wiedzy, kompetencji i umiejętności” jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Strona internetowa (WWW)

- Dokument tekstowy, który może być zinterpretowany i wyświetlony przez przeglądarkę.
- Strony internetowe są najczęściej tworzone w języku HTML (ang. *HyperText Markup Language*).
- Aktywna (dynamiczna) strona internetowa to strona tworzona w części lub w całości przez program działający na serwerze internetowym.
- Do tworzenia stron aktywnych wykorzystywane są technologie ASP, ASP.NET, JSP, PHP, … .

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Skrypty ASP na stronie

- Na stronach aktywnych ASPX można umieszczać kod uruchamiany przed wysłaniem strony do przeglądarki tworzący część lub całość treści strony.
- Kod generujący

```
<%
    Response.Write("Ala ma kota <br />");
%>
```

- Wyrażenie wbudowane

```
<% = "Kot ma Alę <br />" %>
<% =DateTime.Now.ToLongDateString() %>
```

- Posługując się skryptami można stworzyć każdą aplikację internetową.

Projekt „SezAM wiedzy, kompetencji i umiejętności” jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Wydruk parametrów żądania

```
<%
// Wydruk parametrów żądania
for (int i = 0; i < Request.Params.Count; i++)
{
    Response.Write("Indeks : " + i.ToString() + "<br />");
    Response.Write("Nazwa  : " + Request.Params.GetKey(i) + "<br />");
    Response.Write("Ilość  : " + Request.Params.GetValues(i).Length.ToString() + "<br />");

    for (int j = 0; j < Request.Params.GetValues(i).Length; j++)
        Response.Write("Wartość:" + Request.Params.GetValues(i)[j] + "<br /><br />");
}
%>
```

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Wydruk nagłówków żądania

```
  int loop1, loop2;
NameValueCollection coll;

// Load Header collection into NameValueCollection object.
coll=Request.Headers;

// Put the names of all keys into a string array.
String[] arr1 = coll.AllKeys;
for (loop1 = 0; loop1 < arr1.Length; loop1++)
{
    Response.Write("Key: " + arr1[loop1] + "<br>");
    // Get all values under this key.
    String[] arr2 = coll.GetValues(arr1[loop1]);
    for (loop2 = 0; loop2 < arr2.Length; loop2++)
    {
        Response.Write("Value " + loop2 + ": " + Server.HtmlEncode(arr2[loop2]) + "<br>");
    }
}
```

Microsoft help

Przykład czytania całego żądania w przykładzie TestHttpListener

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Klasa Page

- Klasa Page reprezentuje stronę internetową.
- Posługując się właściwościami klasy Page można stworzyć każdą aplikację internetową, daje to dużo satysfakcji, ale jest, nie tyle trudne, co bardzo pracochłonne!
- Ważne właściwości klasy Page
  - `Response`
  - `Request`
  - `IsPostBack`

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Właściwość Response

- Zwraca obiekt `HttpResponse`, który reprezentuje odpowiedź HTTP na bieżące żądanie
- Ważne właściwości klasy `HttpResponse`
  - `Cookies` – kolekcja ciasteczek wysyłanych do przeglądarki
- Ważne metody klasy `HttpResponse`
  - `Write` – zapisuje dane do strumienia HTTP
  - `Redirect` – przełącza żądanie do innej strony

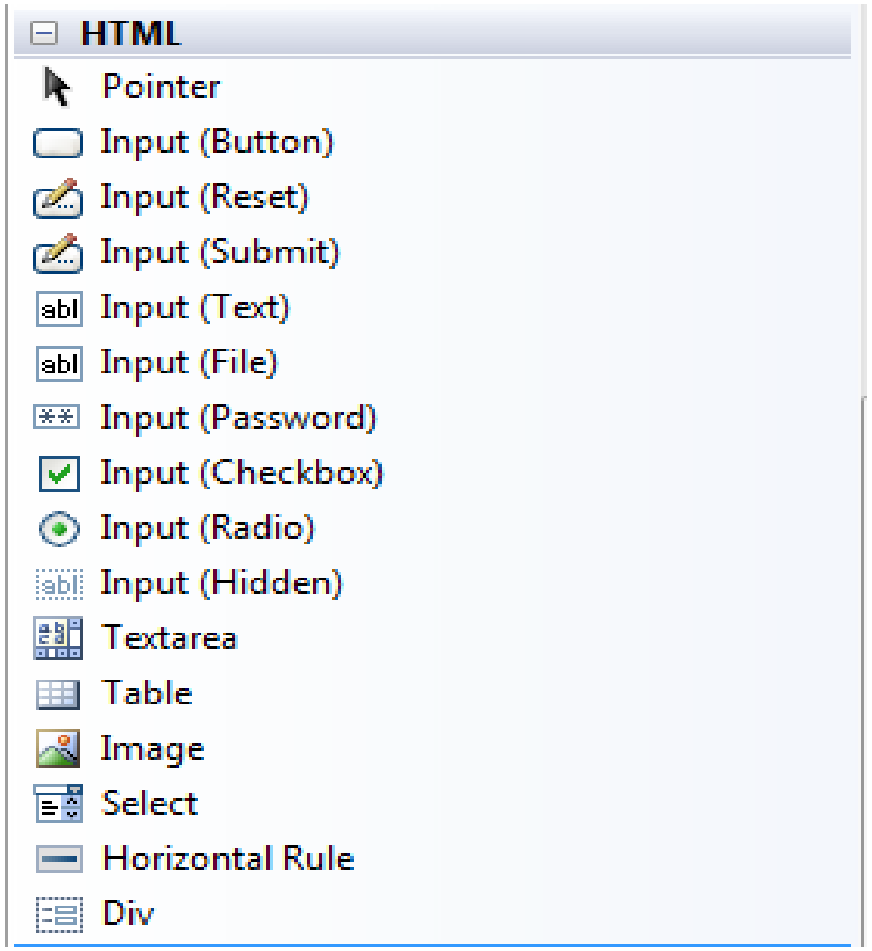
Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Właściwość Request

- Zwraca obiekt `HttpRequest`, który reprezentuje bieżące żądanie HTTP wysłane przez przeglądarkę
- Ważne właściwości klasy `HttpRequest`
  - `Cookies` – zwraca kolekcję ciasteczek odesłanych przez przeglądarkę,
  - `Form` – zwraca kolekcję danych formularza wysłanych z żądaniem,
  - `QueryString` – zwraca kolekcję danych wysłanych w ciągu zapytania razem z żądaniem,
  - `ServerVariebles` – zwraca kolekcję nazwanych danych serwera wysłanych z żądaniem,
  - `Params` – zwraca łączną kolekcję `Cookies`, `Form`, `QueryString` i `ServerVariebles`,
  - `FilePath` – zwraca wirtualną ścieżkę do bieżącej strony,
  - `PhisicalPath` – zwraca fizyczną ścieżkę do bieżącej strony,
  - `ApplicationPath`,
  - `Phisical ApplicationPath`.

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Formularze i kontrolki HTML `input`



- Kontrolki `Input` służą do umieszczenia przez klienta danych na stronie w celu przesłania ich do aplikacji działającej na serwerze.
- Dane formularza domyślnie odsyła się do tej strony z której pochodzą, ale można je przesłać do dowolnej strony, adres strony określa atrybut action.

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Własności kontrolek HTML

- Kontrolka HTML nie zachowuje stanu po przeładowaniu strony i nie jest widoczna przez skrypty działające po stronie serwera.
- Kod zachowujący stan kontrolki może mieć postać

```
<input id="Text1" type="text" name="x1"
<%  if (Request.Params.GetValues("x1") == null)
          Response.Write("value=\"0\"");
     else Response.Write("value=\"" + Request.Params["x1"].ToString()+"\""); %> />
```

- lub

```
<input id="Text2" type="text" name="x2" value= "
<%
if (Request.Params["x2"] == null) Response.Write("0");
else Response.Write(Request.Params["x2"].ToString());  %>"
/>
```

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Zdarzenia

- Strona jest odsyłana na serwer po kliknięciu przycisku typu `Submit`.
- Co należy zrobić ze stroną serwer musi rozpoznać na podstawie parametrów przesłanych wraz z żądaniem.
- Na stronie (w formularzu) można umieścić wiele przycisków typu `Submit`.
- Jeśli przycisk odsyłający stronę jest nazwany, to do listy parametrów zostanie dodany parametr o nazwie przycisku i wartości równej napisowi na przycisku (atrybut `value`).
- Przyciski mogą mieć te same nazwy, do żądania dołączane są tylko dane przycisku wywołującego odesłanie strony.

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Rozpoznawanie, który przycisk odesłał stronę

- Gdy przyciski mają unikalne nazwy
  - sprawdzamy czy nazwa przycisku jest wśród nazw parametrów żądania:
  - `if (Request.Params ["nazwa"] != null)`
- Gdy przyciski mają identyczne nazwy – na podstawie napisu (atrybutu `value`).
  - sprawdzamy wartość parametru o nazwie zgodnej z nazwą przycisków
  - `if (Request.Params["nazwa"] == "+")`

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Rozpoznawanie innych zdarzeń

- Na podstawie przesłanego żądania.
- Wykrycie zmiany tekstu w kontrolce tekstowej,
  - Np. na podstawie porównania odesłanej wartości z wartością początkową.
  - Do porównania wymagany jest zapisu stanu wysyłanej strony.
  - Zapisu stanu strony (kontrolek) nożna dokonać za pomocą ukrytych kontrolek umieszczonych na stronie.

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Atrybut `runat`

- Kontrolka HTML np.
  `<input id="T2" type="text" />`
  nie zachowuje stanu po przeładowaniu strony i nie jest widoczna przez skrypty działające po stronie serwera.
- Po dodaniu do kontrolki HTML atrybutu `runat=server` np.
  `<input id="T2" type="text" runat=server />`
  kontrolka staje się kontrolką serwerową i zachowuje swój stan po przeładowaniu strony oraz jest widoczna przez skrypty działające po stronie serwera – skrypty mogą czytać i zmieniać jej atrybuty.

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Właściwości kontrolek serwerowych

- Kontrolkom serwerowym nie trzeba nadawać nazw, zrobi to automatycznie serwer przed wysłaniem strony do przeglądarki.
- Jeśli kontrolkom serwerowym nadane zostaną nazwy, serwer je zignoruje i nada własne nazwy identyczne z wartością atrybutu `id`.
- W kodzie skryptu kontrolka jest identyfikowana przez atrybut `id`.
- ```
  <input id="T2" type="text" runat=server />
  ```
  - `T2.value = "Ala ma kota";`
  - `T2.Style["color"]="#FF0000";`
  - `Y = T2.value + "Kot ma Alę";`

Projekt „SezAM wiedzy, kompetencji i umiejętności” jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój
  ```

Deklaracje kontrolek ASP.NET

```
<asp:nazwaKontrolki id="identyfikator" runat="server" «inne atrybyty» >
</asp:nazwaKontrolki>
```

- Przykłady
  - Kontrolka tekstowa
    ```
    <asp:TextBox ID="TextBox1" runat="server">TEKST</asp:TextBox>
    ```
  - Przycisk
    ```
    <asp:Button ID="Button1" runat="server" Text="Button" />
    ```
  - Etykieta
    ```
    <asp:Label ID="Label1" runat="server" Text="Label">Napis<br />
    ```

- Przed wysłaniem strony serwer zamienia kontrolki ASP.NET na kontrolki HTML zrozumiałe dla przeglądarki.
- Oryginalna strona ASPX, nieprzetworzona przez serwer, jest niezrozumiała dla przeglądarki.

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Atrybuty kontrolek ASP.NET

- BackColor
- BorderColor
- BorderWidth
- Enabled
- Font
- Height
- TabIndex
- Width
- ID – unikalny identyfikator
- Controls – kolekcja kontrolek potomnych
- Parent – odwołanie do kontrolki nadrzędnej
- Visible

Projekt „SezAM wiedzy, kompetencji i umiejętności” jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Dostęp do atrybutów kontrolek z poziomu kodu C#

- W kodzie C# kontrolki ASP.NET są widoczne jak obiekty o identyfikatorach zgodnych z atrybutem ID kontrolki.
- Z poziomu kodu można czytać wartości atrybutów kontrolek jako właściwości odpowiadających im obiektów, można też zmieniać wartości atrybutów
  - `TextBox3.Text = TextBox1.Text + TextBox2.Text;`
  - `Label1.Text = "Ala ma kota";`
- Do kontrolek typu paragraf `<p> </p>` należy dodać atrybuty `id` i `runat="server"`

Projekt „SezAM wiedzy, kompetencji i umiejętności” jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Deklaracja zdarzeń

- Atrybuty zdarzeń
- <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="Button" />
  - onclick – nazwa zdarzenia
  - Button1_Click – nazwa metody obsługi zdarzenia

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Zdarzenia

- Zdarzenia związane z cyklem życia strony
  - OnInit, OnLoad, OnPreRender, OnUnload.
- Zdarzenia PostBack
  - Onclick.
- Zdarzenia Change
  - OnTextChanged, SelectedIndexChanged.
- Atrybut AutoPostBack

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Tabela

- Kontrolka Table

```
<asp:Table ID="Table1" runat="server">
        kolekcja wierszy
    </asp:Table>
```

- Kontrolka TableRow

```
<asp:TableRow runat="server">
        kolekcja komórek
    </asp:TableRow>
```

- Kontrolka TableCell

```
<asp:TableCell runat="server">Tekst celi</asp:TableCell>
```

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Przykład tabeli

```
<asp:Table ID="Table1" runat="server">
  <asp:TableRow runat="server">
        <asp:TableCell runat="server">cela1</asp:TableCell>
        <asp:TableCell runat="server">cela2</asp:TableCell>
   </asp:TableRow>
</asp:Table>
```

W przykładzie tabela ma identyfikator a wiersz i komórki nie. Do komórek jest dostęp programowy poprzez kolekcje wierszy Rows i kolekcje komórek Cels

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Dynamiczne tworzenie tabeli

```
for (i = 0; i < lw; i++)
        {
            TableRow y = new TableRow();
            for (k = 0; k < lk; k++)
            {
                TableCell x = new TableCell();
                y.Cells.Add(x);
                x.Controls.Add(new TextBox());
            }
            Table1.Rows.Add(y);
        }
```

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Programowy dostęp do komórek tabeli

```
for(int i = 0; i < Table1.Rows.Count; i++)
    for (int k = 0; k < Table1.Rows[i].Cells.Count; k++)
        Table1.Rows[i].Cells[k].Text = "Ala ma kota";

for (int i = 0; i < Table1.Rows.Count; i++)
    for (int k = 0; k < Table1.Rows[i].Cells.Count; k++)
      ((TextBox)Table1.Rows[i].Cells[k].Controls[0]).Text = "Ala ma kota";
```

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Cykl życia strony

- Sprawdzenie trybu PostBack.
- Wstępna inicjalizacja Preinit (OnPreInit).
- Inicjalizacja Init (OnInit).
- Wstępne wczytanie strony PreLoad (On PreLoad).
- Wczytanie strony Load (OnLoad)
  - Uruchomienie kodu użytkownika,
  - Kontrolki danych pokazują dane po stronie klienta,
  - Dostępny stan widoku,
  - Możliwy dostęp do kontrolek strony.
- Ukończenie wczytania strony LoadComplite (OnLoadComplite).
- Wstępne generowanie strony PreRender (On PreRender).
- Generowanie strony
  - Strona i kontrolki wygenerowane jako kod HTML.
- Usunięcie strony z pamięci Unload (OnUnload).

Projekt „SezAM wiedzy, kompetencji i umiejętności” jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Metoda Page_Load

- Zdarzenie On_Page_Load zachodzi przed zdarzeniami typu Change i PostBack
  - jeśli tworzymy dynamicznie kontrolki, to wygodnie kod ich tworzenia umieścić w tej metodzie.
- Zdarzenie On_Page_Load zachodzi podczas każdego przeładowania strony
  - jeśli w kodzie metody Page_Load tworzy się kontrolki, to będą widoczne dla kodu obsługi innych zdarzeń, które zachodzą później.

Projekt „SezAM wiedzy, kompetencji i umiejętności" jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój