

NÁVRH A TVORBA DATADÁZ VO VÝVOJOVOM PROSTREDÍ DELPHI

BORŽÍKOVÁ Jana, SR

Resumé: Príspevok sa zaoberá využitím aplikačných úloh pri vyučovaní programovacích jazykov na technických fakultách. Špeciálne v tejto druhej časti ide o tvorbu konkrétnej databázy s využitím ovládacích prvkov a dotazov.

Kľúčové slová: relačné databázy, prostredie Delphi, dotazy

THE DESIGN AND THE CREATION OF DATABASES BY DEVELOPMENT ENVIRONMENT DELPHI

Abstract: The paper deals with using of applications problems in teaching of programming languages at technical faculties. Specially, there is created concrete database with using components and queries.

Keywords: relational databases, development environment Delphi, queries

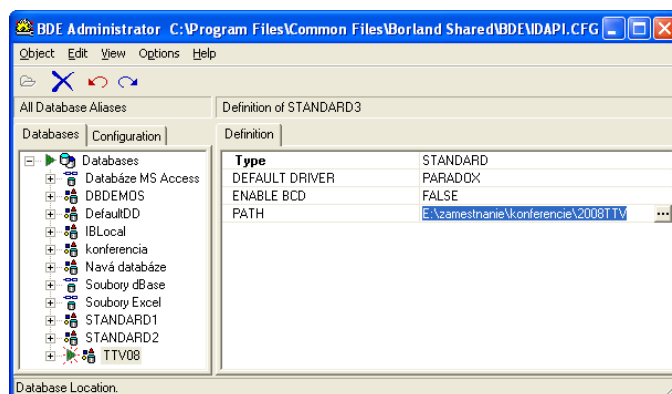
1 Úvod

Na základe predchádzajúceho súvisiaceho článku vytvoríme databázu. Využívame pritom návrh databázy naznačený na obrázku 2. Databáza je členená do troch tabuliek medzi nimi je navrhnutá relácia 1:N. Takto je definovaná báza dát, k nej následne vytvoríme systém riadenia bázy dát, ktorý zaistí nasledujúce činnosti:

- definovanie štruktúry
- ukladanie dát
- výber dát
- oprava dát
- komunikácia medzi užívateľom a systémom. (Kadlec, 2001)

2 Nástroj BDE Administrátor

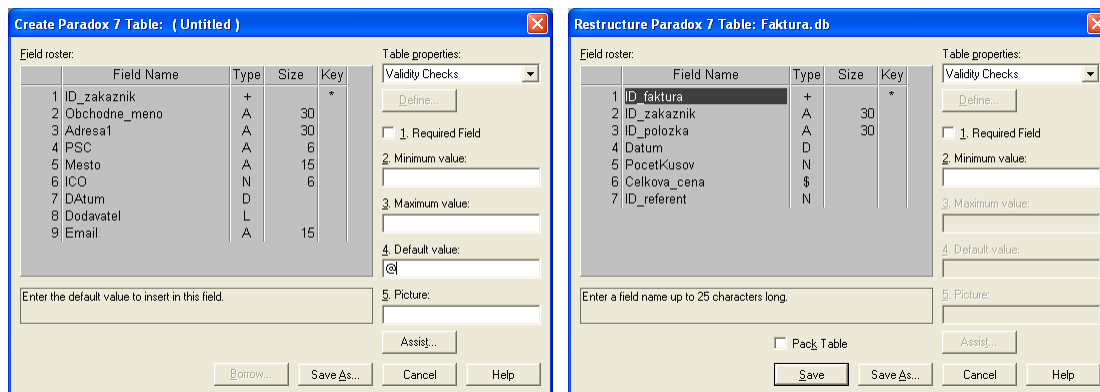
Pomocou nástroja BDE Administrátor vytvoríme Alias, aby sme mohli cez rozhranie pristupovať v rôznom typom databáz. Toto riešenie je vhodnejšie pre nie príliš rozsiahle databázy. (Obrázok 1)



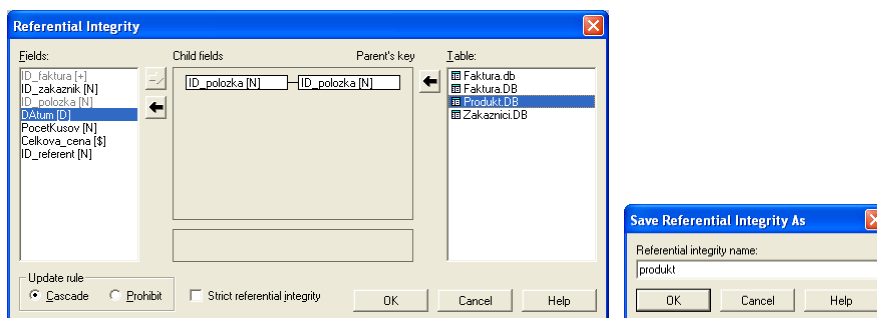
Obrázok 1. Vytvorenie Aliasu s nastavením prístupu k databáze.

3 Program Database Desktop

Pomocou klientského programu Database Desktop definujeme jednotlivé tabuľky, polia, typy polí, nastavujeme default hodnoty, dolné a horné ohraňovania. (Obrázok 2) Následne definujeme relácie medzi navrhnutými tabuľkami, tak ako bolo uvedené v predchádzajúcej časti, na obrázku 1. (Obrázok 3)



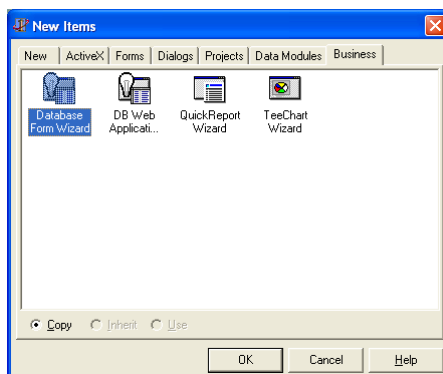
Obrázok 2. Definovanie tabuľky Zákazníci a tabuľky Faktúra podľa návrhu.



Obrázok 3. Definovanie relačných vzťahov, uloženie pod menom.

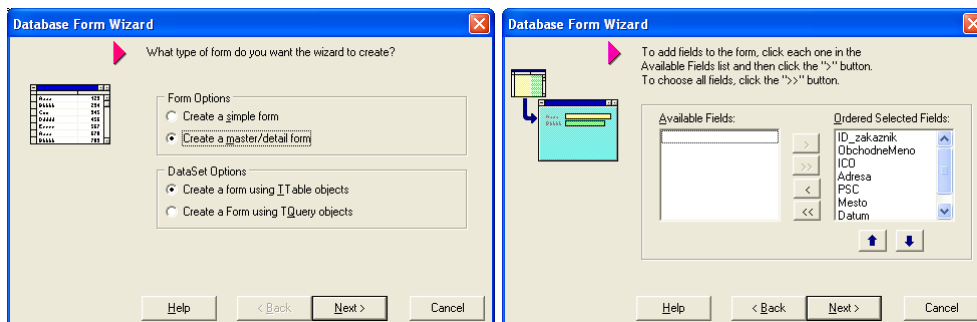
4 Vytvorenie aplikácie pomocou sprievodcu

Ak vytvárame aplikáciu, ktorá prístupuje k jednej tabuľke využívame jednotlivé komponenty z palety nástrojov Data Access a Data Control. Pri komponentách Datasource, Table a následne aj pri použití DBGrid a DBNavigator potrebujeme v Object Inspector nastaviť vlastnosti tak, aby komponenty navzájom „komunikovali“.



Obrázok 4 Database from Wizard

Podrobnejšie v literatúre a programátorských príručkách (Kadlec, 2001, Eller, 2002) Vzhľadom na nastavené relácie je využijeme sprievodcu Tvorbou databázy (Obr. 4, 5)



Obrázok 5. Pridávanie polí, ktoré potrebujem vo formulári

Výsledkom je vstupný formulár, ktorý bude užívateľ využívať na napĺňanie databázy. V závislosti od typu poľa, môžeme ovládaci prvok Edit nahradiť iným a vložiť ďalšie komponenty. (Napríklad, ak predpokladám, že zákazníci sa budú vyskytovať, len v obmedzenom počte miest, nahradíme položku mesto rozbaľovacím ComboBox-om.)

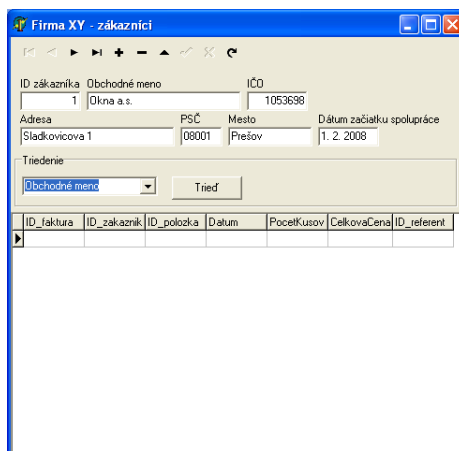
5 Vylepšenie aplikácie doplnením o Query

Aby boli dáta jednoducho prístupné užívateľovi, aby bolo možné ich ďalej spracovávať a vyhodnocovať (čo je vlastne hlavným cieľom podnikového IS) je nevyhnutné doplniť formulár o ďalšie ovládacie prvky:

1. Dotaz = Query z DataAccess. Jeho vlastnosť DatabaseName nastavíme na našu databázu.
2. GroupBox. Jeho vlastnosť Caption nastavíme na Triedenie
3. Rozbaľovací zoznam = ComboBox. Naplníme ho položkami ID Zákazníka, Obchodné meno, IČO, podľa ktorých budeme databázu triediť.
4. Tlačidlo = Button. Vlastnosť Caption nastavíme na Tried'.

A využiť príkazy z jazyka SQL:

5. **SELECT** <zoznam_výstupných_položiek> **FROM** <zoznam_tabuliek>
[**WHERE** <podmienka>] [**ORDER BY** <kritériá_triedenia>]



Obrázok 6. Výsledný formulár.

K metóde OnClick tlačidla, prípadne ku metóde OnClick rozbaľovacieho zoznamu priradíme kód:

```
Case combobox1.itemindex of
0:Begin
  DataSource1.Dataset:=Query1 ;
  Query1.SQL.Clear;
  Query1.SQL.Add('SELECT * FROM zakaznici1 ORDER BY id_zakaznik');
  Query1.Open;
End;
1 :Begin
  DataSource1.Dataset:=Query1 ;
  Query1.SQL.Clear;
  Query1.SQL.Add('SELECT * FROM zakaznici1 ORDER BY ObchodneMeno');
  Query1.Open;
End;
2 :Begin
  DataSource1.Dataset:=Query1 ;
  Query1.SQL.Clear;
  Query1.SQL.Add('SELECT * FROM zakaznici1 ORDER BY ICO');
  Query1.Open;
End;
end;
```

Obrázok 7. Časť zdrojového kódu s príkazmi dotazovacieho jazyka SQL

6 Záver

Článok sa zaoberá tvorbou databázy navrhnuť v predchádzajúcej časti. Výsledkom je databáza dát a k nej vytvorená aplikácia, pomocou ktorej s dátami pracujeme. Uvedený príklad je vhodný ako motivačná úloha na semestrálnu prácu z predmetu Programovacie jazyky, prezentuje prepojenie s praxou a s inými predmetmi na našej fakulte.

Úloha bola vypracovaná v rámci projektu VEGA č.1/0345/08 s názvom Modelovanie a simulácia mechatronických systémov pre strojárstvo.

4 Literatúra

1. BASL, J – Blažíček, R. *Podnikové informačné systémy*. 2. vyd. Praha : Grada Publishing, 2008. s. 283. ISBN 978-80-247-2279-5
2. ELLER, F. *Delphi 6. I.* vyd. Praha : Grada Publishing, 2002. s. 272. ISBN 80-247-0303-3.
3. KADLEC, V. *Učíme se programovat v Delphi a jazyce Object Pascal*. 1. vyd. Praha : Computer Press, 2001. s. 288. ISBN 80-7226-245-9.

Lektoroval: Peter Šebej, Ing. PhD.

Kontaktní adresa:

Jana Boržíková, PaedDr. Ph.D.,
Katedra matematiky, informatiky a kybernetiky,
Fakulta výrobných technológií TU KE so sídlom v Prešove,
Bayerova 1, 080 01 Prešov, SR,
tel. 00421 51 7723931, fax 0042 51 7734551,
e-mail jana.borzikova@tuke.sk