

PxCaseStudio v.1.12 for the Programings to Asp.Net application

PxCaseStudio v.1.12

for ASP.NET and C#

**Univerzálny nástroj pre modelovanie, evidenciu dát a generovanie funkčnej
databázovej aplikácie bežiackej pod databázami Oracle, MS SQL, MySQL,
Firebird, Interbase**

Obsah

1. Úvod
2. Zadávacie nové užívateľské skupiny (Form „užívateľia/users“)
3. Zadávacie formuláre (Form „Formuláre“)
4. Zadávacie tabuľky/ entít (Form „Entity / Tabuľky“)
5. Zadanie prepojenia formulára s tabuľkou (Form „Components to Form“)
6. Zadanie prepojenia medzi tabuľkami (Form „Field to Field / to Link“)
7. Vytvorenie menu aplikácie (Form „Menu“)
8. Výsledné generovanie aplikácie

1. Úvod

PxCaseStudio je univerzálny nástroj pre modelovanie, evidenciu dát a generovanie funkčnej databázovej aplikácie bežiackej pod ASP.NET, programovací jazyk C#.

PxCaseStudio Vám umožní navrhovať rozsiahle databázové aplikácie pre rôzne databázy behom niekoľkých minút! Podpora zahŕňa databázové servery, ako sú Oracle, MS SQL Server, MySQL, Firebird a Interbase server.

PxCaseStudio generuje SQL script a natívny kód aplikácie v jazyku C# s jednotlivými formulármi, tabuľkami, gridmi, editačnými prvkami, menu a pod. Výstupný kód je do značnej miery vystavaný na komponentách Px Frameworku, pričom Px Framework bol špeciálne vyvinutý na to aby umožnil takéto generovanie aplikácie z PxCaseStudia.

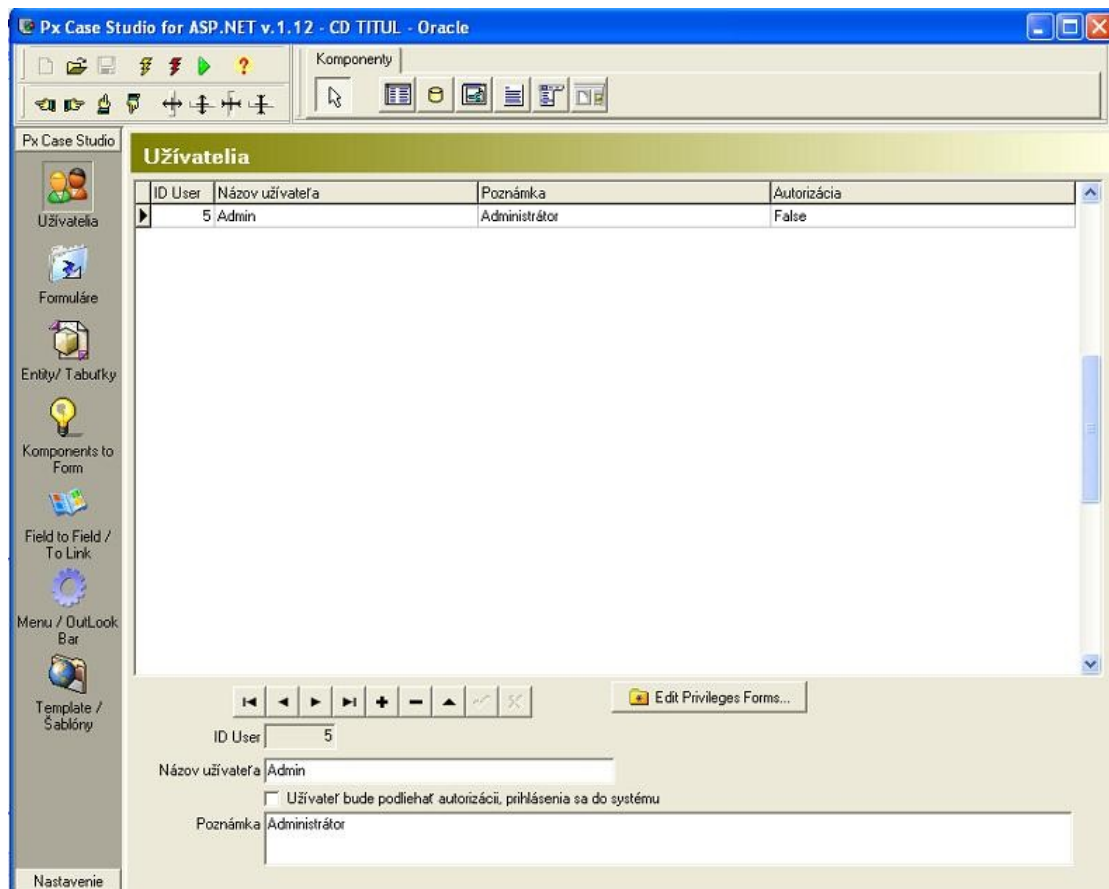
Tu sa jedná o presné modelovanie a komponenty Px Frameworku sú presnými stavebnými tehličkami, ktoré toto presné modelovanie umožňujú.

Px Framework zavádza filozofiu odkladania konfiguračných dát, vzťahov a prepojení, jednotlivých typov validácií a pod. na databázovej komponente PxWebQuery, cez metódu AddParam... a jej odvodeniny. Toto značne urýchľuje vývoj aplikácie, jej ľahšiu modifikovateľnosť a zlepšuje prehľadnosť zdrojového kódu. Pokiaľ chcete PxCaseStudio využívať efektívne, doporučuje sa Vám naštudovať si filozofiu a princíp funkcie komponent Px Frameworku.

2. Zadávanie nových užívateľských skupín (Form „užívateľia/users“)

Do formulára užívateľa sa zadávajú všetky užívateľské skupiny, ktoré by mali existovať pre danú aplikáciu, ktorú chceme vytvárať. Vložiť novú užívateľskú skupinu, môžeme cez tlačítka „plus“.

Tlačítkom „Edit Privileges Form“ sa nám otvorí formulár, v ktorom môžeme pridelovať privilégia, prístupové práva pre jednotlivé formuláre, a tak pridelovať prístupové práva jednotlivým skupinám. Ale tento formulár otvárame len vtedy, keď máme zadané všetky formuláre. Do položky „Názov užívateľa“, vkladáme názov skupiny bez diakritiky a medzier.



3. Zadávanie formulárov (Form „Formuláre“)

V tomto formulári si zadáme všetky formuláre, ich názov, Caption, a zaškrtneme si či chceme generovať View Form, AddEdit Form a Info Form. Teda pri definovaní jedného formulára, sa nám môžu vygenerovať až tri rôzne formuláre z odlišnou funkciou. „View Form“ je formulár ktorý obsahuje v sebe grid(PxSuperGrid), a slúži na zobrazenie dát tabuľky, ich selekciu, filtráciu, sortovanie a pod. Formulár „AddEdit Form“ slúži na vkladanie nových dát a ich editáciu a pod., a je vyskladaný z editačných prvkov PxEdit, PxComboBox, PxJSDatePicker a pod. Formulár „Info Form“ slúži na zobrazenie detailu daného zvoleného riadka z tabuľky, a je vyskladaný z prvkov pre zobrazenie dát PxLabel a pod.

Ku formuláru „AddEdit Form“ si zadáme jednotlivé titles názvy, ktoré sa budú zobrazovať v záhlaví formulára. Toto môžeme zadať pre dva stavy, a to vkladanie a editáciu riadka tabuľky. Ďalej sa môžeme rozhodnúť, ktorý z troch formulárov a to „View Form“, „AddEdit Form“ a „Info Form“ chceme generovať. Ak chceme generovať všetky tri, tak zaškrtneme všetky tri formuláre. Do položky „Názov formulára“, vkladáme názov formulára bez diakritiky a medzier.

ID Form	Názov užívateľa	Default Start	Caption	Poznámka
29	CDTitul	True	CD Titul	
30	Author	False	Autor	
31	Songs	False	Piesne	
32	Zaner	False	Žáner	
33	CDNosic	False	CD Nosič	
34	CDFormat	False	CD Formát	
35	Vydavatel	False	Vydavateľ	
37	Kateg_Zaner	False	Kategória žaner	

Id Form	29			
Názov formulára	CDTitul			
Caption	CD Titul			
Typ ViewFormu	frGrid	<input checked="" type="checkbox"/> Use ViewForm	Add Caption	Pridať nový CD Titul
Typ AddEditFormu	frTextBox	<input checked="" type="checkbox"/> Use AddEditForm	Edit Caption	Editovať existujúci CD Titul
Typ InfoFormu	frLabel	<input checked="" type="checkbox"/> Use InfoForm		
	<input checked="" type="checkbox"/> Default Start			
Poznámka				

4. Zadávanie tabuliek / entít (Form „Entity / Tabuľky“)

V tomto formulári si zadávame tabuľky, viewi a pod., a potom zadávame ich štruktúru. Tu môžeme zadávať niekoľko typov entít, ktorých zoznam nájdeme v „type tabuľky“. Sú to „Tabuľky“, „Číselníky“, „Query“, „Internal Query“ a „Viewi“. Do položky „Názov tabuľky“, vkladáme názov entity bez diakritiky a medzier.

4.1. Vloženie, alebo vytvorenie novej „Tabuľky“

Vložiť novú tabuľku môžeme cez tlačítko „plus“. Vyplníme názov tabuľky bez diakritiky a medzery a zadáme typ tabuľky „ttTable“. Tlačítkom na ktorom je fajočka, uložíme do databázy. Potom stlačíme tlačítko „Edit Structure Table“. Otvorí sa nám formulár „Add Structure Table“. Tu zadávame jednotlivé stĺpce tabuľky, ich názov bez diakritiky a medzery, ďalej Field text, typ fieldu a jeho veľkosť. Ďalej zaškrtnáme či stĺpec je primárny kľúč, či field bude „Not Null“ poprípade jedinečný (IsUnique).

Každá tabuľka musí obsahovať stĺpec s primárnym kľúčom. Px Framework vyžaduje pre svoju správnu funkciu aby tabuľka, mala primárny kľúč.

Keď tvoríte stĺpec primárneho kľúča, dajte mu názov podľa nasledujúceho pravidla. Využite názov tabuľky pre ktorú sa tvorí tento stĺpec a pred ňu dajte slovo „ID“.

Napríklad pre tabuľku „CdTitul“, bude sa stĺpec primárneho kľúča volať „IDCdTitul“

Vytvorenie názvu stĺpca, ktorý bude primárny kľúč:

ID+<Name of Table>

4.2. Vloženie, alebo vytvorenie nového „Číselníka“

Číselník sa vytvára presne ako tabuľka, len sa pre typ tabuľky zvolí hodnota „ttCiselnik“. Podobne ako pri tabuľke sa zadávajú stĺpce tabuľky, nezabudnúť na stĺpec s primárnym kľúčom.

4.3. Vloženie, alebo vytvorenie novej „Query“

Pokiaľ je už v databáze vytvorená daná tabuľka, potom môžeme túto tabuľku využiť cez typ entity a to je „ttQuery“.

Query je čistý Sql príkaz, v tvare „select * from table_name“, ktorý zadávame do položky „SQL Script“. Takže pri vyplňaní fieldov budeme zadávať iba popisok, t.j. Caption tabuľky.

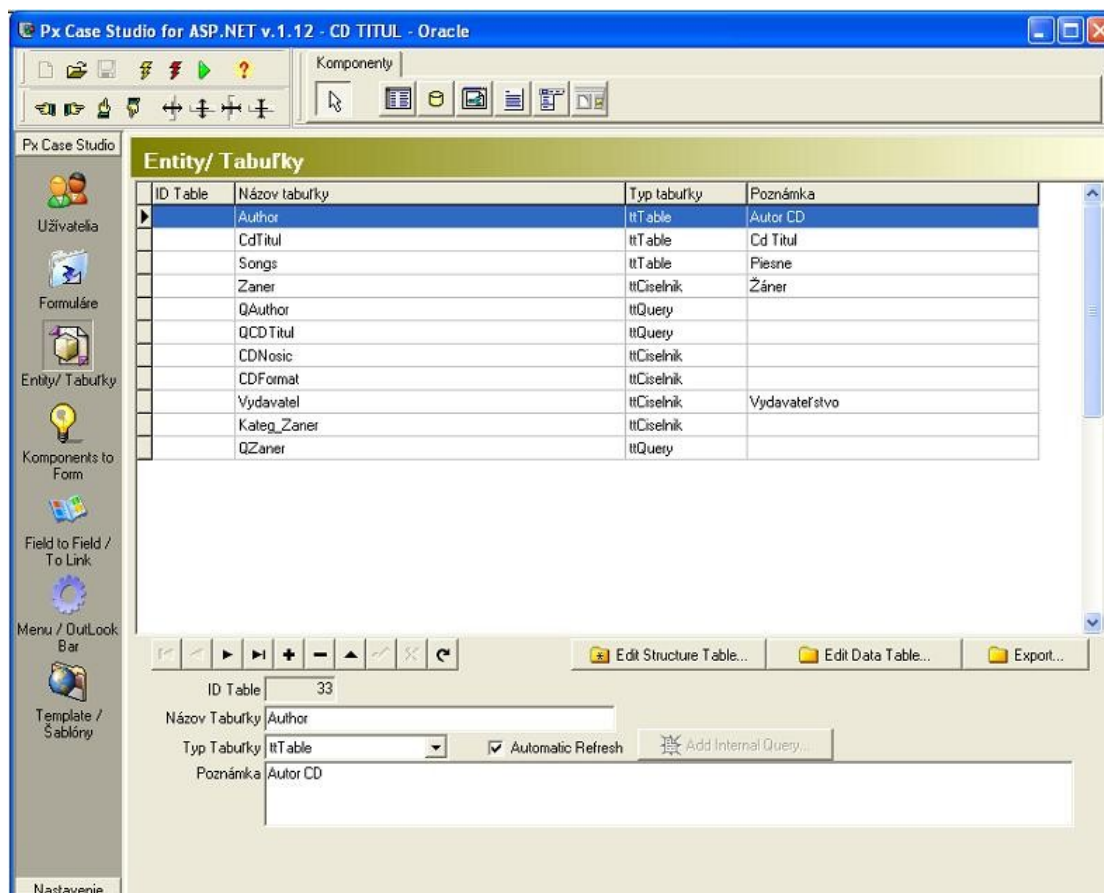
4.4. Vloženie, alebo vytvorenie novej „Internal Query“

Pokiaľ sme do PxCaseStudia nahodili tabuľku typu „ttTable“ a „ttCiselnik“, môžeme ho využiť na zadanie entity, ktorá je typu „ttInternalQuery“. Pri internal query zadávame čistý SQL select, už nad existujúcimi tabuľkami, ktoré sú nahodené v PxCaseStudiu. K tomu nám môže dopomôcť tlačítko „Add Internal Query“. Po stlačení tohto tlačítka, sa nám otvorí formulár kde si vyberieme tabuľku z ktorej, si zaškrtneme jednotlivé stĺpce, z ktorých sa nám vytvorí konkrétny SQL príkaz.

Pri vypĺňaní fieldov budeme zadávať iba popisok, t.j. Caption tabuľky.

4.5. Vloženie, alebo vytvorenie nového „Viewu“

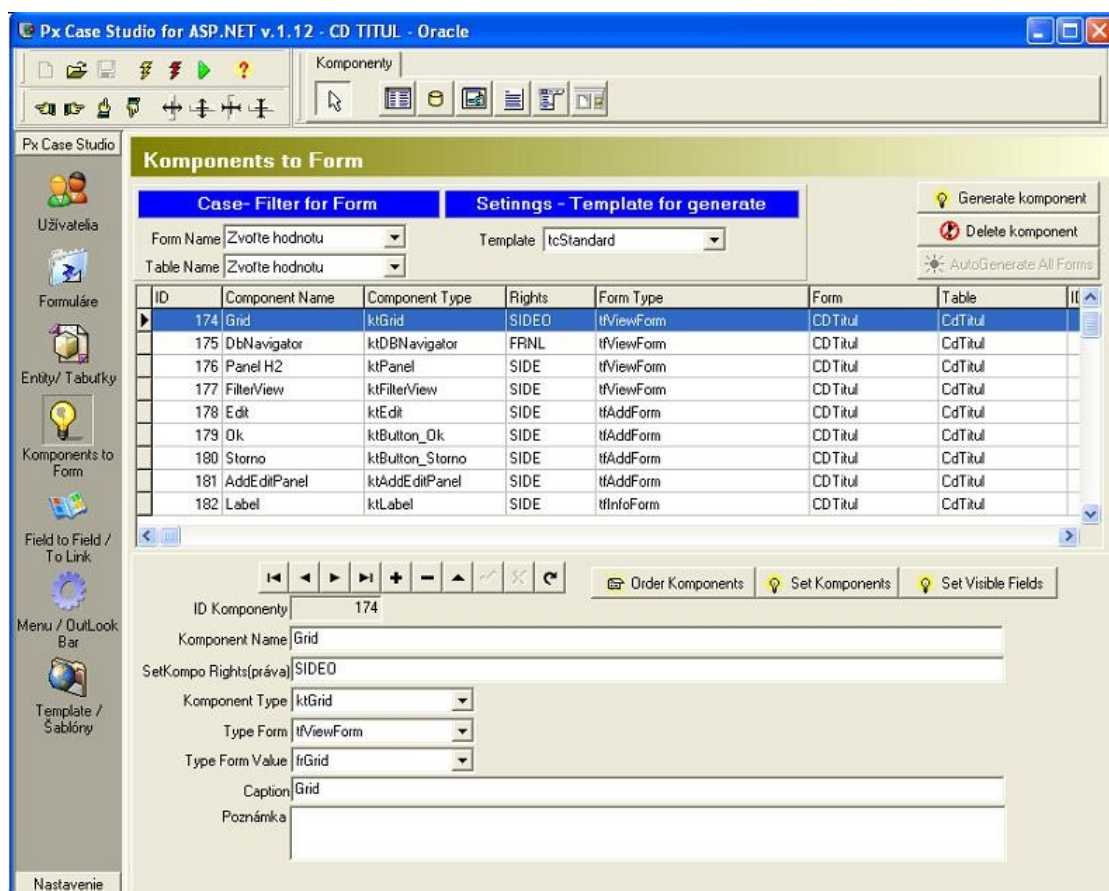
V prípade že sme vytvárali entity typu „ttTable“ a „ttCiselnik“, sa nám pri generovaní výsledného SQL Scriptu tieto entity vygenerujú ako tabuľka databázy. Pri entite typu „ttView“ sa nám generuje, alebo vkladá SQL príkaz ktorý sme vložili do položky „SQL Script. Tento typ entity je vhodný na zadávanie Viewov. Stĺpce tabuľky sa nevyplňajú.



5. Zadanie prepojenia formulára s tabuľkou (Form „Components to Form“)

V tomto formulári prebieha pred generovanie, ku jednotlivým formulárom sa pripájajú jednotlivé tabuľky, číselníky, query a pod. Najprv si zvolíme formulár a potom tabuľku ktorá bude mať výstup na danom formuláry a zvolíme si ešte typ šablóny(template). Keď toto máme stlačíme tlačítko „Generate component“. Toto predgenerovanie opakujeme pre každý formulár zvlášť.

Po stlačení tlačítka „Generate component“ sa nám podľa šablóny vygenerujú jednotlivé prvky, komponenty na daný formulár. Tieto komponenty nám prepájajú formulár s jednotlivými tabuľkami v databázy. Keď chceme generovať formulár s číselníkom, použijeme šablónu s názvom „tcČíselník“. Inak používame štandardnú šablónu „tcStandard“.



Cez tlačítko „Order component“ si zvolíme poradie zobrazenia komponent smerom od vrchu dole na formulári v ktorom budú zobrazené.

Cez tlačítko „Set Visible Fields“ si nastavujeme, ktoré položky, fieldy budú zobrazené v jednotlivých formulároch a to vo formulároch „View Form“, „AddForm“ a „Info Form“. Vo formulári „ViewForm“, sa redukuje počet zobrazených stĺpcov, pokiaľ ich je mnoho, a zobrazujú sa len tie najviac dôležité. Vo formulároch „AddForm“ a „Info Form“ sa zobrazujú prevažne všetky stĺpce vo formulári, ktoré chceme editovať.

6. Zadanie prepojenia medzi tabuľkami (Form „Field to Field / to Link“)

Pomocou tohto formulára vytvárame prepojenie medzi jednotlivými entitami, tabuľkami, vlastne zadávame vzťah medzi jednotlivými tabuľkami (relations ship). Tieto jednotlivé vzťahy sú potom pri generovaní aj premietnuté do vygenerovanej aplikácie.

Medzi jednoduchý vzťah medzi tabuľkou a číselníkom patri vzťah „AddParamKey“ a „AddParamWebQuery“. Zložitejší vzťah, ktorý umožňuje zadať kategóriu a podkategóriu je „AddParamFlyComboBox“.

Ku týmto jednotlivým vzťahom si treba preštudovať jednotlivé parametre komponenty PxWebQuery (AddParam...). Komponenta PxWebQuery je komponentou Px Frameworku.

The screenshot shows the 'Field to Field / ParamKey / ParamCheck' configuration window in Px Case Studio. The window title is 'Px Case Studio for ASP.NET v.1.12 - CD TITUL - Oracle'. The main area contains a table with the following data:

ID	ID Table	ID Table of Code List	Field Name	Field Name of CL	Field Name of CL
79	34	37	IdAuthor1	IdAuthor	Name
80	35	38	IDCDTitul	IDCDTITUL	name
81	34	39	IdCDNosit	IdCDNosit	Name
82	34	40	IDCDFormat	IDCDFormat	Name
83	34	42	IdVydavatel	IdVydavatel	Vydavatel
84	34		Author		
85	34	45	ZanerFull	IDKATEG_ZANER;IDZaner	
86	34	37	IdAuthor2	IdAuthor	Name
87	34	37	IdAuthor3	IdAuthor	Name

Below the table, there are several configuration fields:

- ID FieldToField: 79
- Type Field To Field: fvParamWebQuery
- Table Name: CdTitul
- FieldName of Table: IdAuthor1
- Table Name Číselník: QAuthor
- FieldName1 of Číselník: IdAuthor
- FieldName2 of Číselník: Name
- Key: IdAuthor
- Value: Name
- Field To View: aValue
- Field To Db: aKey

The window also includes a 'Poznámka' (Note) field and a 'Nastavenie' (Settings) button at the bottom left.

7. Vytvorenie menu aplikácie (Form „Menu“)

Tento formulár nám slúži na vytvorenie menu a jeho prepojenie s jednotlivými formulármi.

V položke „Menu name“, zadávame názov menu a to bez diakritiky a bez medzery.

V položke „Caption“ zadávame skutočný názov menu, ktorý sa objaví v aplikácii.

V položke „Link to Form“ si z menu vyberáme formulár na ktorý má byť napojené menu. Po kliknutí na menu sa nám otvorí formulár ktorý je zadaný v položke „Link to Form“. Pokiaľ by sme v menu chceli vytvoriť submenu, môžeme to urobiť cez položku „Parent Menu“, do ktorej zadáme názov koreňového menu. Pokiaľ by sme chceli položky menu zoradiť v istom poradí môžeme to urobiť cez tlačítko „Order Menu“.

The screenshot displays the 'Menu / Outlook Bar' configuration window in Px Case Studio. The window title is 'Px Case Studio for ASP.NET v.1.12 - CD TITUL - Oracle'. The interface includes a toolbar with various icons and a 'Komponenty' section. The main area shows a table of menu items with columns for IDMenu, Menu Name, Caption, and Link. Below the table is a control panel with navigation buttons and an 'Order Menu' button. The form fields are populated with the following data:

IDMenu	Menu Name	Caption	Link
30	mnuCDTitul	Cd Titul	23
31	mnuAuthor	Autor	30
32	mnuSongs	Piesne	31
33	mnuCis	Číselníky	32
34	mnuZaner	Žáner	32
35	mnuCDNosic	CD Nosič	33
36	mnuCDFormat	CD Formát	34
37	mnuVydavatel	Vydavateľ	35
39	mnuKateg_Zaner	Kategória Žáner	37

Control Panel Fields:

- ID Menu: 30
- Menu Name: mnuCDTitul
- Caption: Cd Titul
- Link To Form: CDTitul
- Parameter: (empty)
- Parent Menu: (empty)
- Poznámka: (empty)

Buttons: Order Menu

8. Výsledné generovanie aplikácie

Keď máme nahodené všetky dáta a máme ich vzájomne poprepájané môžeme začať s výsledným generovaním, kompilovaním aplikácie. Ale ešte pred tým vojdeme do nastavenia aplikácie, a tam nastavíme Connection String, title a názov aplikácie a tiež typ databázy pre ktorú bude aplikácia generovaná. Môžeme si zvoliť z nasledujúcich databáz: Oracle, MS SQL, MySQL, FireBird a Interbase.

V ďalšom tu máme popis tlačítok pomocou ktorých môžeme generovať výsledný obsah aplikácie:



- cez toto tlačítko generujeme výsledný SQL Script



- cez toto tlačítko generujeme výsledný kód aplikácie v ASP.NET a C#



- cez toto tlačítko generujeme SQL Script a výsledný kód aplikácie v ASP.NET

Výsledný vygenerovaný kód a Sql script sa uloží do adresára, ktorý sme zadávali pri vytvorení projektu. Tento výstupný adresár môžeme zistiť pri otváraní projektu, posunutím kurzora tabuľky do ľava, výstupný adresár je zapísaný v stĺpci „Output Dir“. V nastavení si môžeme zvoliť pre aký typ servera, chceme vygenerovať aplikáciu. Ak sme zmenili typ servera, potom musíme úplne vymazať adresár „Output Dir“ a spustiť kompiláciu celého projektu od znova.

Keď máme celú aplikáciu vygenerovanú, mala by sa nachádzať v adresári „Output Dir“. V tomto adresári by sa mal nachádzať aj súbor „SQLScript.sql“. Zoberieme tento súbor a vytvoríme z neho databázu pre nami vygenerovanú aplikáciu.

Potom si spustíme MS Visual Studio, a vygenerovaný projekt si otvoríme cez položku „**Open website...**“.

V inštalácii PxCaseStudia, je nahratý projekt s názvom CD Titul. Tento projekt je plne funkčný príklad, ako pomocou PxCaseStudia, môžeme vytvoriť plne funkčnú databázovú aplikáciu. Pokiaľ by sme si chceli vygenerovať výsledný projekt, pred jeho generovaním si v nastavení zvolíme pre aký typ databázy má byť projekt vygenerovaný.

Prajeme Vám veľa úspechov pri práci s aplikáciou PxCaseStudio.

Vzt'ahy medzi jednotlivými tabuľkami(Relation Ship).

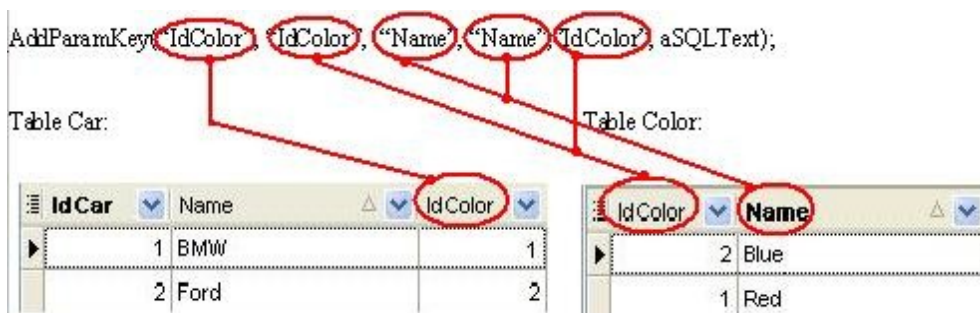
Parameter AddParamKey() komponenty PxWebQuery:

Def. fields:

AddParamKey(aFieldName, aFieldNameKey, aFieldNameValue, aFieldToView, aFieldToDB, aSQLText);

aSQLText = "select IdColor, Name from Color";

PxWebQuery.AddParamKey("IdColor", "IdColor", "Name", "Name", "IdColor", aSQLText);



Výstup v komponente PxSuperGrid:

...	IdCar	Name	IdColor
	1	BMW	Red
	2	Ford	Blue

PRIDAŤ NOVÝ RIADOK...

Page 1 Počet záznamov: 2

Výstup v komponente PxComboBox:

IdColor

- Zvolte hodnotu
- Blue
- Red

Zadanie parametra AddParamKey v PxCaseStudiu:

Type Field To Field: fvParamKey

Table Name: Car

FieldName of Table: IdColor

Table Name Číselník: Color

FieldName1 of Číselník: IdColor

FieldName2 of Číselník: Name

FieldName3 of Číselník:

FieldName4 of Číselník:

Key: IdColor

Value: Name

Field To View: aValue

Field To Db: aKey

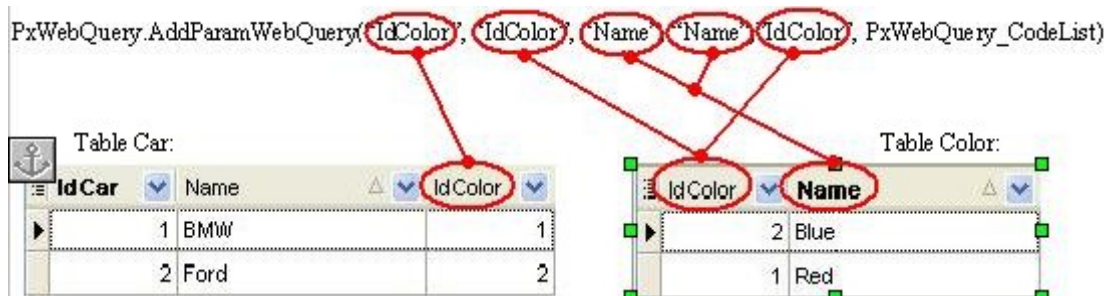
Parameter AddParamWebQuery() komponenty PxWebQuery:

Def. fields:

AddParamWebQuery(aFieldName, aFieldNameKey, aFieldNameValue, aFieldToView, aFieldToDB, PxWebQuery_CodeList);

PxWebQuery_CodeList.SQLSelect = "select IdColor, Name from Color";

PxWebQuery.AddParamWebQuery("IdColor", "IdColor", "Name", "Name", "IdColor", PxWebQuery_CodeList);



Výstup v komponente PxSuperGrid:

...	IdCar	Name	IdColor
	1	BMW	Red
	2	Ford	Blue
PRIDAŤ NOVÝ RIADOK...			
Page 1		Počet záznamov: 2	

Navigation icons: back, forward, search, refresh, close.

Výstup v komponente PxComboBox:

IdColor

- Zvolte hodnotu
- Blue
- Red

Zadanie parametra AddParamWebQuery v PxCaseStudios:

Type Field To Field: fvParamWebQuery

Table Name: Car

FieldName of Table: IdColor

Table Name Číselník: Color

FieldName1 of Číselník: IdColor

FieldName2 of Číselník: Name

Key: IdColor

Value: Name

Field To View: aValue

Field To Db: aKey

