

Aplikacje WWW - laboratorium

ASP.NET.

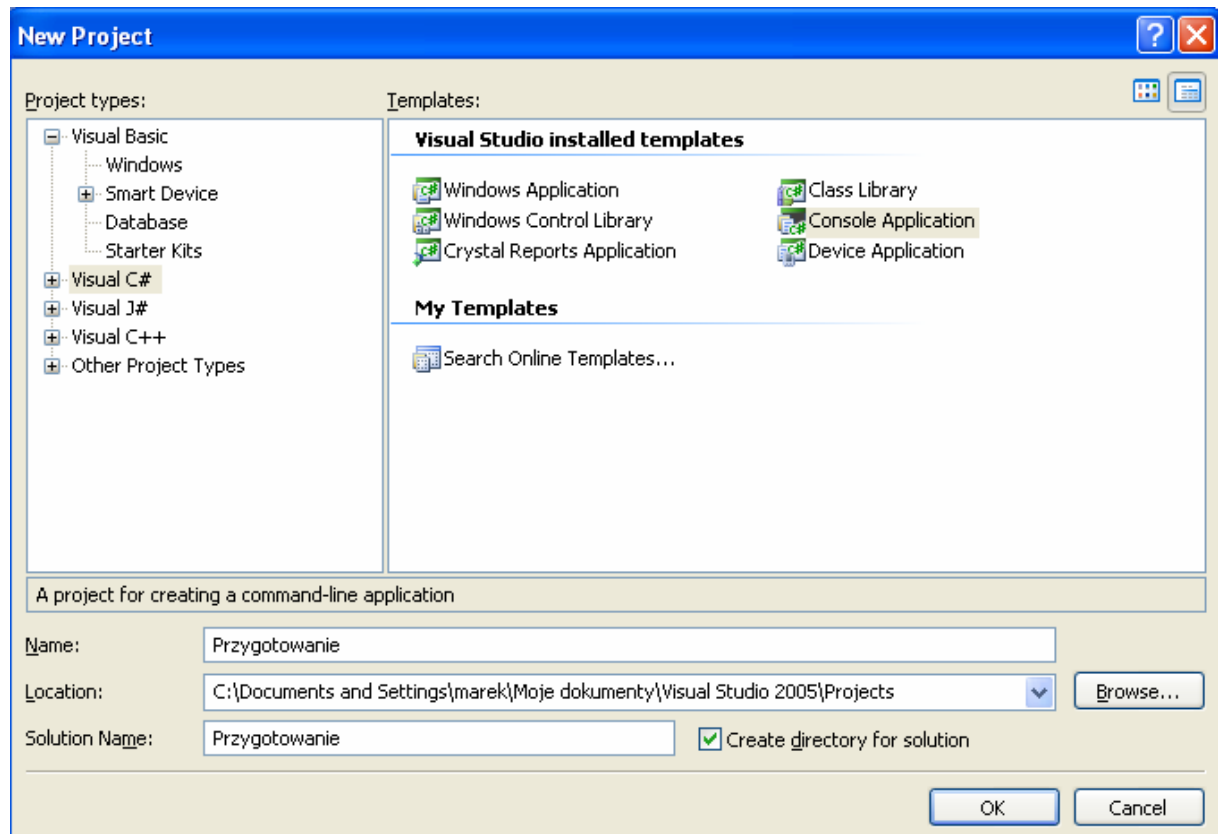
Do wykonania ćwiczeń potrzebne jest zintegrowane środowisko programistyczne Microsoft Visual Studio 2005 oraz dostęp do serwera Microsoft SQL Server 2005. Treść ćwiczenia zakłada wykorzystanie instancji serwera bazy danych Microsoft SQL Server 2005 Express Edition, instalowanej razem ze środowiskiem Visual Studio 2005. Visual Studio 2005 jest udostępniane studentom w ramach programu MSDN Academic Alliance (<http://www.microsoft.com/poland/edukacja/uczelnie/default.msp>). W celu uzyskania pakietu instalacyjnego Microsoft Visual Studio 2005 należy skontaktować się z prowadzącym zajęcia.

Przygotowanie bazy danych

W ćwiczeniu będzie wykorzystana tabela PRACOWNICY zawierająca kolumny ID_PRAC, NAZWISKO, IMIE, ETAT i PLACA_POD. Wymagana tabela zostanie utworzona w bazie danych tempdb instancji SQLEXPRESS serwera Microsoft SQL Server 2005 Express Edition, instalowanej razem ze środowiskiem Visual Studio 2005. Utworzenie i wypełnienie tabeli danymi zostanie zrealizowane za pomocą prostego programu w języku C#, uruchamianego z linii poleceń.

Kroki ćwiczenia:

1. Utworzenie projektu typu Console Application w języku Visual C#.ol style="list-style-type: none;">- a) Uruchom narzędzie Microsoft Visual Studio 2005.
- b) Z menu głównego wybierz File→New Project. Wybierz typ projektu Visual C# i szablon Console Application. Jako nazwę projektu podaj "Przygotowanie". Kliknij przycisk **OK**.



- c) W pliku Program.cs dodaj dwie poniższe dyrektywy using, importujące przestrzeń nazw dla klas wykorzystywanych w komunikacji z bazą danych SQL Server:

```
using System.Data;
using System.Data.SqlClient;
```

- d) Jako treść metody Main klasy Program wklej poniższy kod (każdy z łańcuchów znaków powinien w całości być zawarty w pojedynczym wierszu). Jeśli wykorzystujesz inną instancję serwera SQL Server niż instalowaną razem z Visual Studio 2005, zmodyfikuj odpowiednio łańcuch połączenia przekazany jako parametr konstruktora SqlConnection.

```
SqlConnection prConn = new SqlConnection("Data
Source=localhost\\SQLEXPRESS;Integrated Security=True;Initial
Catalog=tempdb");
prConn.Open();
SqlCommand cmd = prConn.CreateCommand();
cmd.CommandText = "CREATE TABLE PRACOWNICY (ID_PRAC NUMERIC(4) PRIMARY
KEY, NAZWISKO VARCHAR(15), IMIE VARCHAR(15), ETAT VARCHAR(10), PLACA_POD
NUMERIC(6,2))";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES (100,'Marecki','Jan'
,'DYREKTOR',4730.00)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES (110,'Janicki','Karol'
,'PROFESOR',3350.00)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES (120,'Nowicki','Pawel'
,'PROFESOR',3070.00)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES (130,'Nowak','Piotr'
,'PROFESOR',3960.00)";
```

```

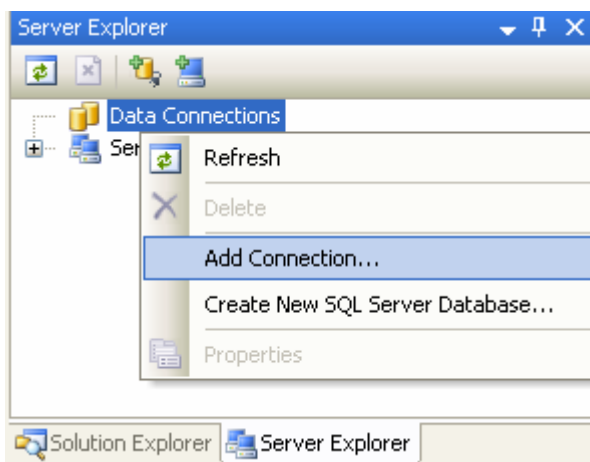
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES
(140,'Kowalski','Krzysztof','PROFESOR', 3230.00)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES
(150,'Grzybowska','Maria','ADIUNKT', 2845.50)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES
(160,'Krakowska','Joanna','SEKRETARKA', 1590.00)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES (170,'Opolski','Roman'
,'ASYSTENT', 1839.70)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES (190,'Kotarski','Konrad',
,'ASYSTENT', 1971.00)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES (180,'Makowski','Marek',
,'ADIUNKT', 2610.20)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES (200,'Przywarek','Leon'
,'DOKTORANT', 900.00)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES
(210,'Kotlarczyk','Stefan','DOKTORANT', 900.00)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES (220,'Siekierski',
,'Mateusz','ASYSTENT', 1889.00)";
cmd.ExecuteNonQuery();
cmd.CommandText = "INSERT INTO PRACOWNICY VALUES (230,'Dolny','Tomasz'
,'ASYSTENT', 1850.00)";
cmd.ExecuteNonQuery();
prConn.Close();

```

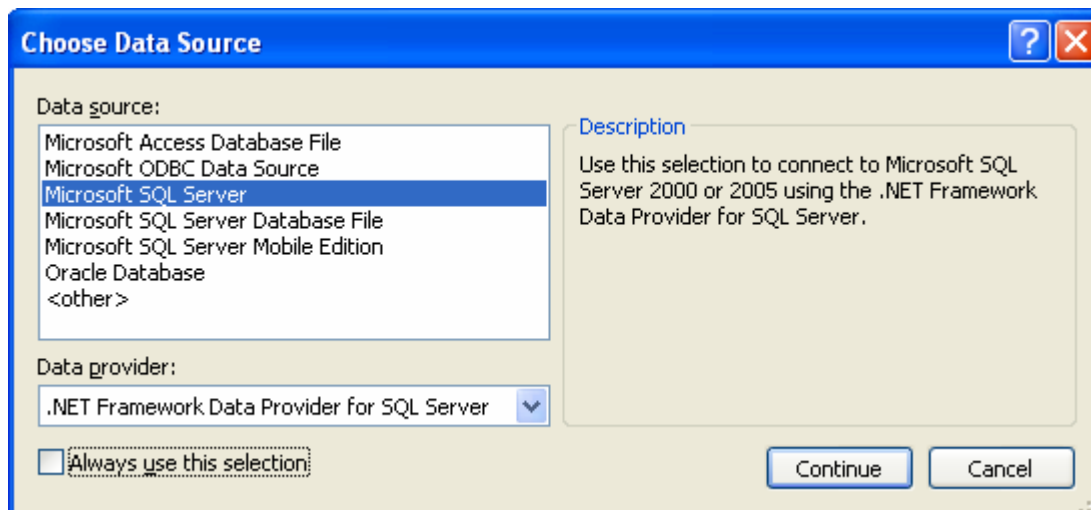
e) Uruchom aplikację (Debug→Start Debugging lub Debug → Start Without Debugging).

2. Sprawdzenie poprawności utworzenia tabeli i wypełnienia jej danymi.

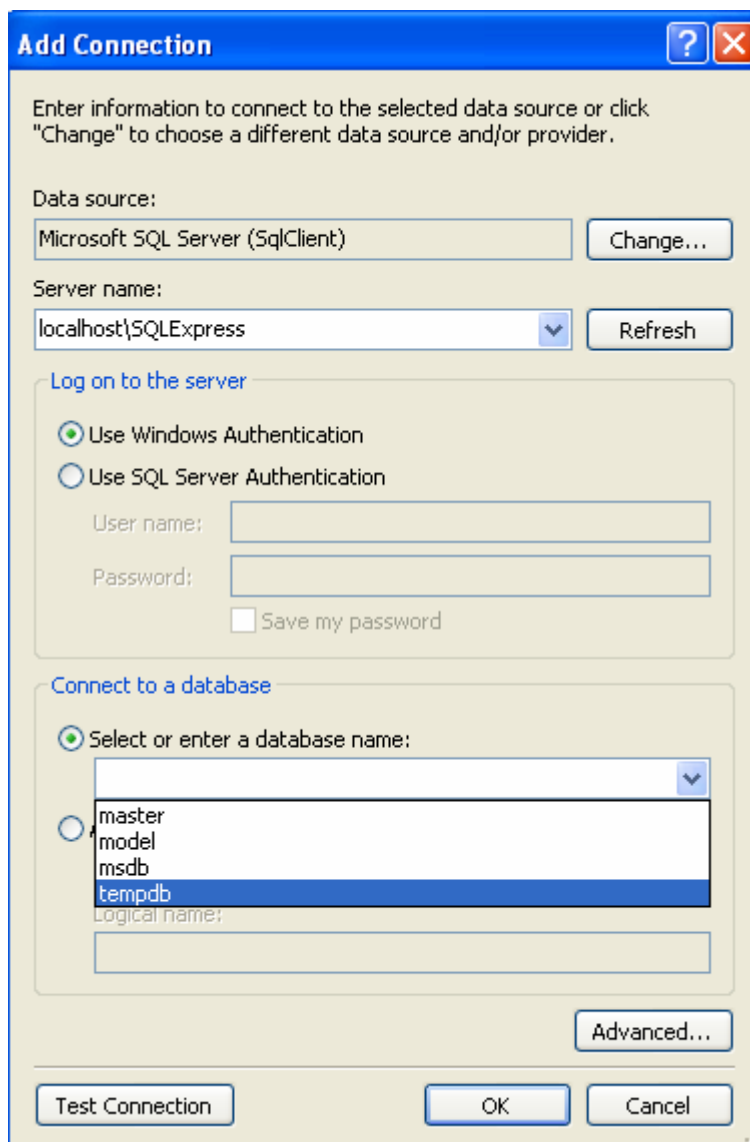
a) W panelu Server Explorer wywołaj prawym klawiszem myszy menu kontekstowe dla węzła Data Connections i wybierz opcję Add Connection.



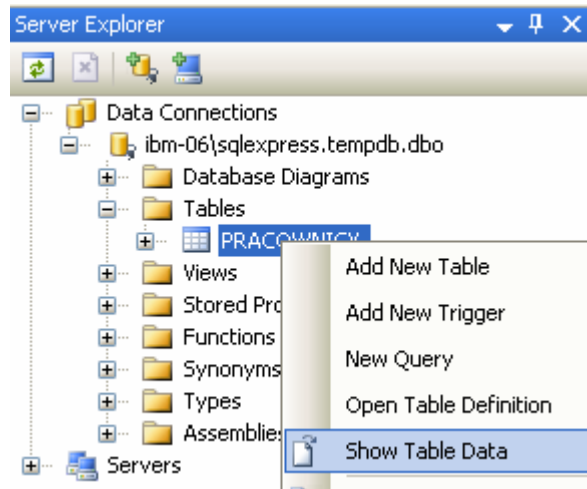
b) Jako źródło danych wybierz Microsoft SQL Server, a jako dostawcę danych .NET Framework Data Provider for SQL Server. Kliknij przycisk **Continue**.



- c) Jako nazwę serwera wpisz ręcznie „localhost\SQLEXPRESS”. Następnie z listy rozwijanej wybierz bazę danych „tempdb”. Kliknij przycisk **OK**.



- d) W panelu Server Explorer rozwiń węzeł reprezentujący utworzone połączenie, następnie rozwiń węzeł Tables i dla tabeli PRACOWNICY wywołaj prawym klawiszem myszy menu kontekstowe i wybierz z niego opcję Show Table Data.



	ID_PRAC	NAZWISKO	IMIE	ETAT	PLACA_POD
▶	100	Marecki	Jan	DYREKTOR	4730,00
	110	Janicki	Karol	PROFESOR	3350,00
	120	Nowicki	Pawel	PROFESOR	3070,00
	130	Nowak	Piotr	PROFESOR	3960,00
	140	Kowalski	Krzysztof	PROFESOR	3230,00
	150	Grzybowska	Maria	ADIUNKT	2845,50
	160	Krakowska	Joanna	SEKRETARKA	1590,00
	170	Opolski	Roman	ASYSTENT	1839,70
	180	Makowski	Marek	ADIUNKT	2610,20
	190	Kotarski	Konrad	ASYSTENT	1971,00
	200	Przywarek	Leon	DOKTORANT	900,00
	210	Kotlarczyk	Stefan	DOKTORANT	900,00
	220	Siekierski	Mateusz	ASYSTENT	1889,00
	230	Dolny	Tomasz	ASYSTENT	1850,00
*	NULL	NULL	NULL	NULL	NULL

- e) Zamknij projekt (File→Close Project) i okno przedstawiające zawartość tabeli PRACOWNICY (File→Close).

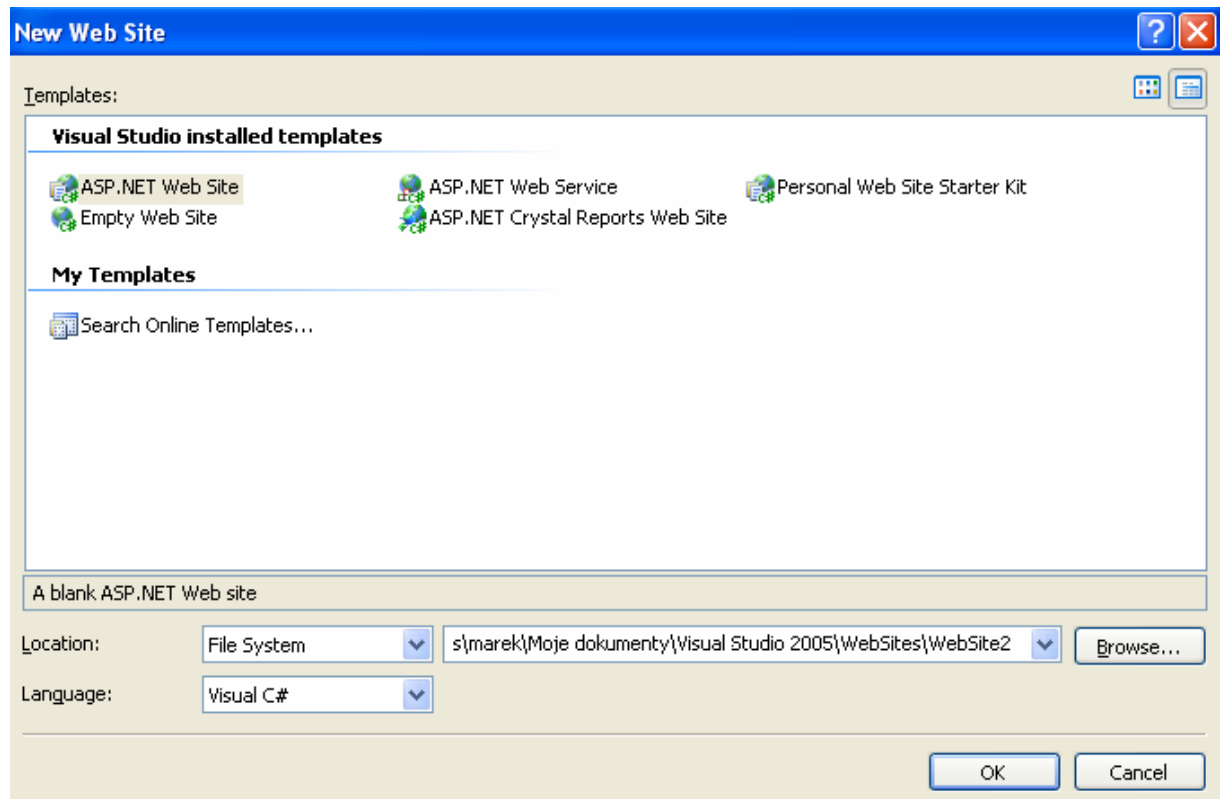
Ćwiczenie

Celem ćwiczenia jest utworzenie aplikacji internetowej umożliwiającej przeglądanie informacji o wybranych pracownikach. Aplikacja będzie obejmowała dwie strony ASP.NET. Pierwsza strona będzie służyła do wyboru etatu i progów płacy. Druga strona zaprezentuje pobrane z bazy danych informacje o pracownikach spełniających wprowadzone kryteria selekcji.

Kroki ćwiczenia:

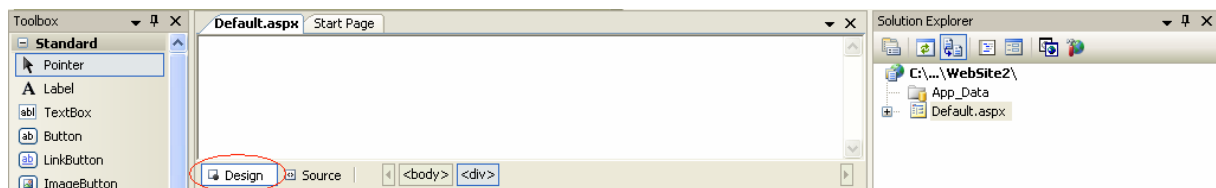
1. Utworzenie nowej witryny (Web Site).

- a) Z menu głównego wybierz File→New Web Site. Wybierz szablon ASP.NET Web Site, język Visual C# i lokalizację File System. Zaakceptuj zaproponowany katalog lub zmień go na inny gdy nie masz prawa zapisu w proponowanym katalogu. Kliknij przycisk OK.

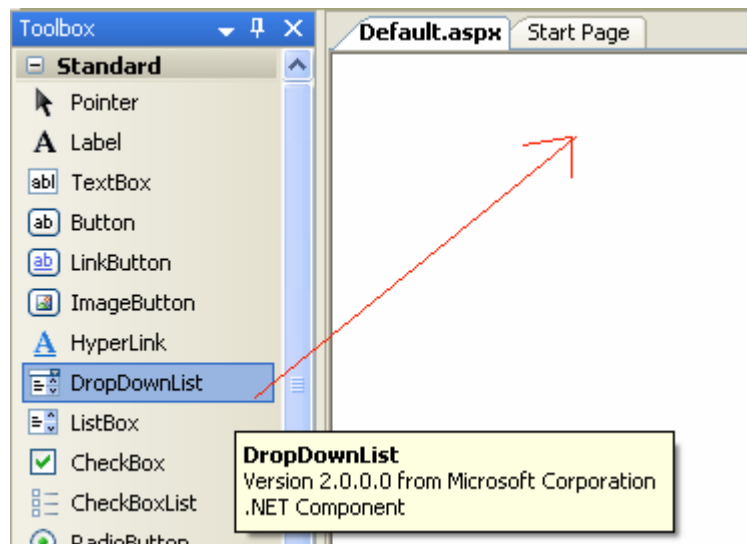


2. Utworzenie formularza do wprowadzania kryteriów selekcji pracowników.

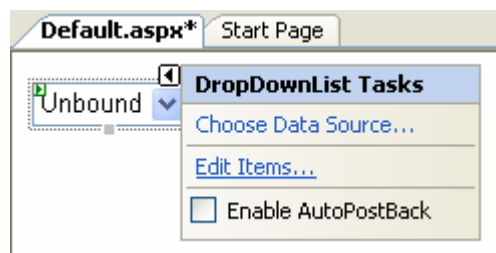
- a) Przejdź do edycji utworzonej wraz z projektem strony Default.aspx w trybie Design.



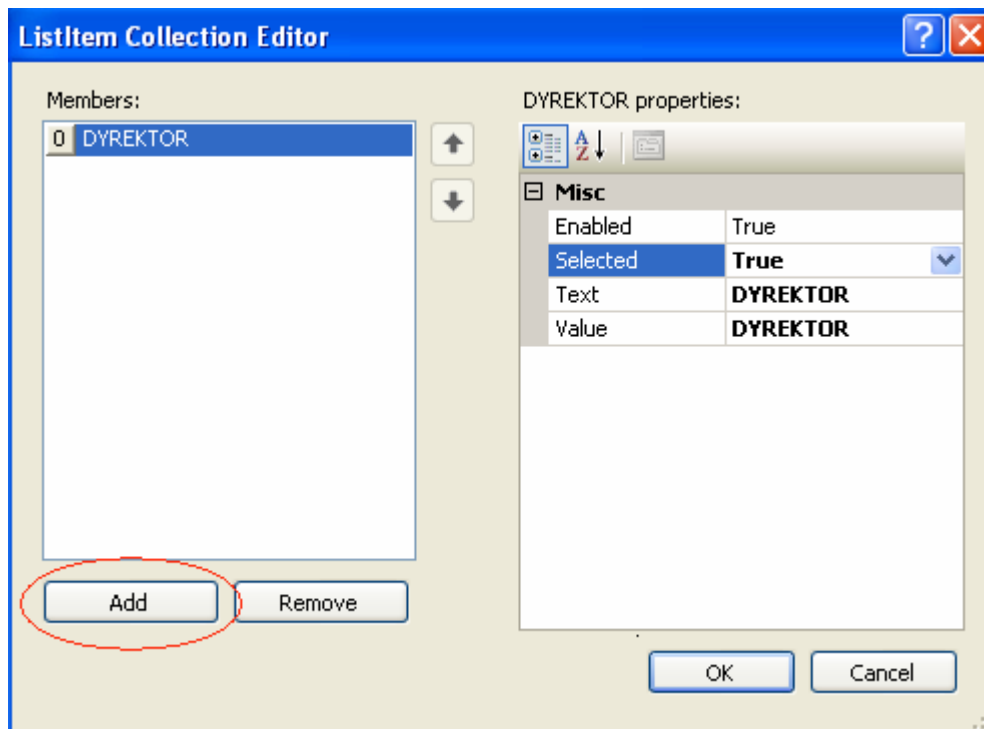
- b) Technika drag-and-drop umieść na stronie kontrolkę DropDownList z grupy Standard.



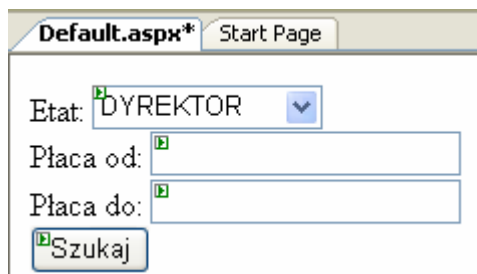
- c) Z menu, które pojawi się po umieszczeniu kontrolki na stronie wybierz opcję Edit Items. (Wskazówka: menu z typowymi zadaniami dotyczącymi kontrolki możesz wywoływać i ukrywać wybierając myszą strzałkę znajdującą się na górnej krawędzi kontrolki.)



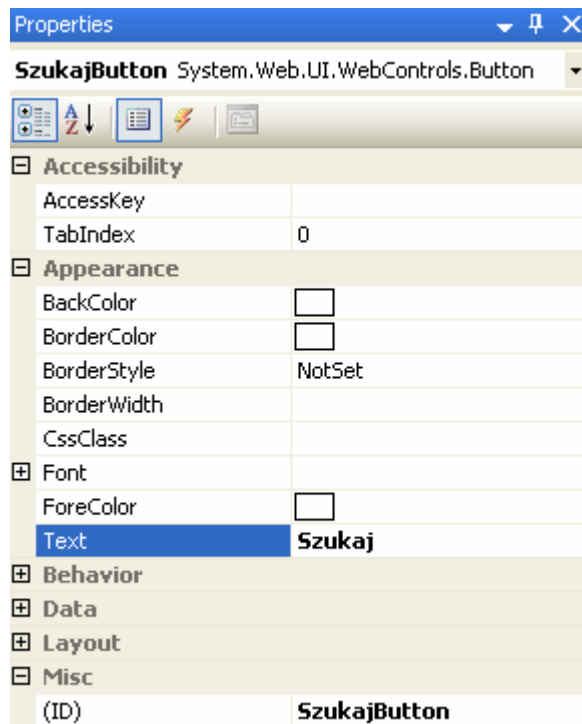
- d) Klikając przycisk Add dodaj kolejno pozycje DYREKTOR, PROFESOR, ADIUNKT, ASYSTENT, DOKTORANT i SEKRETARKA, podając każdorazowo nazwę etatu jako Text i Value. Etatem domyślnie wybranym (Selected) powinien być DYREKTOR. Po wprowadzeniu wszystkich etatów kliknij przycisk **OK**.



- e) Umieść na stronie kolejne trzy kontrolki z grupy Standard: dwie kontrolki TextBox i kontrolkę Button. Rozmieść kontrolki jedna pod drugą umieszczając wskaźnik myszy między kontrolkami i naciskając klawisz Enter. Przed każdą z kontrolek do wprowadzania danych wprowadź tekst opisujący znaczenie pola (odpowiednio: „Etat:”, „Płaca od:” i „Płaca do:”).

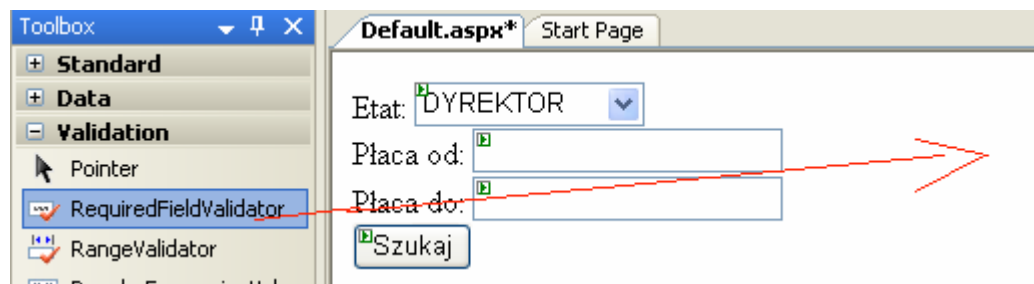


- f) Korzystając z palety właściwości (Properties) zmień identyfikatory (ID) kontrolki na „EtatDropDownList”, „DolnyTextBox”, „GornyTextBox” i „SzukajButton”. Również poprzez paletę zmień etykietę przycisku (Text) na „Szukaj”.

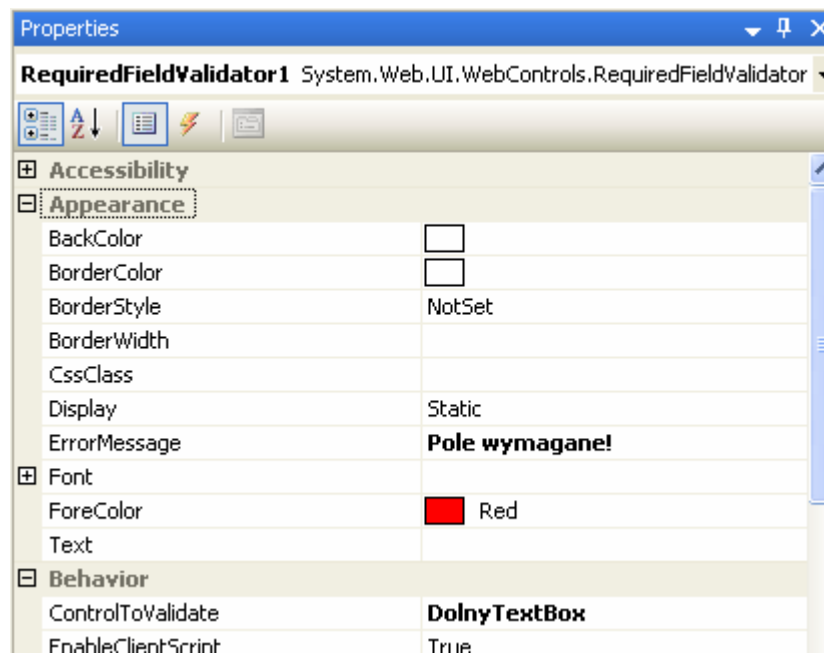


3. Rozbudowa formularza do wprowadzania kryteriów selekcji pracowników o mechanizmy walidacji danych. Wprowadzenie progów płacy ma być obowiązkowe, wartości progów powinny być liczbami całkowitymi z zakresu od 0 do 9999, przy czym próg dolny nie może być większy od górnego.

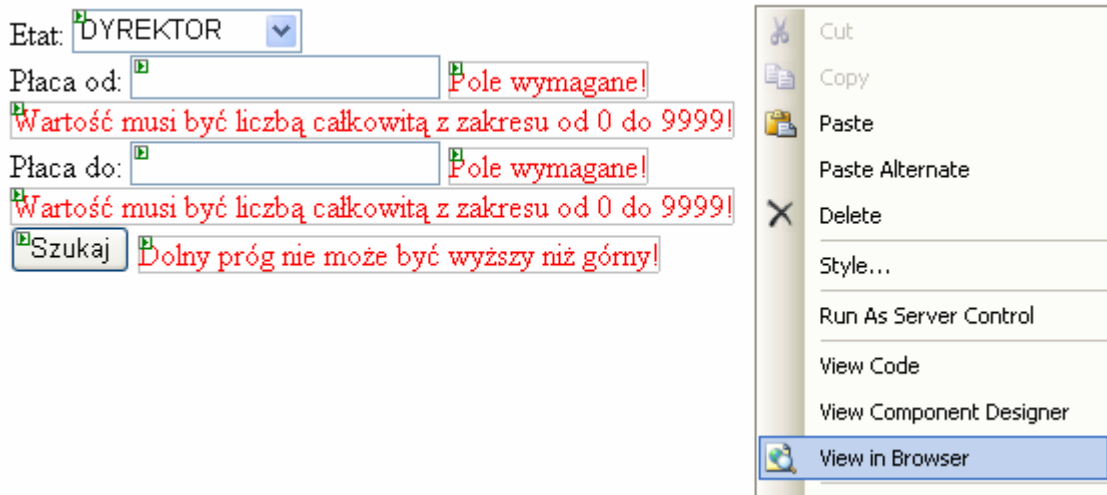
- a) Techniką drag-and-drop umieść obok kontrolki do wprowadzania dolnego progu płacy kontrolkę RequiredFieldValidator z grupy Validation.



- b) Korzystając z palety właściwości zmień komunikat o błędzie związany z walidatorem (ErrorMessage) na „Pole wymagane!”, a jako walidowaną kontrolkę (ControlToValidate) wybierz DolnyTextBox. Dla pozostałych właściwości pozostaw wartości domyślne.



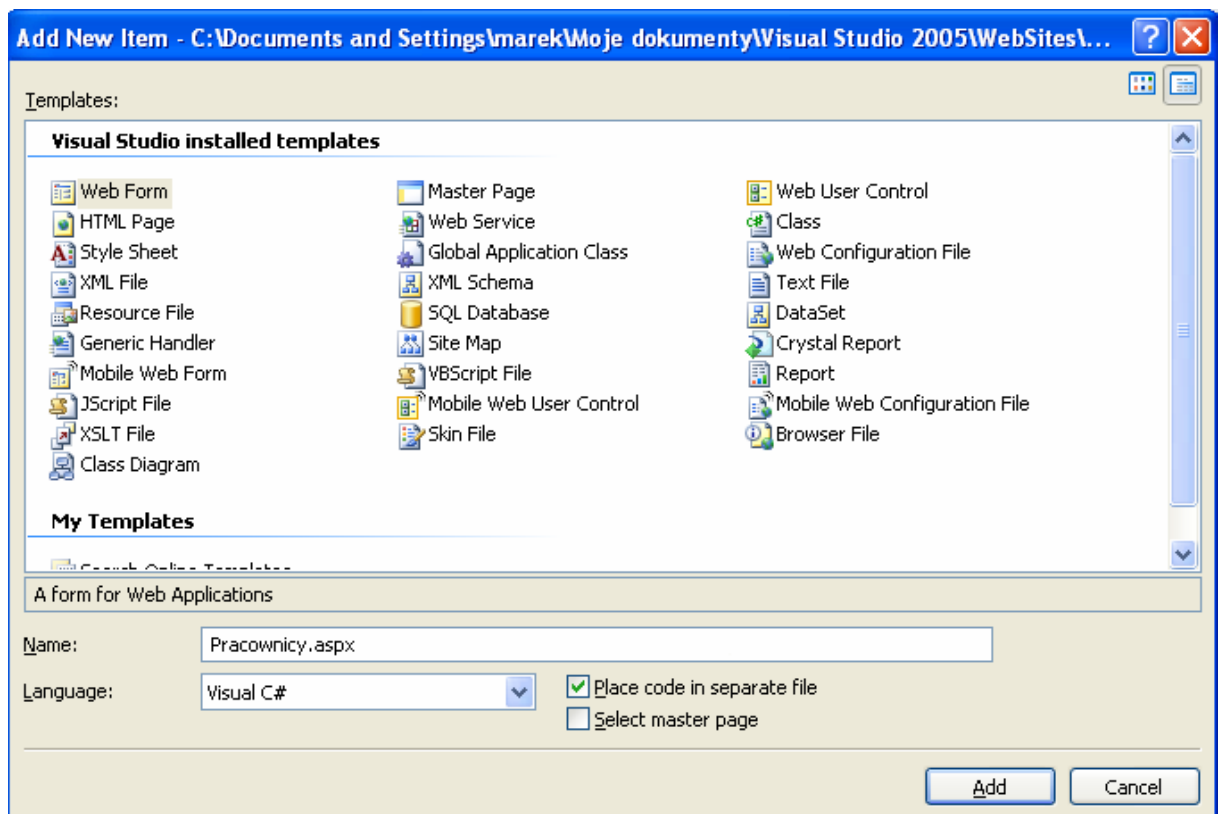
- c) Analogicznie do punktów a) i b) dodaj walidator RequiredFieldValidator dla kontrolki do wprowadzania górnego progu płacy.
- d) Technika drag-and-drop umieść obok walidatora RequiredFieldValidator dla dolnego progu płacy kontrolkę RangeValidator z grupy Validation.
- e) Korzystając z palety właściwości zmień komunikat o błędzie związany z walidatorem (ErrorMessage) na „Wartość musi być liczbą całkowitą z zakresu od 0 do 9999!”, jako walidowaną kontrolkę (ControlToValidate) wybierz DolnyTextBox, jako wartość maksymalną (MaximumValue) podaj 9999, jako wartość minimalną (MinimumValue) podaj 0, a jako typ danych (Type) wybierz Integer. Dla pozostałych właściwości pozostaw wartości domyślne.
- f) Analogicznie do punktów d) i e) dodaj walidator RangeValidator dla kontrolki do wprowadzania górnego progu płacy.
- g) Technika drag-and-drop umieść obok przycisku kontrolkę CompareValidator z grupy Validation.
- h) Korzystając z palety właściwości zmień następujące właściwości dodanego walidatora typu CompareValidator: jako komunikat o błędzie związany z walidatorem (ErrorMessage) wprowadź „Dolny próg nie może być wyższy niż górny!”, jako walidowaną kontrolkę (ControlToValidate) wybierz GornyTextBox, jako kontrolkę do porównania (ControlToCompare) wybierz DolnyTextBox, jako typ danych (Type) wybierz Integer, a jako operator porównania (Operator) wybierz GreaterThanEqual. Dla pozostałych właściwości pozostaw wartości domyślne.
- i) Zapisz wszystkie zmiany (np. File→Save All).
- j) Uruchom stronę wybierając z menu kontekstowego opcję View in Browser.



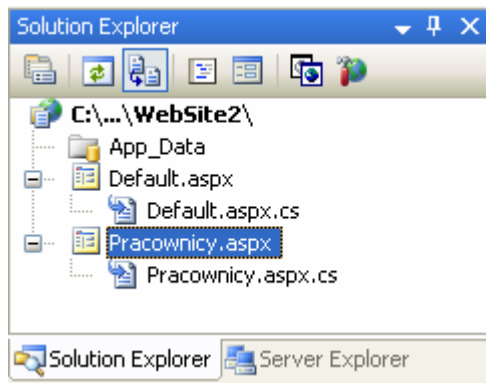
k) Przetestuj działanie walidacji na stronie wprowadzając różne kombinacje wartości dolnego i górnego progu płacy, a następnie zatwierdzając formularz poprzez kliknięcie przycisku.

4. Utworzenie strony ASP.NET prezentującej odczytane z bazy danych dane o pracownikach w formie tabelki.

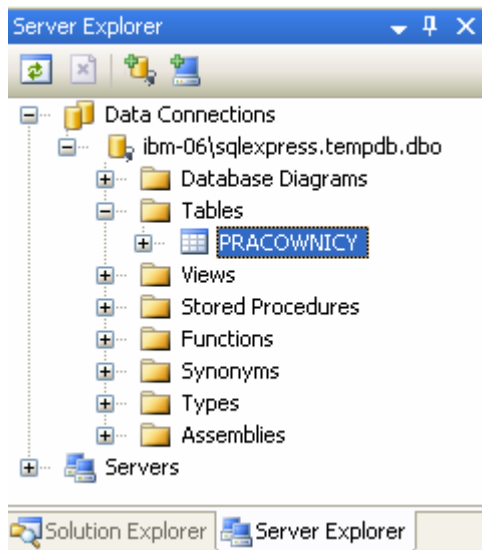
a) Utwórz w ramach bieżącej witryny nową stronę, wybierając z menu opcję File→New File, a następnie szablon Web Form. Jako jej nazwę podaj „Pracownicy.aspx”, a jako język „Visual C#”.



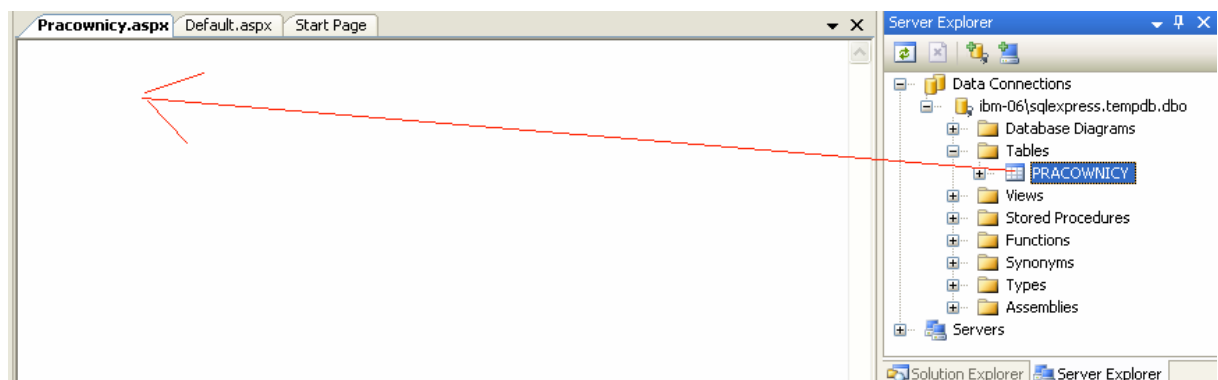
b) Sprawdź w panelu Solution Explorer, że nowa strona została dodana do witryny.



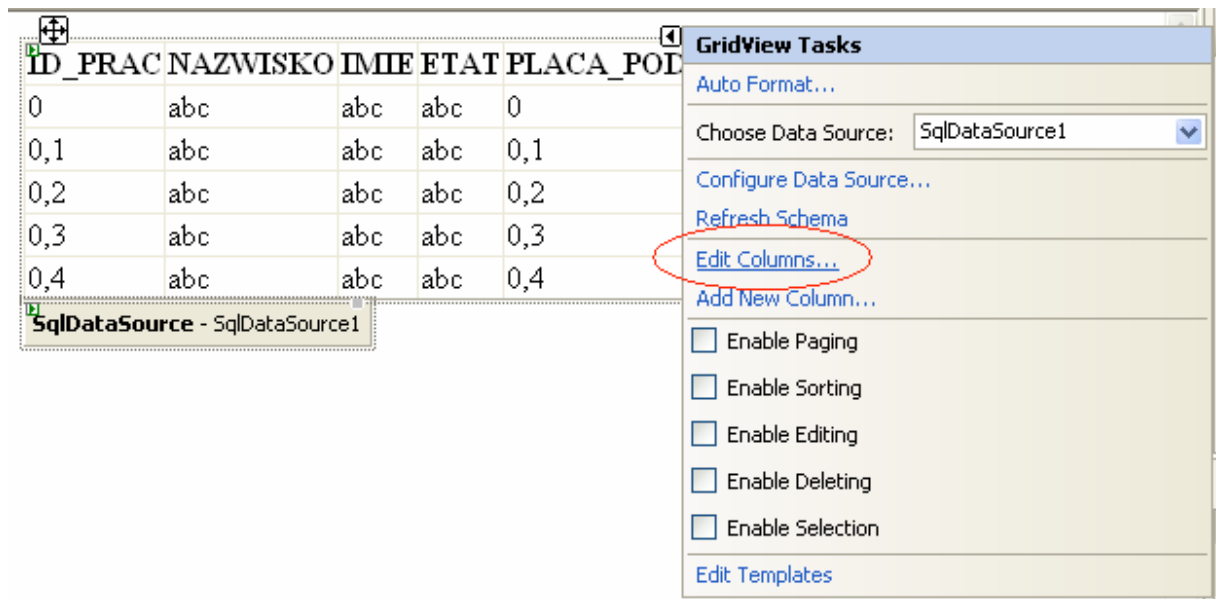
- c) Przełącz się na edycję strony Pracownicy.aspx w trybie Design.
- d) Przejdź do panelu Server Explorer. Rozwiń gałąź reprezentującą połączenie z bazą danych utworzone w drugim kroku przygotowania do ćwiczenia. Następnie rozwiń dla tego połączenia gałąź Tables.



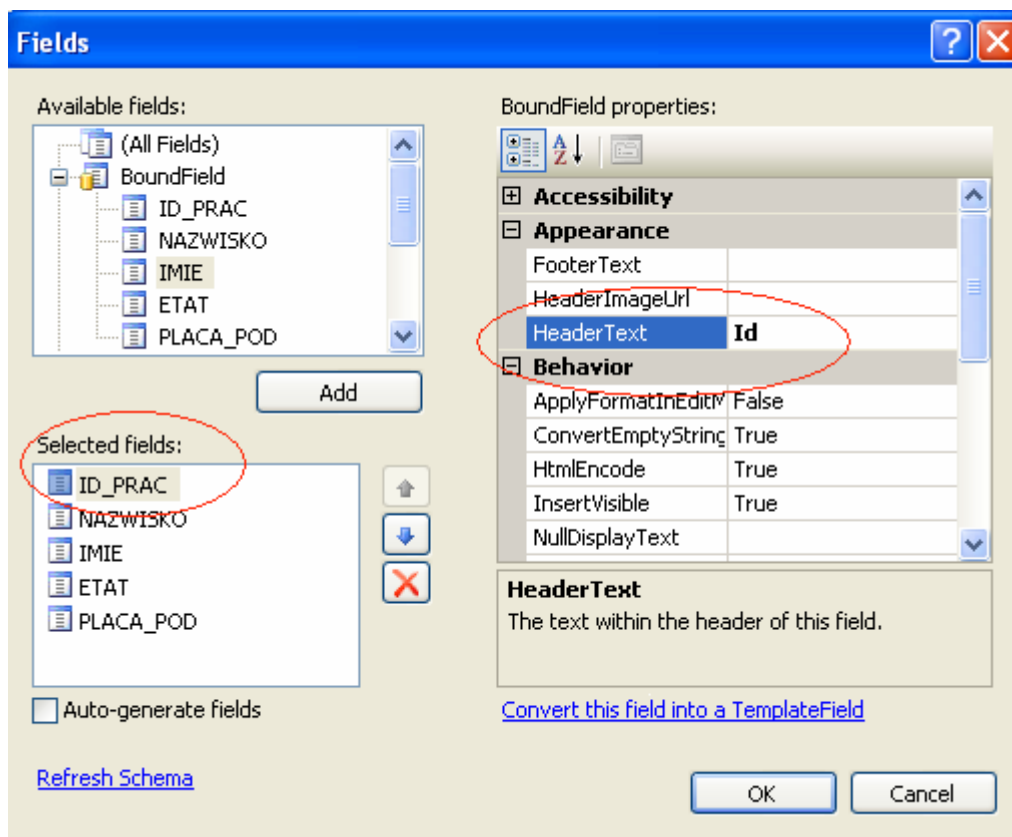
- e) Technika drag-and-drop umieść tabelę PRACOWNICY na stronie Pracownicy.aspx.



- f) Z menu podręcznego dla umieszczonej przez kreator kontrolki typu GridView wybierz opcję Edit Columns.



- g) W wywołanym oknie Fields kolejno wybierając poszczególne kolumny w panelu Selected Fields zmień nagłówki kolumn (HeaderText) na „Id”, „Nazwisko”, „Imię”, „Etat” i „Placa”. Na zakończenie kliknij przycisk **OK**.



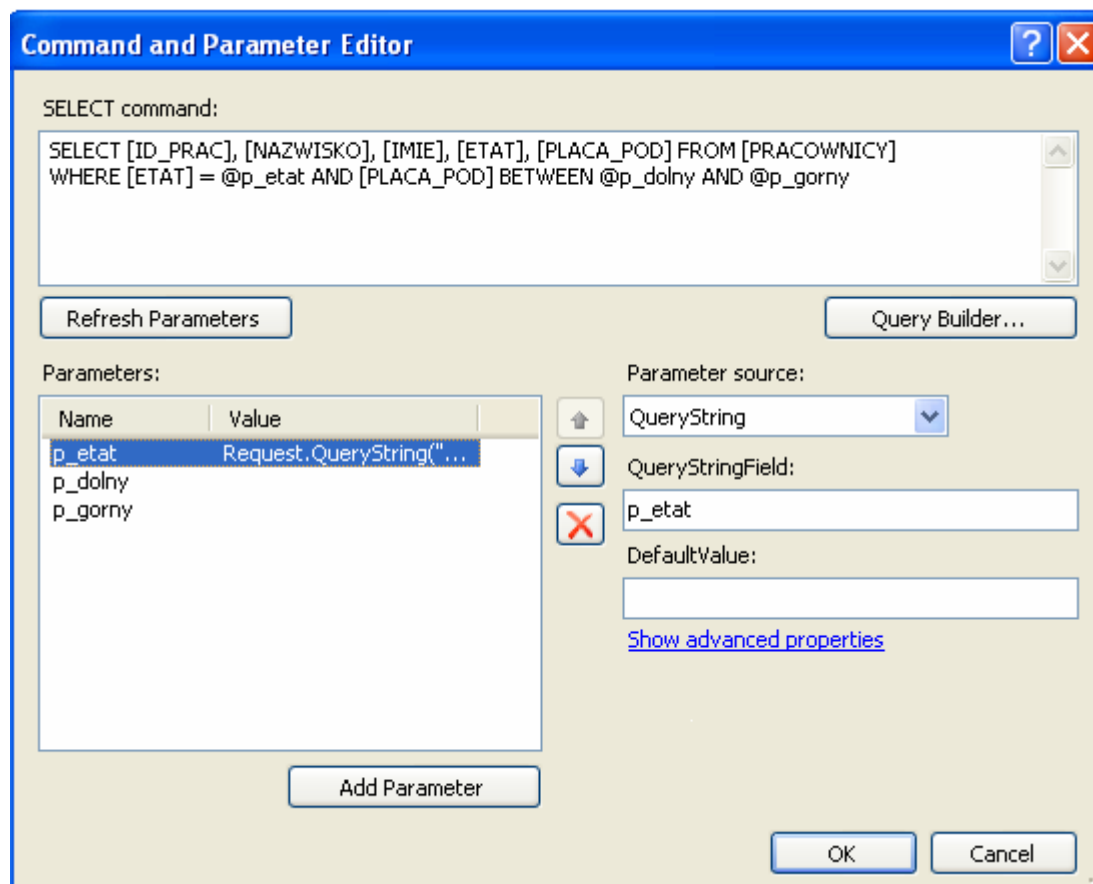
- h) Zapisz wszystkie zmiany (np. File→Save All).
- i) Uruchom stronę wybierając z menu kontekstowego opcję View in Browser. Strona w przeglądarce powinna zawierać tabelkę z danymi wszystkich pracowników.

5. Parametryzacja źródłowego zapytania dla kontrolki GridView.

- a) W trybie Design edycji strony `Pracownicy.aspx` kliknij myszą kontrolkę źródła danych (SqlDataSource), a następnie w paletce właściwości znajdź właściwość `SelectQuery` tej kontrolki. Wywołaj edytor wartości dla tej właściwości zaznaczając w paletce nazwę właściwości, a następnie klikając związany z nią przycisk `...`, który pojawi się po zaznaczeniu właściwości.
- b) W oknie edytora treści zapytania `SELECT` dla źródła danych rozszerz zapytanie o następującą klauzulę `WHERE`:

```
WHERE [ETAT] = @p_etat AND [PLACA_POD] BETWEEN @p_dolny AND @p_gorny
```

- c) Kliknij przycisk `Refresh Parameters`, aby w panelu `Parameters` pojawiły się wykorzystywane w zapytaniu parametry `p_etat`, `p_dolny` i `p_gorny`.
- d) Jako źródło parametru `p_etat` wybierz `QueryString`. Następnie jako nazwę parametru w adresie URL (`QueryStringField`) podaj „`p_etat`”.



- e) Powtórz krok d) dla parametrów `p_dolny` i `p_gorny`, podając jako nazwę parametru w adresie URL odpowiednio „`p_dolny`” i „`p_gorny`”. Kliknij przycisk `OK`.
 - f) Za pomocą palety właściwości zmień wartość właściwości `EmptyDataText` dla komponentu `GridView` na „Brak pracowników spełniających zadane kryteria.”.
 - g) Zapisz wszystkie zmiany (np. `File`→`Save All`).
6. Dodanie linku umożliwiającego powrót ze strony wyświetlającej dane pracowników (`Pracownicy.aspx`) do strony startowej, umożliwiającej wprowadzanie kryteriów selekcji (`Default.aspx`)

- a) Technika drag-and-drop umieść na stronie `Pracownicy.aspx` pod kontrolką źródła danych kontrolkę `HyperLink` z grupy `Standard`.
- b) Poprzez paletę właściwości ustaw dla dodanej kontrolki `HyperLink` właściwość `Text` na „Powrót”, a dla właściwości `NavigateURL` wywołaj kreator zaznaczając w palecie nazwę właściwości, a następnie klikając związany z nią przycisk `...`, który pojawi się po zaznaczeniu właściwości. Jako stronę, do której ma prowadzić link wybierz `Default.aspx` i kliknij przycisk `OK`.
- c) Zapisz wszystkie zmiany (np. `File`→`Save All`).

7. Implementacja nawigacji do strony z informacjami o pracownikach ze strony startowej z formularzem do wprowadzania kryteriów selekcji.

- a) Przejdź do edycji strony `Default.aspx` w trybie `Design`.
- b) Kliknij dwukrotnie przycisk w celu oprogramowania zdarzenia kliknięcia przycisku.
- c) Jako ciało utworzonej przez kreator metody `SzukajButton_Click()` wprowadź poniższy kod:

```
Response.Redirect( "Pracownicy.aspx?p_etat=" +  
    Server.UrlEncode( EtatDropDownList.Text ) +  
    "&p_dolny=" + DolnyTextBox.Text +  
    "&p_gorny=" + GornyTextBox.Text );
```

Powyższy fragment kodu realizuje przekierowanie do strony `Pracownicy.aspx` metodą `Response.Redirect()`, przekazując w adresie URL wartości parametrów `p_etat`, `p_dolny` i `p_gorny`, reprezentujące kryteria selekcji pracowników wprowadzone przez użytkownika do formularza. Ponieważ nazwy etatów mogłyby potencjalnie zawierać znaki zabronione w adresach URL, nazwa wybranego etatu jest kodowana metodą `Server.UrlEncode()`.

- d) Zapisz wszystkie zmiany (np. `File`→`Save All`).
- e) Uruchom stronę `Default.aspx` wybierając z menu kontekstowego opcję `View in Browser`.
- f) Przetestuj aplikację wybierając z listy różne etaty i różne progi płacy.

8. Przygotowanie aplikacji do instalacji na serwerze produkcyjnym.

- a) Wybierz z menu opcję `Build`→`Publish Web Site`.
- b) Zaakceptuj i zapamiętaj zaproponowany przez kreator katalog docelowy. Pozostaw wszystkie opcje domyślne i kliknij przycisk `OK`.
- c) We wskazanym katalogu umieszczona zostanie skompilowana aplikacja gotowa do skopiowania na produkcyjny serwer.