

**SLOVENSKÁ POĽNOHOSPODÁRSKA UNIVERZITA**  
**V NITRE**  
**TECHNICKÁ FAKULTA**

**PREPOJENIE A PRENOS ÚDAJOV MEDZI RÔZNYMI**  
**TYPMI DATABÁZ**

**Diplomová práca**

Študijný program:	Informačná a automatizačná technika v kvalite produkcie
Študijný odbor:	2386800, Kvalita produkcie
Školiace pracovisko:	Katedra elektrotechniky, automatizácie a informatiky
Školiteľ:	Ing. Ondrej Lukáč, PhD.

**Nitra, 2011**

**Michal Stanko, Bc.**



## **Čestné vyhlásenie**

Podpísaný Bc. Michal Stanko vyhlasujem, že som záverečnú prácu na tému „Prepojenie a prenos údajov medzi rôznymi typmi databáz“ vypracoval samostatne s použitím uvedenej literatúry.

Som si vedomý zákonných dôsledkov v prípade, ak uvedené údaje nie sú pravdivé.

V Nitre 22.apríla 2011

**Bc. Michal Stanko**

## **Pod'akovanie**

Ďakujem môjmu školiteľovi Ing. Ondrejovi Lukáčovi, PhD. za odborné vedenie práce, cenné rady a pripomienky. Ďakujem spoločnosti Semecs s.r.o Vráble za poskytnutie informácií potrebných k úspešnému dokončeniu práce, konkrétne Vladimírovi Budovi za pomoc a aktívnu spoluprácu a mojej rodine za ich podporu a porozumenie.

## **Abstrakt**

V tejto diplomovej práci sa zaoberáme tvorbou aplikácie, ktorá je schopná previesť údaje z databázy programu Microsoft Excel do Microsoft Access a naopak. Je teda potrebné vytvoriť program- „konvertor“, ktorý bude schopný prevádzať údaje medzi spomínanými typmi. Hlavný cieľ zahŕňa požiadavky na aplikáciu. Musí byť jednoduchá, aby prácu s ňou zvládol aj človek s menšími zručnosťami pri práci s počítačom. Nemala by byť náročná na výpočtovú zložitosť, aby nezaťažovala procesor a jej činnosť netrvala dlhý čas. Práca je rozdelená na dve časti. Prvá je venovaná súčasnej problematike databáz a programom, ktoré v práci používame, teda Delphi 7, MS Excel a MS Access. V druhej časti opisujeme tvorbu aplikácie, metódy a procedúry, ktoré sme použili, ako aj postup aplikácie. Súčasťou práce sú aj zdrojové kódy.

Kľúčové slová: databáza, Delphi, Excel, Access

## **Abstract**

The aim of this thesis is creation of an application which is capable of transferring data from the program database of Microsoft Excel to Microsoft Access and vice versa. Therefore it is necessary to create a program – “converter” which will be capable of transferring data between programs mentioned above. The main point involves requests for the application. It has to be simple so that even less skilful person in case of managing the PC can succeed. It should not be difficult according to computational complexity theory so that it does not apply a load to the processor and its operation does not take a long time. Work is divided into two parts. The first one refers to the actual issue of databases and programs which are used in the operations such as Delphi 7, MS Excel and MS Access. The second part describes the creation of the application, methods and procedures used as well as proceeding of the application. The thesis includes the source codes.

**Key words:** database, Delphi, Excel, Access

# Obsah

<b>Použité pojmy a skratky .....</b>	<b>9</b>
<b>Úvod .....</b>	<b>10</b>
<b>1. Prehľad o súčasnom stave riešenej problematiky .....</b>	<b>11</b>
1.1. Databázy .....	11
1.1.1. Typy databáz podľa obsahu .....	11
1.1.2. Rôzne typy databáz .....	12
1.1.2.1. Relačné databázy .....	12
1.1.2.2. Objektovo orientované databázy .....	12
1.1.2.3. Deduktívne databázy.....	12
1.1.2.4. Temporálne databázy.....	12
1.2. Vývojové prostredie Delphi.....	13
1.2.1. Typy súborov v Delphi .....	14
1.3. Tabuľkový procesor.....	16
1.3.1. Microsoft Excel.....	17
1.4. Microsoft Access .....	19
1.5. Spoločnosť Semecs, s.r.o. ....	21
<b>2. Cieľ práce .....</b>	<b>22</b>
<b>3. Metodika práce .....</b>	<b>23</b>
3.1. Požiadavky na riešenie diplomovej práce.....	23
3.2. Postup vytvorenia aplikácie .....	23
3.3. Použité nástroje a programy .....	23
<b>4. Vlastná práca a dosiahnuté výsledky .....</b>	<b>25</b>
4.1. Úvodný formulár aplikácie .....	25
4.2. Tvorba aplikácie .....	26
4.2.1. Načítanie súboru XLS.....	26
4.2.2. Zápis údajov do súboru mdb.....	27

4.2.3.	Načítanie súboru mdb .....	28
4.2.4.	Zápis údajov do súboru xls .....	28
4.2.5.	Kontrola funkčnosti aplikácie .....	29
<b>5.</b>	<b>Záver .....</b>	<b>31</b>
<b>6.</b>	<b>Použitá literatúra .....</b>	<b>32</b>
<b>7.</b>	<b>Prílohy .....</b>	<b>34</b>



## **Použité pojmy a skratky**

ADO	ActiveX Data Objects- prístup na databázový zdroj
OLE	Object Linking and Embedding- vkladanie a prepájanie objektov
SQL	Structured Query Language- jazyk na prácu s údajmi

# Úvod

Vývoj informatiky a digitalizácia informácií nám umožňuje pristupovať k nim kdekoľvek na svete. Zároveň s rastom množstva informácií rástla aj náročnosť ich spracovania a ukladania. Zber, spracovanie a ukladanie informácií je jedna zo základných vlastností informatiky. Skôr, keď sa v takom množstve ako dnes nevyužívala výpočtová technika, bol proces s uchovávaním informácií veľmi prácny, náročný na materiál ale aj na ľudskú prácu. Za databázu môžeme považovať takmer všetko, čo ukladá dáta rôzneho druhu a typu v akejkoľvek podobe. Databáza môže byť kniha dochádzok, katalóg výrobkov, ale aj pracovných postupov, zoznam zamestnancov a veľa iného. V dnešnej dobe je trend ukladať elektronicky všetky údaje, ktoré sa len uložiť dajú. Má to niekoľko výhod. Jedna z hlavných je jednoduchosť a úbytok práce, ďalej jednotnosť alebo podobnosť formátu. Nemenej dôležité je vyhľadávanie. Keď si predstavíme, že máme hľadať nejaký údaj v zozname stoviek až tisícov záznamov, určite je jednoduchšie nájsť potrebnú informáciu po pár kliknutiach. S rastom a vývojom počítačov a informatiky ako takej okolo nás sa znižuje namáhavosť na prácu s údajmi. Vznikajú nové technológie, ktoré uľahčujú prácu s údajmi, ponúkané sú nové programové balíky, ktoré ponúkajú čoraz viac možností. Na ukladanie údajov sa využívajú databázy rôznych typov.

V práci sa budeme zaoberať tvorbou programu pre prácu s databázami pomocou programu Delphi 7 Enterprise od spoločnosti Borland. Delphi je známe, zrozumiteľné a obľúbené vývojové prostredie pre Windows. Využíva klasické objektovo orientované programovanie pomocou formulárov doplnených o funkcie a rôzne komponenty. Naša aplikácia by mala rýchlo a bez strát prenášať jednotlivé údaje z programu Excel do programu Access a naopak. Práca je smerovaná na opis databáz, opis jednotlivých programov pre prácu s databázami a v neposlednom rade sa venuje aj samotnej aplikácii.

Pri výbere témy rozhodol hlavne fakt, že sa rád venujem práci s databázami a je to zaujímavá oblasť informatiky. V doterajšom štúdiu v odbore aplikovaná informatika sme sa práci s databázami venovali a tak som mohol využiť v práci aj tieto získané poznatky a vedomosti.

# 1. Prehľad o súčasnom stave riešenej problematiky

## 1.1. Databázy

Databáza je súbor údajov, údajových tabuliek a iných objektov, ktoré sú usporiadané a organizované tak, aby podporili vykonávanie špecifických požiadaviek, ako sú plynulé vyhľadávanie, usporiadanie a vzájomná kombinácia údajov.

Databáza je súbor informácií, ktoré sú organizované tak, aby mohli byť ľahko prístupné, upravované a aktualizované. V práci na počítači, databázy sú niekedy klasifikované podľa ich organizačného prístupu. Najčastejšie sa používajú relačné databázy, tabuľkové databázy, v ktorých sú dáta definované tak, že môže byť reorganizovaná a prístup v mnohých rôznych spôsoboch. Distribuovaná databáza, je tá, ktorá môže byť rozptýlená alebo sa dá replikovať medzi rôznymi bodmi siete. Objektovo-orientovaná programovaná databáza je tá, ktorá sa zhoduje s údajmi definovanými v objektoch tried a podtried.

Počítačové databázy typicky obsahujú zhrnutie údajov, dokumentov alebo súborov, ako sú predajné transakcie, katalógy a súpisy a zákaznícke profily. Typicky správca databáz poskytuje užívateľom možnosti ovládania pre čítanie a zápis s uvedením generovania správ a analýzy použitia. Databázy a databázoví manažéri sú prevládajúci vo veľkých sálových systémoch, ale sú prítomní aj v menších distribuovaných pracovných staniciach a mid-range systémy, ako je AS/400 a na osobných počítačoch. SQL je štandardný jazyk pre vytváranie interaktívnych otázok a aktualizácií databáz DB2 od IBM, Microsoft SQL Server, a databázové produkty od Oracle, Sybase, a Computer Associates.[14]

### 1.1.1. Typy databáz podľa obsahu

Databázy sa podľa svojho obsahu delia na faktografické, bibliografické a fulltextové.

**Faktografické databázy:** obsahujú fakty, údaje, čísla, časové rady, grafy, snímky (letecké, makroskopické apod.).

**Bibliografické databázy:** obsahujú len záznamy článkov či publikácií, často

formou citácií a údaje o ich obsahu (kľúčové slová, abstrakty). Užívateľ si musí plné texty primárnych zdrojov dohľadať inde (podobne ako bežný knižný katalóg)

**Plno textové databázy:** obsahujú nielen citované údaje, ale predovšetkým neskrátené zdrojové texty, ktoré je často možné fulltextovo prehľadávať.

### 1.1.2. Rôzne typy databáz

#### 1.1.2.1. Relačné databázy

Relačné databázy sa vyznačujú predovšetkým tým, že sú relatívne jednoduché, ľahko pochopiteľné a vďaka tomu rozšírené a medzi programátormi a vývojármi informačných systémov veľmi obľúbené. Na bežné aplikácie kancelárskeho typu sú väčšinou postačujúce.

#### 1.1.2.2. Objektovo orientované databázy

Základ na objektovom programovaní. Základom objektovo orientovanej databázy nie je tento krát tabuľka, ale objekt. Každý objekt má atribúty/vlastnosti (tu je vidieť analógia so stĺpcami v tabuľke) a metódy, ktoré nejakým spôsobom manipulujú s hodnotami vlastností. Jednotlivé "záznamy" sú inštancie objektu s konkrétnymi hodnotami (v relačných databázach - 1 riadok). Dajú sa tu využiť všetky výhody dedičnosti (a to i mnohonásobné), zapúzdrenosti a polymorfizmu. Vďaka tomu objektovo orientovaná databáza výrazne rozširuje možnosti tvorby databázových aplikácií.

#### 1.1.2.3. Deduktívne databázy

Sú to typy aplikácií, s ktorými sa bežný smrteľník už vôbec nestretne. Tieto databázy sú nasadzované napr. pre vedecké účely.

Stavebným kameňom deduktívnych databáz sú relácie (napevno definované vzťahy medzi entitami) a deduktívne pravidlá.

#### 1.1.2.4. Temporálne databázy

Je potrebné spracovávať staré a zároveň nové údaje. Jedná sa o burzy, poisťovne,

banky, rezervačné systémy apod. Všetky predchádzajúce typy databáz, ako sme ich uviedli, sú netemporálne alebo časovo nezávislé. U nich totiž platí, že dáta sú všetky platné v dobe dotazu. Do databázy zavedieme napr. systém tzv. časových pečiatok. Každý údaj bude mať svoje časové označenie, podľa ktorého budú aplikácie schopné rozpoznať, či sa jedná o údaj nový alebo starý.[17]

## 1.2. Vývojové prostredie Delphi

Delphi programovací jazyk je nabitý funkciami, ktoré vám pomôžu s vytváraním OOP aplikácií. Všetky triedy majú pôvod v TObject-e. TObject zavádza metódy, ktoré používajú základné správanie, ako je konštrukcia, deštrukcia a spracovanie správ. [3]

- Delphi je vizuálne programové prostredie, ktoré slúži na vývoj aplikácií pre OS Windows (podobne ako Visual Basic, C++Builder a pod.) - takéto prostredia zjednodušujú programovanie Windows aplikácií - vzhľad aplikácie pod Windows môžeme zostavovať z ponuky komponentov, pričom samotné vývojové prostredie (napr. Delphi) pomáha pri začleňovaní zodpovedajúceho kódu
- takéto moderné programovacie jazyky umožňujú začať programovať objektovým prístupom – je to metodológia, bez ktorej sa v súčasnosti nezaobídeme pri práci v tíme, pri práci s veľkými projektmi, pri tvorbe Windows aplikácií
- v Delphi sa programuje moderným programovacím jazykom **objektovým pascalom** - tento vychádza z Turbo Pascalu, ale pritom si treba uvedomiť, že
  - pascal sa v informatickej komunite považuje za "esperanto" programovacích jazykov
  - je ideálny jazyk ako úvodný pri vyučovaní programovania - môžeme sa v ňom naučiť veľmi efektívne programovať modernými metódami
  - kto ovláda pascal, ľahko neskôr prejde na iné procedurálne programovacie jazyky
- Delphi poskytujú veľmi dobré ladiace prostriedky.[7]

### 1.2.1. Typy súborov v Delphi

Delphi programy obsahujú 3 hlavné súbory s príponami .dpr .pas .dfm.

Project1.**dpr**- delphi projekt(textový súbor= hlavný „pascalovský“ program)- zatiaľ je pre nás nezaujímavý.

Project1.**res**- obsahuje len ikonu výsledného programu.

Project1.**dof**- nastavenia kompilátora(textový súbor= možno si ho pozrieť v textovom editore).

Project1.**dsk**- rozloženie okien a súborov počas práce Delphi(textový súbor).

Project1.**exe**- spustiteľný program.

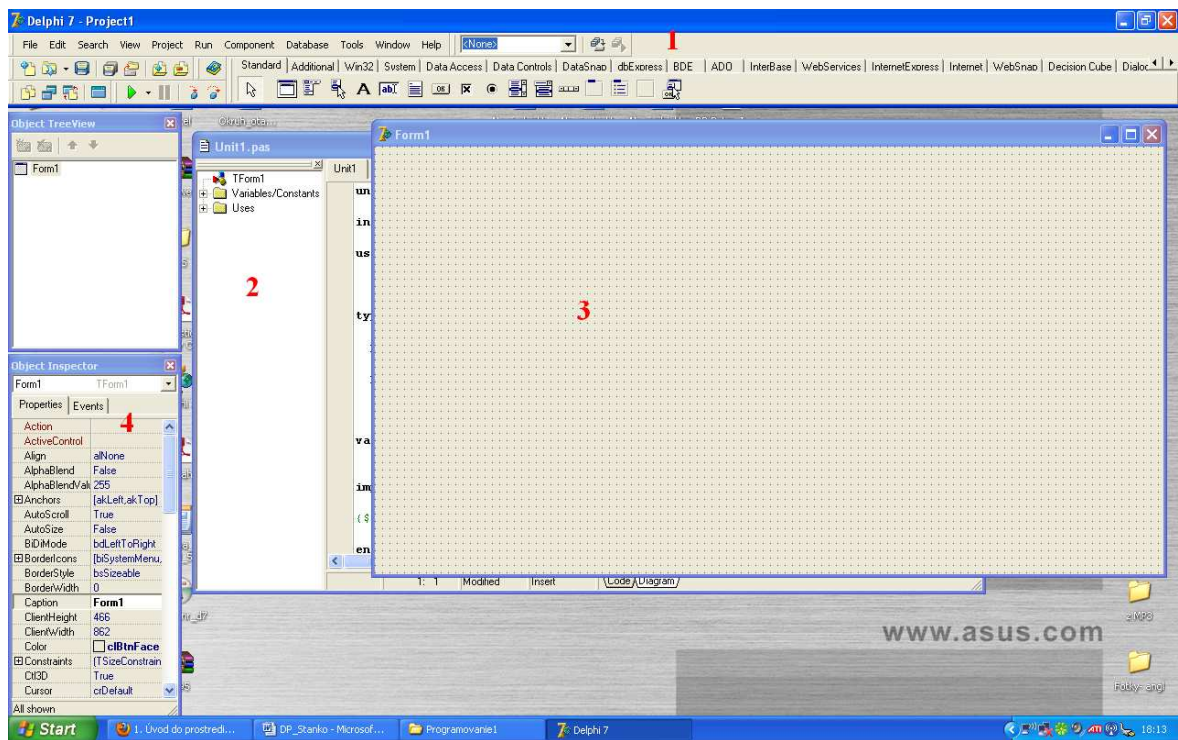
Unit1.**pas**- programový popis formuláru, čo a ako má formulár robiť= podobá sa na klasický „pascalovský“ program.

Unit1.**dfm**- vnútorná definícia formuláru= aké a kde sú komponenty na formulári(napr. Button 1, Image1...).

Unit1.**dcu**- preložený Unit1.pas(v starom pascale.tpu).

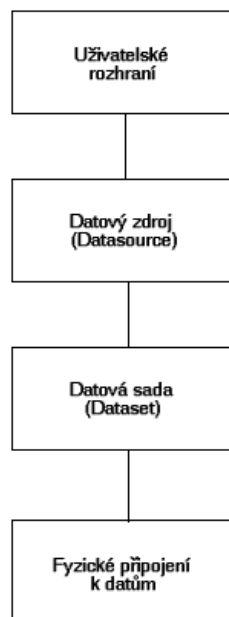
Každý formulár(Form) musí mať Unit. Unit bez formulára môže existovať, keďže nesie iba kód. Súbor .dfm má špeciálny pseudokódovaný tvar. Na jeho zviditeľnenie môžeme použiť skratku Alt+F12. Ak vkladáme objekty na formulár, súbor .dfm sa vytvára sám.

[8]



Obrázok 1 Prostredie Delphi 7

Po spustení programu Delphi 7 vidíme okno, ktoré je na obrázku 1. Číslom 1 je označená paleta komponentov, číslo 2 je časť editovacieho okna, v ktorom píšeme celý program- kód. Číslo 3- prázdny formulár na ktorý vkladáme jednotlivé komponenty. Číslo 4- tzv. objekt inšpektor, na ktorom meníme vlastnosti jednotlivých komponentov na formulári.



**Obrázok 2** Architektúra databáz v Delphi [7]

### 1.3. Tabuľkový procesor

Tabuľkový procesor je program, ktorý premení obrazovku počítača na list papiera, na ktorý možno do ľubovoľného miesta zapisovať údaje rôznych typov ale aj príkazy na ich spracovanie pomocou vzorcov. Umožňuje pracovať s myšou alebo s klávesnicou počítača. Všetko, čo máme zobrazené na obrazovke môžeme vytlačiť na pripojenej tlačiarni (režim WISIWIG).

Zopár najpoužívanejšie vlastností, ktoré z neho robia veľmi silný prostriedok:

- automatické formátovanie buniek podľa toho, akú hodnotu do bunky vložíme. Po vložení dátumu je napríklad bunka trvalo definovaná formátom dátum až do chvíle, kedy tento formát nezrušíme.
- na spôsob označovania premenných (buniek) napr. A1 čo znamená, že premenná leží v stĺpci A, v riadku 1 alebo R1C1 (row 1, collum 1). Je však možné priradiť premennej aj ľubovoľný vlastný názov.
- jedna bunka môže obsahovať súčasne niekoľko informácií: hodnotu, vzorec, formát, poznámku.
- časti obrazovky je možné podľa potreby skryť, zakázať do nich zápis alebo takéto činnosti povoliť po napísaní hesla



- dá sa využiť rozdiel medzi relatívnymi a absolútnymi adresami
- možnosti vytvárať rôzne číselné alebo dátumové rady ale aj vkladanie vlastných textových postupností
- tvorba grafov z rôznych typov dát . Typ grafu je možné vybrať z veľkého množstva preddefinovaných typov , ktoré možno dodatočne upravovať a dopĺňať . Dáta v grafe sú priamo prepojené s tabuľkou , takže akákoľvek zmena hodnôt v tabuľke má za následok zmenu grafu.
- možnosti vytvárať tzv. makrá. Sú to programy priamo prepojitelné s objektmi tabuľkového procesora v jazyku konkrétneho tabuľkového procesora. Je to jazyk na úrovni Pascalu alebo Basicu.[14]

Na základnú myšlienku tabuľkových procesorov – vytváranie, vyplňanie a prepočítavanie tabuliek automatizovaným spôsobom na počítači – prišiel roku 1977 študent Harvardskej univerzity R. Birkin so svojím priateľom a počítačovým odborníkom R. Franklinom. Program, ktorý vytvorili, našiel široké uplatnenie v mnohých odboroch ľudskej činnosti. Autori ho začali predávať pod obchodným názvom *VisiCalc* pre osobné počítače Apple. Neskôr sa tejto myšlienky chytila firma LOTUS a vyvinula profesionálny tabuľkový procesor s názvom LOTUS-1-2-3. Postupne aj ostatné softwarové firmy prinášali na trh svoje produkty ako SuperCalc, Multiplan, QuatroPro, Quatro Pro, Framework a iné.[19]

Programom tohto druhu sa v anglicky hovoriacich štátoch začalo hovoriť – spreadsheet [spredšít], čo vo voľnom preklade znamená rozprestretá plocha alebo zošit na veľkej ploche. Inak v slovenčine sa ujal názov – Tabuľkový kalkulátor alebo procesor.

Dnes je už tabuľkový procesor komplexným nástrojom pre výpočty akéhokoľvek druhu. U nás asi najrozšírenejším a najpoužívanejším programom tohto druhu je program EXCEL z balíka *OFFICE* od firmy Microsoft.

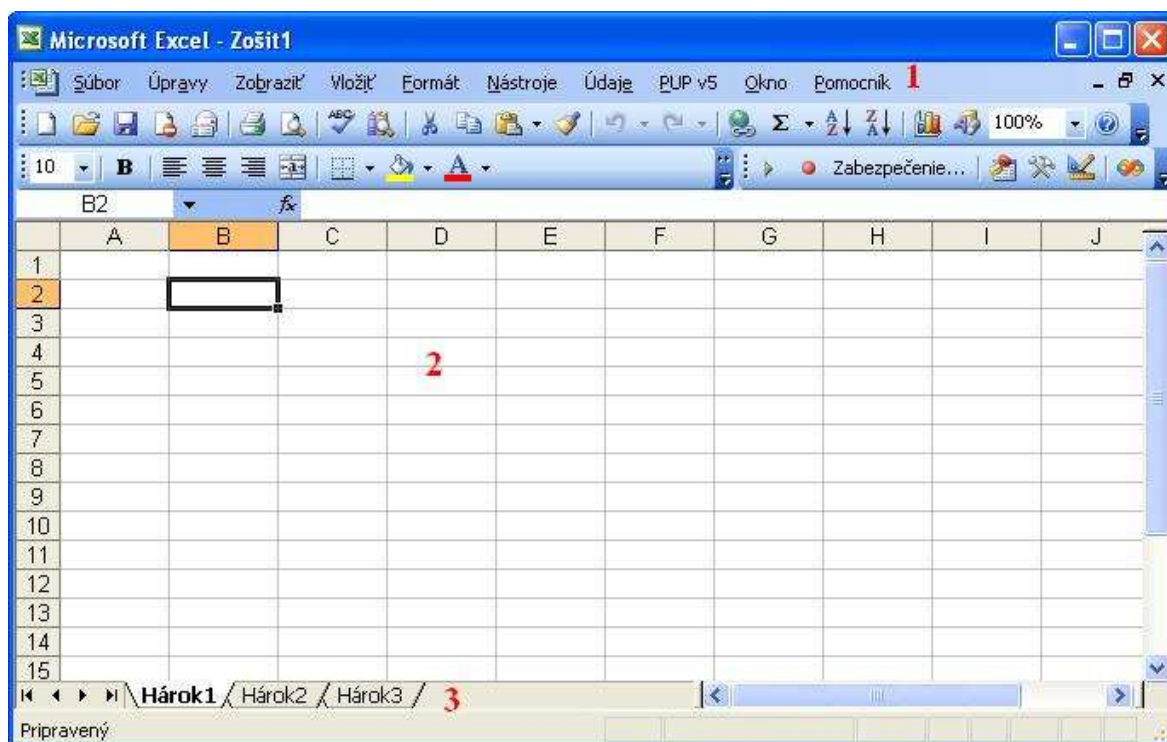
### **1.3.1. Microsoft Excel**

Program Microsoft Excel 2010 umožňuje analyzovať, spravovať a zdieľať informácie viacerými spôsobmi ako doteraz a vďaka tomu používateľom umožňuje prijímať inteligentnejšie rozhodnutia. Nové nástroje na analýzu a vizualizáciu pomáhajú sledovať a zvýrazňovať dôležité údajové trendy a prístup k dôležitým údajom získate takmer z ľubovoľného webového prehľadávača alebo telefónu, ktorý tento typ práce

s údajmi podporuje. Dokonca máte možnosť odovzdať súbory na web a pracovať na nich s ďalšími používateľmi v režime on-line. Bez ohľadu na to, či vytvárate finančné zostavy alebo spravujete osobné výdaje, program Excel 2010 vám ponúka širšiu flexibilitu a efektívnosť pri dosahovaní stanovených cieľov.

Program Excel 2010 ponúka výkonné nové funkcie a nástroje, ktoré slúžia ako pomôcka pri objavovaní vzoriek alebo trendov, ktoré môžu viesť k informovanejším rozhodnutiam a môžu zlepšiť schopnosť používateľa v oblasti analýzy veľkých množín údajov. [13]

- Získame vizuálny súhrn údajov pomocou miniatúrnych grafov, ktoré sa zmestia do bunky spolu s údajmi – použijeme nové krivky.
- Rýchlo a intuitívne filtrujeme veľké objemy informácií pomocou novej funkcie rýchleho filtra a vylepšujeme vizuálnu stránku analýzy v kontingenčnej tabuľke alebo kontingenčnom grafe.



**Obrázok 3** Pracovné okno programu MS Excel

Bod 1- Panel nástrojov

Bod 2- Aktuálny pracovný list- „Hárok1“

Bod 3- Všetky listy zošita

Najväčšie výhody MS Excel:

- vysoká presnosť spracovávaných výpočtov ( rozsah čísel je  $\pm 1.798.10^{+308}$ ).
- zobrazovanie čísel s rozsahom 15 platných cifier.
- externé odkazy na bunky.
- práca s nesúvislými oblasťami.
- organizovanie tabuľkových súborov na zošity s listami.
- veľké množstvo zabudovaných matematických, štatistických, zaokrúhľovacích a maticových funkcií, ale aj vybraných funkcií použiteľných v inžinierskej praxi. Má zabudované aj funkcie pre komplexnú aritmetiku a fourierovu transformáciu.
- použitie riešiteľa (Solver) na riešenie problémov pomocou iteračných metód.
- možnosť sprístupniť podprogramy, uložené v externých dynamicky linkovaných knižniciach (Dynamic Link Libraries - DLL).
- odovzdávanie hodnoty z a do iných programov použitím dynamickej výmeny dát (Dynamic data Exchange -DDE ).
- vkladanie a pripájanie objektov (Object linking and Embedding - OLE).
- tvorba HTML dokumentov. [1]

#### 1.4. Microsoft Access

Čo je to databáza? Sčítané a podčiarknuté, je to organizovaný zber dát. Databázový systém, ako je Access, FileMaker Pro, Oracle alebo SQL Server nám poskytuje softvérové nástroje, ktorými potrebujeme usporiadať dáta pružným spôsobom. Údaje môžeme pridať, zmeniť alebo odstrániť z databázy, môžeme vytvárať dotazy (otázky) na údaje uložené v databáze a vytvárať správy, ktoré obsahujú vybraný obsah.

Program MS Access je relačný databázový systém. Umožňuje pracovať so záznamami nielen textovej povahy, ale aj so zvukom, obrázkom, hypertext. odkazom,

t.j. s akýmkoľvek iným objektom (technológia OLE). V súbore môžeme dáta rozdeliť do tzv. **tabuliek**; dáta v tabuľke môžete prehliadať, pridávať a aktualizovať pomocou formulárov; požadované dáta môžete hľadať a načítať pomocou dotazov; pomocou zostáv ich môžeme analyzovať alebo tlačiť v určitom rozvrhnutí.

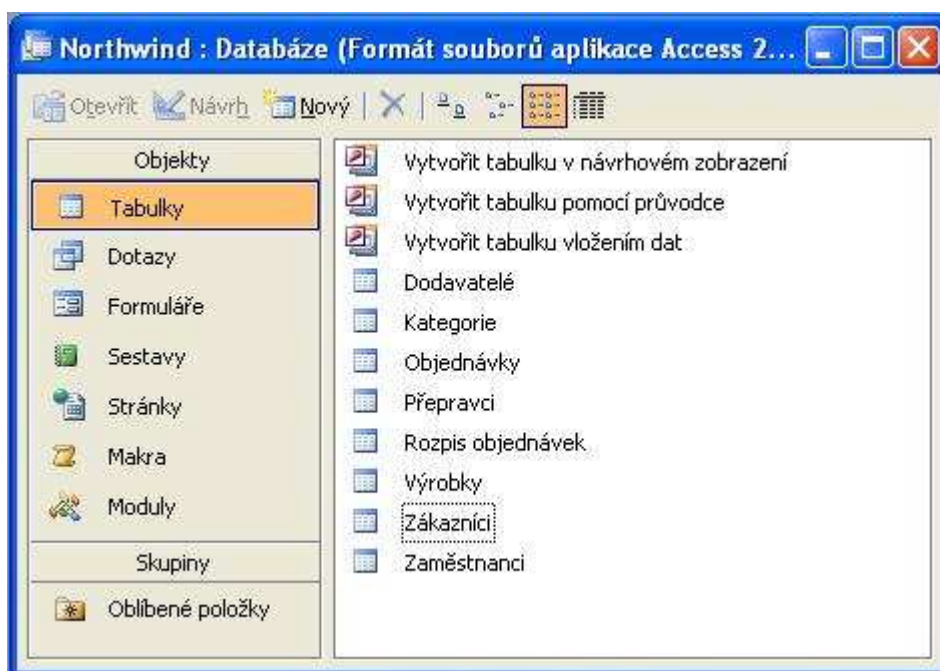
Pri ukladaní dát môžeme vytvoriť pre všetky typy sledovaných informácií jednotlivé tabuľky. Ak chceme spojiť dáta z viacerých tabuliek v dotaze, vo formulári, alebo v zostave, musíme definovať **relácie** medzi tabuľkami.

**Formulár** – používame vtedy, ak chceme prezerat', zadavat' alebo menit' dáta priamo v tabuľke. Formulár nám umožňuje inak usporiadať údaje na obrazovke. Môžeme ho vytvoriť pomocou Sprievodcu formulárom, alebo pomocou novo vytvoreného rozvrhnutia. Je možné zobraziť údaje aj z viacerých prepojených tabuliek.

**Dotaz** – používa sa k výberu určitých záznamov podľa prania užívateľa, používajú sa k spojovaniu údajov z rôznych tabuliek. Dotaz je samostatný objekt a je možné ho opakovane použiť (t.j. používame ho pri hľadaní a načítaní dát splňujúcich stanovené podmienky, vrátane dát z viacerých tabuliek). Dotazom možno aktualizovať či odstrániť viac záznamov naraz a previesť zabudované alebo vlastné výpočty dát.

**Filter** – niečo ako dotaz, používa sa k jednorazovému výberu údajov z tabuliek.

**Zostavy** – používame vtedy, ak chceme analyzovať dáta a prezentovať ich určitým spôsobom v tlačenej forme. Môžete napríklad vytlačiť zostavu, v ktorej sú zoskupené dáta a vypočítané celkové súčty, a inú zostavu, v ktorej sú sformátované dáta, ktoré budú vytlačené ako poštovné štítky.[2]



**Obrázok 4** Úvodné okno programu Access

Na obrázku vidíme v ľavom stĺpci objekty, medzi ktoré patria tabuľky, dotazy, formuláre, tlačové zostavy a iné.

## 1.5. Spoločnosť Semecs, s.r.o.

Spoločnosť Semecs, s.r.o. so sídlom vo Vrábľoch sa zameriava na výrobu elektrických strojov a prístrojov. Prevládajúca činnosť je montáž elektronických celkov a príbuzných výrobkov. Medzi hlavné práce patrí osádzanie dosiek plošných spojov axiálnymi a radiálnymi SMD automatmi, automatické spájkovanie aj v ochranej atmosfére dusíka, testovanie výrobkov na ICT, testovanie na funkcionálnych a zahorovacích testeroch.[16]

## **2. Cieľ práce**

Cieľom diplomovej práce je vytvorenie aplikácie premiestňujúcej údaje z programu Microsoft Excel do programu Microsoft Access a naopak. Riešenie pomôže užívateľovi vybrať údaje z databázy Access a vloží ich do zošita Excelu a rovnako aj v opačnom smere. Tým prepojí existujúce typy databáz v spomínaných typoch a prevedie údaje medzi nimi. Hlavný cieľ je program, v ktorom si užívateľ vyberie databázu na konvertovanie a typ, do ktorej ju chce uložiť. Dôležité je, aby program bol jasný, prehľadný a jednoduchý na používanie.

Nakoniec aplikáciu otestujeme, čím potvrdíme jej funkčnosť.

### **3. Metodika práce**

Samotná vlastná práca sa bude zaoberať tvorbou aplikácie pre prácu s údajmi.

#### **3.1. Požiadavky na riešenie diplomovej práce**

K hlavným požiadavkám patria najmä, aby vytvorená aplikácia prepojila existujúce databázy a menila(konvertovala) údaje bez akejkolvek straty medzi nimi. Hlavne je treba vybrať vhodný softvér na vytvorenie aplikácie. Medzi základné požiadavky na riešenie patrí jednoduchosť a komplexnosť riešenia. Čím jednoduchšie bude riešenie, tým menšie budú požiadavky na technické a softvérové vybavenie a zníži sa aj časová náročnosť na vykonávané úkony. Program musí byť stabilný, aby nedochádzalo k jeho častým pádom a tak zbytočným stratám informácii.

#### **3.2. Postup vytvorenia aplikácie**

Prvým a dôležitým bodom je naštudovanie programov MS Excel a MS Access. Hlavne typy a štruktúry údajov v databázach. Ďalej vyberieme softvér vhodný na riešenie nášho problému, teda taký, ktorý bude schopný pracovať s oboma typmi bez akejkolvek straty. Na vypracovanie aplikácie vyberieme program Delphi, ktorý je so svojimi komponentmi vhodný na riešenie. Aplikáciu si rozdelíme na menšie časti-procedúry. Jedna bude načítavať údaje z programu Excel, ďalšia z programu Access, iná bude ukladať údaje do Excelu a posledná do Accessu. Ostatná časť programu sa bude venovať práci a zápisu údajov.

#### **3.3. Použité nástroje a programy**

Pre prácu je potrebný nasledujúci hardvér a softvér:

- Notebook značky ASUS A6R Series s procesorom Intel Celeron s frekvenciou 1,86 GHz s operačným systémom MS Windows Home Edition.
- Programovací softvér Delphi 7 Enterprise- aplikácia bude tvorená práve v tomto programe.
- Voľne šíriteľné komponenty pre Delphi 7.

- Programový balík MS Office 2010- z neho využijeme najmä programy Excel a Access.

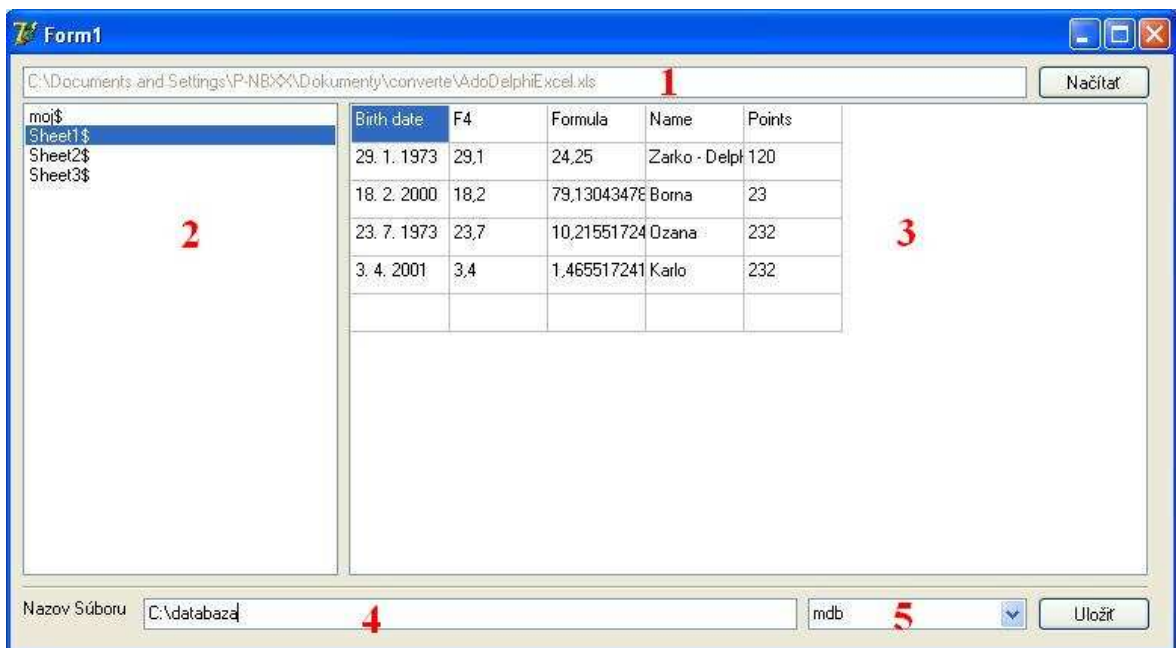


## 4. Vlastná práca a dosiahnuté výsledky

Vlastná práca obsahuje aplikáciu vytvorenú vo vývojovom prostredí Delphi. Tento program je spustiteľný vo všetkých operačných systémoch Windows. Samotnú aplikáciu sme rozdelili do viacerých častí- procedúr.

### 4.1. Úvodný formulár aplikácie

Po spustení aplikácie sa používateľovi zobrazí okno(na obrázku).



Obrázok 5 Hlavné okno aplikácie

V edite označenom číslom 1 sa zobrazuje adresa súboru, ktorý chceme skonvertovať na opačný typ a ktorý sme načítali po stlačení tlačidla „Načítať“. V okne číslo 2 sa zobrazuje počet listov, ktoré obsahuje otvorený zošit Excelu, prípadne tabuľky Accessu. V okne 3 sa nám zobrazuje aktuálna tabuľka, ktorú sme si vybrali kliknutím v okne 2. V edite číslo 4 uvádzame adresu súboru, ktorý chceme uložiť a vo výbere 5 volíme mdb alebo xls, teda opačný typ ako ten otvorený.

Program bude prenášať údaje v 2 smeroch:

- Z typu XLS do typu MDB

- Z typu MDB do typu XLS

## 4.2. Tvorba aplikácie

Po spustení úvodného formulára sa príkazom *DatabaseBuffer := TDatabase.Create()* vytvorí vyrovnávacia pamäť pre tabuľky.

Celú prácu začíname kliknutím na tlačidlo „Načítať“. Pomocou Opendialogu načítame súbor, ktorý chceme previesť na opačný. Ak je to súbor s koncovkou .xls, volá sa funkcia *LoadXlsData*, ak naopak súbor s koncovkou .MDB je volaná funkcia *LoadMdbData*. Tu môže program pokračovať dvoma smermi.

### 4.2.1. Načítanie súboru XLS

Najprv vytvoríme prázdnu databázu. Do nej pomocou komponentu ADO urobíme pripojenie načítaného súboru. Ako zdroj použijeme súbor z adresy 1 na vyššie uvedenom obrázku.

```
ado := TADOConnection.Create(nil);
ado.Connected:=False;
ado.ConnectionString:='Provider=Microsoft.Jet.OLEDB.4.0;' + 'Data Source=' +
Edit1.Text + ';' + 'Extended Properties=EXCEL 8.0';
ado.LoginPrompt:= false;
```

Pomocou OLE modulu vytvoríme spojenie s Excelom. Načítame zoznam tabuliek.

```
for i := 0 to dbTables.Count-1 do
begin
if (Pos('$', dbTables[i]) = 0)then continue;
t := dbTables[i];
table:= TStringGrid.Create(self);
table.HelpKeyword:= dbTables[i];
```

Načítame všetky tabuľky a uložíme ich do TDatabase objektu. Načítame ďalej všetky názvy stĺpcov a načítame všetky údaje z tabuľky.

```
try
ADOQuery := TADOQuery.Create(nil);
try
ADOQuery.Connection := ado;
t:= 'Select * from [' + dbTables[i] + ']';
ADOQuery.SQL.Text := 'Select * from [' + dbTables[i] + ']';
ADOQuery.Open;
```

```

k:=1;
while not AdoQuery.eof do
begin
for j := 0 to cols.Count-1 do
begin
table.RowCount:= k+1;
table.Cells[j,k] := VarToStr(ADOQuery.FieldValues[table.Cells[j,0]]);
end;
k := k + 1;
AdoQuery.Next;
end;

```

Pomocou dotazu *Query* sa vyberie pripojenie. Postupne načítavame údaje až pokiaľ nie je koniec súboru. Prvé riadky v stĺpcoch v Exceli sú u nás rovnako prvé riadky v stĺpcoch. Príkazom *db.Add(table)* pridáme tabuľku do našej štruktúry. Keď je koniec načítavaného súboru, končí aj funkcia.

#### 4.2.2. Zápis údajov do súboru mdb

Keďže sme načítali súbor typu .xls zapisovať budeme do súboru .mdb, teda do programu MS Access. Na toto ukladanie máme procedúru *SaveMdbData* pracujúcu s názvom typu string a databázou- db typu TDatabase. Začiatkom sa inicializuje práca s Com objektmi. Využitím OLE vytvárame novú prázdnu databázu. Nový súbor aplikácie Access vytvárame príkazom *access:= CreateOleObject('Access.Application')*. Pomocou komponentu ADO vytvoríme spojenie, to následne otvoríme. Prechádzame všetky načítane tabuľky uložené v stringgrid v Delphi a ukladáme ich do novovytvorených v Access. Pridáme názvy stĺpcov tabuliek, upravíme a dokončíme query- dotazy.

```

for i := 1 to db.Tables[d].RowCount-1 do
begin
queryText := 'INSERT INTO '+Replace(db.Tables[d].HelpKeyword, '$', '')+'
('+rowsQueryText+') VALUES(';
for j := 0 to db.Tables[d].ColCount-1 do
begin
queryText := queryText + " + QuotedStr(db.Tables[d].Cells[j, i]) + ',';
end;
delete(queryText, length(queryText), 1);
queryText := queryText + ');';
ado.Execute(queryText);

```

V uvedenom zdrojovom kóde je cyklus na vkladanie údajov do tabuľky, pridávame údaje po riadkoch a následne odošleme query- dotaz. Pokiaľ sa počas práce nevyskytla

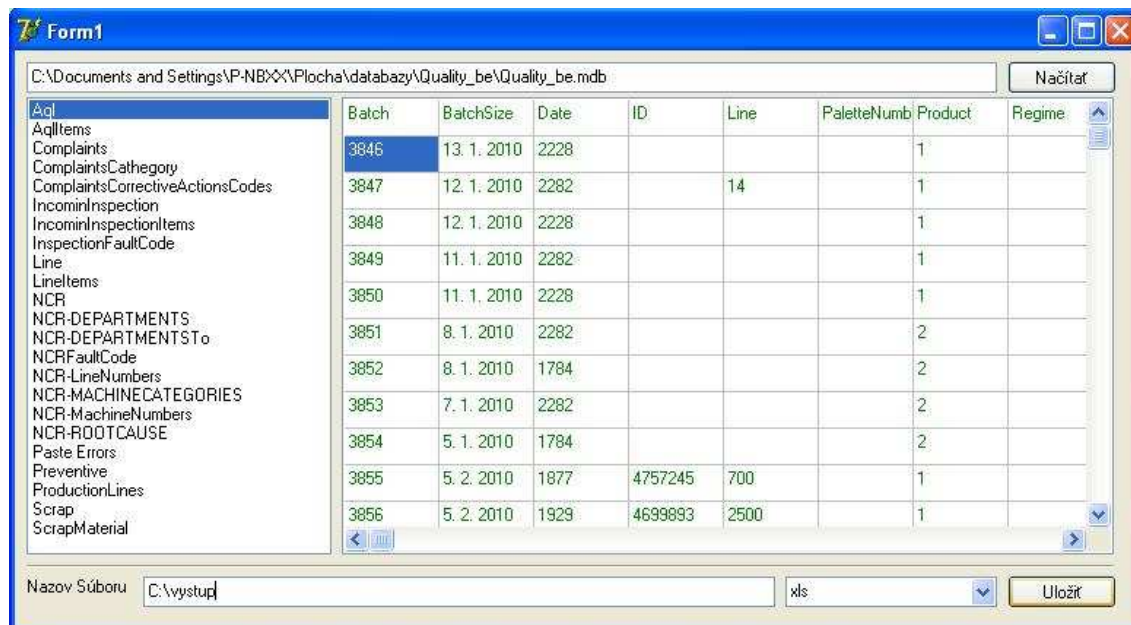
chyba, údaje z Excelu sú uložené v programe Access.

### 4.2.3. Načítanie súboru mdb

Táto časť je obsiahnutá v procedúre *LoadMdbData*. Na začiatku vytvoríme prázdnu databázu. Cez ADO vytvoríme pripojenie, načítame reťazec nastavený v edite v úvodnom okne. Načítame všetky tabuľky nazvané dbTables a uložíme ich do TDatabase objektu. Nasledujúcim príkazom načítavame názvy stĺpcov.

```
ado.GetFieldNames(dbTables[i], cols);
```

Ďalej pokračujeme podobne ako pri práci so súborom Excelu. Teda prácou s *ADOQuery* zistíme všetky údaje. Prechádzame celý súbor pokiaľ nebude koniec, označený EOF(End of File). Keď narazíme na koniec, zavrieme *Query* a pridáme tabuľku do našej štruktúry.



Obrázok 6 Okno načítania databázy Access

### 4.2.4. Zápis údajov do súboru xls

Údaje zo stringgridu do Excelu ukladáme procedúrou *SaveXlsData*. Na začiatku sa inicializuje práca s objektmi.

```
xls := CreateOLEObject('Excel.Application');
```

```
wb := xls.Workbooks.Add;
```

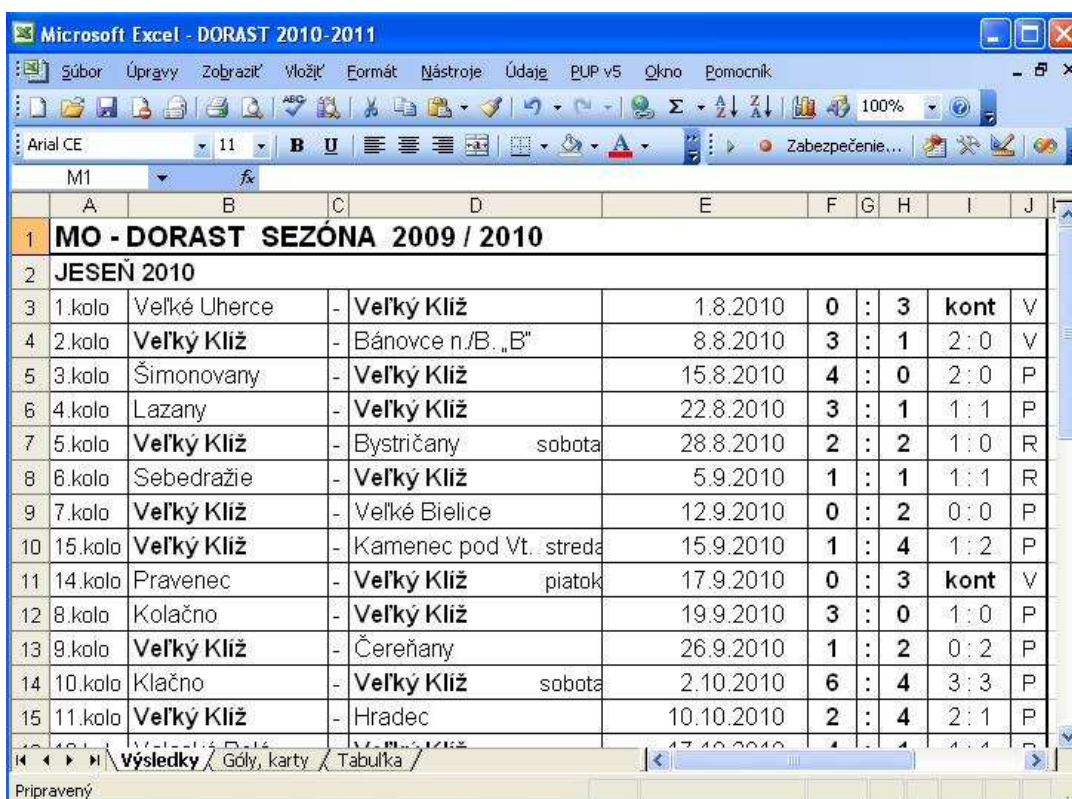
Uvedenými príkazmi vytvoríme prázdny OLE objekt, teda náš súbor Excelu. Do neznámej *wb* pridáme prázdny zošit.

```
for d := 1 to xls.Workbooks[1].WorkSheets.Count-1 do  
xls.Workbooks[1].WorkSheets[d].Delete;
```

Zmažeme pôvodné listy v pracovnom zošite Excelu, ale aspoň jeden musí zostať. Prechádzame všetky tabuľky od 0 až po počet-1 a ukladáme ich do pracovných listov zošita. Zoberieme prvý list, alebo ak neexistuje vytvoríme nový. Nastavíme meno listu podľa mena tabuľky z Accessu a vložíme údaje zo stringgridu formulára. Napokon uložíme ako súbor s koncovkou .xls, teda súbor Excelu. Na záver uvoľníme súbor a ukončíme prácu s com objektmi.

#### 4.2.5. Kontrola funkčnosti aplikácie

Pre kontrolu programu sme zvolili ľubovoľnú tabuľku typu Excel (obrázok 7). Sú to výsledky futbalových zápasov. Túto tabuľku sme načítali v našej aplikácii.



The screenshot shows a Microsoft Excel spreadsheet titled "DORAST 2010-2011". The spreadsheet contains a table of football match results for the 2009/2010 season. The table has columns for round number, home team, away team, date, goals scored by home team, goals scored by away team, and the final score. The matches are listed from round 1 to 15, with some rounds having multiple matches. The table is formatted with bold text for team names and dates, and a 'kont' (goals) column for the final score.

Kolo	Domáci	Hostia	Dátum	Skóre Domáci	Skóre Hostia	Skóre	Stav
1. kolo	Veľké Uherce	Veľký Kliž	1.8.2010	0	3	0 : 3	kont
2. kolo	Veľký Kliž	Bánovce n./B. „B“	8.8.2010	3	1	3 : 1	2 : 0
3. kolo	Šimonovany	Veľký Kliž	15.8.2010	4	0	4 : 0	2 : 0
4. kolo	Lazany	Veľký Kliž	22.8.2010	3	1	3 : 1	1 : 1
5. kolo	Veľký Kliž	Bystričany sobota	28.8.2010	2	2	2 : 2	1 : 0
6. kolo	Sebedražie	Veľký Kliž	5.9.2010	1	1	1 : 1	1 : 1
7. kolo	Veľký Kliž	Veľké Bielice	12.9.2010	0	2	0 : 2	0 : 0
15. kolo	Veľký Kliž	Kamenec pod Vt. streda	15.9.2010	1	4	1 : 4	1 : 2
14. kolo	Pravenec	Veľký Kliž piatok	17.9.2010	0	3	0 : 3	kont
8. kolo	Kolačno	Veľký Kliž	19.9.2010	3	0	3 : 0	1 : 0
9. kolo	Veľký Kliž	Čereňany	26.9.2010	1	2	1 : 2	0 : 2
10. kolo	Klačno	Veľký Kliž sobota	2.10.2010	6	4	6 : 4	3 : 3
11. kolo	Veľký Kliž	Hradec	10.10.2010	2	4	2 : 4	2 : 1

Obrázok 7 Vstupná tabuľka v MS Excel

Výsledkom je súbor typu Access s nami zvoleným názvom „tabuľka“.

The screenshot shows the Microsoft Access interface with a data table displayed. The table has the following columns: MO - DČ, F2, F4, F5, F6, F7, F8, F9, and F1. The data rows represent individual rounds of a tournament, including details like round number, location, date, and scores.

MO - DČ	F2	F4	F5	F6	F7	F8	F9	F1
JESEŇ 2010								
1.kolo	Veľké Uherce	- Veľký Klíž	1. 8. 2010	0	:	3	kont	V
2.kolo	Veľký Klíž	- Bánovce n./B.	8. 8. 2010	3	:	1	2:0	V
3.kolo	Šimonovany	- Veľký Klíž	15. 8. 2010	4	:	0	2:0	P
4.kolo	Lazany	- Veľký Klíž	22. 8. 2010	3	:	1	1:1	P
5.kolo	Veľký Klíž	- Bystričany	28. 8. 2010	2	:	2	1:0	R
6.kolo	Sebedražie	- Veľký Klíž	5. 9. 2010	1	:	1	1:1	R
7.kolo	Veľký Klíž	- Veľké Bielice	12. 9. 2010	0	:	2	0:0	P
15.kolo	Veľký Klíž	- Kamenec pod	15. 9. 2010	1	:	4	1:2	P
14.kolo	Pravenec	- Veľký Klíž	17. 9. 2010	0	:	3	kont	V
8.kolo	Kolačno	- Veľký Klíž	19. 9. 2010	3	:	0	1:0	P
9.kolo	Veľký Klíž	- Čereňany	26. 9. 2010	1	:	2	0:2	P
10.kolo	Klačno	- Veľký Klíž	2. 10. 2010	6	:	4	3:3	P
11.kolo	Veľký Klíž	- Hradec	10. 10. 2010	2	:	4	2:1	P

Obrázok 8 Výstupná tabuľka v MS Access

## 5. Záver

Témou diplomovej práce bolo prepojenie a hlavne prenos údajov medzi programami MS Excel a MS Access. Práve tieto dva typy sú dnes veľmi využívané medzi používateľmi so základnými znalosťami práce s počítačom. Tvorba aplikácií zameraných na prenos a konverziu rôznych údajov je v dnešnej modernej a informačnej dobe potrebná. Aj keď je snaha približovať sa čo najviac všeobecným štandardom, stále existujú typy, ktoré nie sú navzájom kompatibilné. Vytvorený program je smerovaný hlavne pre použitie v spoločnosti Semecs s.r.o Vráble, v ktorej prebiehali konzultácie s využitím ich informácií pre vybranú tému. Keďže v menovanej spoločnosti pracujú najmä s údajmi v už spomínaných typoch, je aplikácia vhodná na ďalšie použitie.

Samotná práca bola rozdelená do niekoľkých častí. Úvod práce je venovaný všeobecným informáciám o databázach, práci s údajmi, ale aj programom, ktoré sa tejto problematiky týkajú. Dnes najviac využívané z balíka Microsoft Office pre prácu s množstvom údajov, tabuľkami a štatistikami sú najmä Excel a Access. Ďalšia časť diplomovej práce si všíma program Delphi Enterprise a jeho komponenty, ktoré boli v celej aplikácii využité. Väčšinu programu tvoria procedúry na čítanie a ukladanie informácií. Ich základ je postavený na práci s ADO komponentom a OLE, rovnako sa využívajú aj základné príkazy SQL. Formulár je na pohľad jednoduchý a prehľadný. Po kliknutí sa načíta nami zvolená databáza. Podľa typu súboru nasleduje procedúra načítania údajov buď z Excelu alebo Accessu. Tu sa využíva okrem iného najmä ADO komponent. Okrem tabuliek sa zisťujú aj názvy stĺpcov a samotné údaje, toto všetko je dočasne uložené v vašom formulári. Pri ukladaní je rozdiel hlavne v pomenovaní jednotlivých častí. Pri Exceli pomenujeme v zošite pracovné hárky, ale v Accese sú to tabuľky.

Pri práci sme si zlepšili prehľad a znalosti v práci s databázami. Databázy sú v dnešnej dobe neoddeliteľnou súčasťou každej firmy či spoločnosti. Práve preto je práca s nimi taká dôležitá. Vedieť rýchlo a správne vytvárať tabuľky, reagovať na dotazy či upravovať tlačové zostavy nám môže ušetriť nielen čas, ale aj finančné prostriedky. Možnosť rozšíriť aplikáciu je hlavne v typoch, s ktorými by aplikácia pracovala, tým by sa zvýšila využiteľnosť programu.

## 6. Použitá literatúra

[1] Černý, J. 2008. *Excel 2000-2007*. Praha: Grada, 2008. 192 s. ISBN 978-80-247-2305-1.

[2] Databázový program MS Access.[online]. [cit/ 2011-02-25]. Dostupné na: <<http://www.gymmt.sk/~alica/matury/matury2002/Databaza.htm>>

[3] *Delphi programming*. [online]. [cit. 2011-01-10]. Dostupné na internete: <[http://delphi.about.com/od/windowshellapi/u/advanced\\_delphi.htm#s1](http://delphi.about.com/od/windowshellapi/u/advanced_delphi.htm#s1)>

[4] Frye, C., Freeze, W., *Microsoft Excel 2003 Programming Inside Out*. Redmond. 2004. 594s. ISBN: 0-7356-1985-9 .

[5] Gajic, Z. 2006. *COM Programming with Delphi*. [online]. [cit. 2011-01-23]. Dostupné na: <<http://delphi.about.com/library/weekly/aa121404b.htm>>

[6] HAJNALA, J. 2005. *Delphi 7 - Základy objektového programovacieho jazyka* [online]. Nitra: Gymnázium Párovská 1, 2005. [cit. 2011-02-28]. Dostupné na <<http://www.gymparnr.edu.sk/obsah/predmety/subory/informatika/prostrdelphi.pdf>>.

[7] Kadlec, V., *Databáze v Delphi*. [online]. [cit. 2011-01-27]. Dostupné na internete: <<http://www.zive.cz/clanky/umime-to-s-delphi-75-dil--databaze-v-delphi-obecny-uvod/sc-3-a-110410/default.aspx>>

[8] KADLEC, V. 2007. *Umíme to s Delphi* [online počítačový program]. Ver. 1.1.12.574. 2007 [cit. 2010-12-18]. Dostupné na: <<http://umime-to-s-delphi.wz.cz/download.php>>

[9] Lacko, L. 2001. *Web a databáze*. Praha: Computer Press, 2001. 270 s.

[10] Machová, J. 2004. *Začíname s Delphi*. [online]. [cit. 2011-02-16]. Dostupne na internete:<<http://hq.alert.sk/~mandos/fmfi-k/Informatika/Programovanie/prednasky/pz01.html>>

[11] MS Excel 2007.[citované 2010-12-12]. Dostupné na internete: <[http://www.jfmed.uniba.sk/fileadmin/user\\_upload/editors/Sekretariat\\_Files/Administrativa/ms\\_excel\\_mp\\_2007.pdf](http://www.jfmed.uniba.sk/fileadmin/user_upload/editors/Sekretariat_Files/Administrativa/ms_excel_mp_2007.pdf)>



[12] *New database format enumeration*. Dostupné na internete: <<http://msdn.microsoft.com/en-us/library/bb243739%28v=office.12%29.aspx>>

[13] *Office Excel 2010*. [online]. [cit. 2010-12-16]. Dostupné na: <<http://office.microsoft.com/sk-sk/excel/>>

[14] PC Klub UNSS. 2007. *Excel pre pokročilých*. In *Vizus*. [online]. 2007, vol. 6, no. 2-3 [cit. 2011-01-25]. Dostupné na internete: <[www.pcklubunss.sk/vizus/v2007/Vizus%2023-2007.rtf](http://www.pcklubunss.sk/vizus/v2007/Vizus%2023-2007.rtf)>.

[15] *SearchSQLServer*. 2011. Dostupné na internete: <<http://searchsqlserver.techtarget.com/definition/database>>

[16] Semecs s.r.o. Vráble. Dostupné na: <[http://semecs\\_s\\_r\\_o.sk-firma.com](http://semecs_s_r_o.sk-firma.com)>

[17] Skřivan, J., *Databázy nejsou jen mysql*. Dostupné na: <<http://interval.cz/clanky/databaze-nejsou-jen-mysql/>>

[18] ŠINDELÁŘ, J. 2007. *Tipy a triky v Delphi* [online počítačový program]. Ver. 1.1.13.574. 2007 [cit. 2011-04-03]. Dostupné na: <<http://tipy-a-triky-v-delphi.wz.cz/download.php>>

[19] Tabuľkový procesor Excel. [online]. [cit. 2011-01-13]. Dostupné na: <<http://www.gymmt.sk/~alica/matury/matury2002/Excel1.htm>>

## **7. Prílohy**

### **Príloha A: Elektronický nosič**

Priložené CD obsahuje:

- A.1 Diplomová práca vo formáte pdf
- A.2 Diplomová práca vo formáte doc
- A.3 Aplikácia v Delphi