

Databáze - osnova výuky základů DB

Doc. Dr. Vladimír Homola, Ph.D.

1 Cvičná témata

Při výuce studenti průběžně procvičují přednášená témata na připravených cvičných databázích. Jde o dvě logické úlohy, při výkladu se čerpá z jednoho, ze druhého nebo z obou témat. Autor předpokládá, že z kontextu výkladu nebo zadání resp. řešení úlohy bude zřejmé, o které ze dvou témat jde.

1. Firma má zaměstnance (jméno, datum nástupu, plat). Každý pracuje v jednom z oddělení (název, místo, procento prémie). Oddělení jsou v různých okresech (SPZ, PSČ, počet obyvatel). Soubor ke stažení je [zde](#).

Směřuje se k vytvoření databáze, která odpoví na tyto a podobné otázky:

1. Kteří zaměstnanci mají plat větší než 22.000,- ?
2. Jaká je u těchto zaměstnanců 15% daň z příjmu?
3. Kolik potřebuje každé oddělení na platy svých zaměstnanců?
4. U kterých z těchto oddělení to převyší 300.000,- ?

2. Občas nakupujeme (kdy, co, za kolik). Nakupujeme v různých obchodech (název, sídlo firmy, počet jejich obchodů v republice) různé druhy zboží (název, % DPH). Soubor ke stažení je [zde](#).

Směřuje se k vytvoření databáze, která odpoví na tyto a podobné otázky:

1. Které nákupy byly dražší než 40,- ?
2. Jaká byla u těchto nákupů 15% DPH?
3. Kolik jsem utratil každý den?
4. Které nákupy byly z řetězce, který má nejvíce obchodů?

Poznámka 1: Uvedené cvičné databáze jsou ve formátu ACCDB souborů zpracovávaných např. programem Access 2003-2019. Pro úplné vyzkoušení zmíněným programem je nutno si soubory **stáhnout na vlastní počítač** a otevírat je až tam.

Poznámka 2: Tento článek není výukovým materiálem. Obsahuje pouze heslovitě jednotlivé kapitoly, které jsou tu více, tu méně podrobně ve vlastní výuce probírány. U některých hesel je uveden odkaz na podrobnější studijní materiály.

2 Databáze - základy

2.1 Obecné informace

Databázové systémy pracují nad daty v daném operačním systému a) fyzickým způsobem uloženými, b) logicky organizovanými, c) uživatelsky přístupnými, d) rozumně programovatelnými. Pro kardinální množinu uživatelů současné digitální techniky jde vlastně o hierarchii

- a. datových typů vycházejících z procesorů Intel a jejich klonů,
- b. organizovaných ve formě relačních databází,

- c. přístupných především pomocí SQL,
- d. zpracovávaných také vlastní aplikací zejména objektovými třídami modelujícími databázi.

Výuka sleduje uvedenou hierarchii nejprve na úrovni teoretické (výklad mimo "počítačové" prostředí), poté v praktických krocích na konkrétní demonstrační situaci. Autor tohoto textu se omlouvá čtenářům vyšší úrovně za použití prostředí zhůvěřilého, systémově primitivního, ajťáky proklínaného, značně chybového, ale přece jen velkou částí světa používaného - prostředí MS (Microsoft).

2.2 Úroveň teoretická

2.2.1 Uložení dat, báze dat

Vývoj od zpracování jednotlivých datových hodnot až po systémy řízení báze dat. Stručné studijní materiály [zde](#).

2.2.2 Organizace databází

Hierarchická organizace; logické schéma, uložení dat, pohyb v hierarchické struktuře. Stručné studijní materiály v úvodu [zde](#).

Relační organizace (jen na úrovni popisu); požadavky na data a úrovně jejich zpracování.

2.2.3 Teoretický základ relačních databází

Množina, kartézský součin, relace. Doména vs. pojmenovaný datový typ. Normální formy. Studijní materiály ve druhé části [zde](#).

2.2.4 Datové typy

Průnik datových typů různých relačních databázových systémů. Důvody jejich pestrosti. Studijní materiály [zde](#).

Data číselná, textová, vyjadřující polohu na časové ose, dvouhodnotová. Studijní materiály [zde](#).

2.2.5 Tabulky relační databáze

Teoretický základ vs. praktická realizace: jméno tabulky, atributy sloupce, klíče a vztahy mezi tabulkami.

Tabulky fyzické, virtuální, dočasné. Pohledy, dotazy.

3 Databázový program

3.1 Struktura databáze MDB - ACCDB

Kolekce tabulek, dotazů, formulářů, sestav, modulů. Studijní materiály [zde](#).

Kolekce Relationships (vztahů, propojení, souvislostí, "relací"). Studijní materiály [zde](#).

3.2 Spuštění a prostředí

Access - Office 2010. Logika prostředí, základní funkce, základní činnosti. Studijní materiály [zde](#).

3.2.1 Nová tabulka - návrhové prostředí

Seznam datových polí, seznam vlastností. Studijní materiály [zde](#).

3.2.2 Práce se strukturou tabulky, práce s daty tabulky

Návrhové prostředí vs. datový list. Přepínání prostředí. Úprava struktury, vložení několika řádků dat.

3.2.3 Datový list

Otevření existující cvičné databáze, otevření cvičné tabulky v prostředí datového listu.

Řádky a práce s nimi. Přidávání řádků dat, změna dat, vypouštění řádků dat. Změna výšky řádku. Informace o dotazech typu "Uložit?"

Sloupce a práce s nimi. Označení jednoho a více sloupců, změna šířky sloupce.

Řazení dat; řazení podle jednoho kritéria, podle více kritérií. Řazení vzestupné, sestupné.

Filtrování dat; data číselná, textová, časová. Podmínky výběru: rovnost, relační operátory, vnitřní obsah.

3.2.4 Indexy, klíče

Důvod, geneze (např. řazení, vyhledávání).

Vlastnosti, primární klíč, jeho vytvoření.

Vytvoření jiných klíčů.

3.2.5 Relace

Zde zatím jen přes primární klíč. Návrhové prostředí, vytvoření, nastavení a význam vlastností. Studijní materiály [zde](#).

Vnořený datový list. Studijní materiály [zde](#).

Ověření funkčnosti nastavených vlastností relace.

4 Dotazy

4.1 Návrhové prostředí

Studijní materiály [zde](#).

Nejjednodušší dotaz: 3 pole, filtr, řazení.

Uložení dotazu. Výsledek dotazu vs. návrh dotazu.

Výrazy v dotazech. Studijní materiály [zde](#).

4.2 Podstata dotazu

Dotazovací jazyk SQL, jeho příkaz Select. Při výkladu velmi stručně o SQL. Podrobněji o SQL např. [zde](#).

4.3 Příkaz Select

4.3.1 Jeden datový zdroj, sekvenční procházení

Seznam datových polí (posléze výrazů), jednoduchý filtr na vstupu, řazení.

Z01: Jména, datumy nástupu a plat abecedně řazených zaměstnanců - ale jen těch s platem větším než 22.000,-

```
select JMENO, NASTUP, PLAT
  from ZAMESTNANCI
 where PLAT>22000
 order by JMENO
```

Z02: Jména, datumy nástupu, plat a 15% daň z příjmu abecedně řazených zaměstnanců - ale jen těch s platem větším než 22.000,- Ověření aktualizovatelnosti dat.

```
select JMENO, NASTUP, PLAT, PLAT/100*15
  from ZAMESTNANCI
 where PLAT>22000
 order by JMENO
```

Z03: Definování vlastního jména vypočítávanému sloupci:

```
select ... , PLAT/100*15 as DP_FO ...
```

4.3.2 Jeden datový zdroj, seskupování

Čtení řádků datového zdroje s event. filtrováním na vstupu, z nich vytváření skupin dle zadaného kriteria, a z každé skupiny generování jediného výstupního řádku obsahující zadaná pole a výrazy. AgregáčnÍ funkce ve výrazech. Řazení výstupu. Filtrování na výstupu.

Z04: Kolik potřebuje každé oddělení na platy svých zaměstnanců?

```
select PRACOVISTE, sum (PLAT) as MZDY
  from ZAMESTNANCI
 group by PRACOVISTE
```

U04: Kolik jsem utratil v každém obchodě každý den?

```
select OBCHOD, DATUM, sum (KC) as CELKEM
  from VYDAJE
 group by OBCHOD, DATUM
```

Z05: Kolik potřebuje každé oddělení na platy svých zaměstnanců - ale jen ta oddělení, kde to převyší 300.000,-

```
select PRACOVISTE, sum (PLAT) as MZDY
  from ZAMESTNANCI
 group by PRACOVISTE
 having sum (PLAT) > 300000
```

Z06: Kolik nastoupilo do firmy každý měsíc zaměstnanců?

```
select year (NASTUP) as ROK, month (NASTUP) as MESIC, count (NASTUP)
as POCET
from ZAMESTNANCI
group by year (NASTUP), month (NASTUP)
order by NASTUP
```

4.3.3 Dva datové zdroje, kartézský součin

Sekvenční čtení z kartézského součinu dvou datových zdrojů, filtrování kombinací.

Z07: Jména, datumy nástupu, plat a **název** oddělení.

```
select JMENO, NASTUP, PLAT, NAZEV
from ZAMESTNANCI, ODDELENI
```

Vysvětlení výsledku, zopakování mechanismu generování výstupu, použití filtru. Ověření aktualizovatelnosti dat.

Z08: Jména, datumy nástupu, plat a **název** oddělení.

```
select JMENO, NASTUP, PLAT, NAZEV
from ZAMESTNANCI, ODDELENI
where PRACOVISTE = KOD
```

Diskuse případu stejných jmen v různých tabulkách, prefix datového zdroje.

Z09: Jména, datumy nástupu, plat a **název** oddělení

```
select JMENO, NASTUP, PLAT, NAZEV
from ZAMESTNANCI, ODDELENI
where ZAMESTNANCI.PRACOVISTE = ODDELENI.KOD
```

4.3.4 Alias a jeho použití.

Z10: Jména, datumy nástupu, plat a **název** oddělení.

```
select JMENO, NASTUP, PLAT, NAZEV
from ZAMESTNANCI D, ODDELENI B
where D.PRACOVISTE = B.KOD
```

4.3.5 Více datových zdrojů, kartézský součin

Sekvenční čtení z kartézského součinu více datových zdrojů, filtrování kombinací. Jen popis, bez příkladu.

4.3.6 Pojem Datový zdroj

Jen výklad. Možno i příklad:

Z11: Jména, datumy nástupu, plat a **název** oddělení.

```
select JMENO, NASTUP, PLAT, NAZEV
from (select * from ZAMESTNANCI, ODDELENI where PRACOVISTE=KOD)
```

4.3.7 Jednoduchý příkaz Select v prostředí databázového programu

V naprosté většině složitějších dotazů se návrhové prostředí ukazuje jako zcela nedostatečné. Alespoň základní tvary a použití příkazů SQL musí zvládnout uživatel tohoto prostředí, aby byl schopen získat odpověď na relativně jednoduché dotazy.

Kterí zaměstnanci mají nadprůměrné příjmy? Řešit jen v návrhovém prostředí dotazu. Jako kritérium zadat:

```
> (select avg (PLAT) from ZAMESTNANCI)
```

Druhým případem, kde lze vhodně využít znalosti syntaxe příkazu Select (SQL), je v programu Access možnost definovat ovládací prvek pro některá datová pole obecně ve formuláři, tedy i v datovém listu.

Zajistit, aby v datovém listu bylo možno zadávat kód pracoviště zaměstnanců výběrem z rozvíjecího seznamu (v Accessu CZ přeloženo jako "Pole se seznamem"). Řešit v návrhovém prostředí tabulky ZAMESTNANCI, pole PRACOVISTE, ve VLASTNOSTECH POLE záložka VYHLEDÁVÁNÍ. Jako OVLÁDACÍ PRVEK zvolit POLE SE SEZNAMEM, a vyplnit vlastnost

```
ZDROJ ŘÁDKŮ:      select KOD from ODDELENI
```

Velmi zdařilým (ze strany autorů programu Access) je výsledek řešení následujícího zadání: Zajistit, aby v datovém listu bylo možno zadávat pracoviště zaměstnanců výběrem z rozvíjecího seznamu. Řešit v návrhovém prostředí tabulky ZAMESTNANCI, pole PRACOVISTE, ve VLASTNOSTECH POLE záložka VYHLEDÁVÁNÍ. Jako OVLÁDACÍ PRVEK zvolit POLE SE SEZNAMEM, a vyplnit vlastnosti

```
ZDROJ ŘÁDKŮ:      select KOD, NAZEV from ODDELENI
```

```
POČET SLOUPCŮ:    2
```

```
ŠÍŘKY SLOUPCŮ:    0cm;5cm
```

4.3.8 Dva datové zdroje, spojení (join)

Čerpání ze dvou datových zdrojů BEZ RELACÍ, BEZ KLÍČŮ. BEZ COMBA. Změnit u 5 zaměstnanců pracoviště na X. Nejprve v návrhovém prostředí dotazu (jméno, nástup, plat název oddělení), vlastnosti spojení: všechny zaměstnance, shodná oddělení. Pak vysvětlení toho co chceme s nákresem spojení zleva. Nakonec přechod k SQL:

Z12: Jména, datумы nástupu, plat a **název** oddělení.

```
select JMENO, NASTUP, PLAT, NAZEV
from ZAMESTNANCI left join ODDELENI
on ZAMESTNANCI.PRACOVISTE = ODDELENI.KOD
```

Diskuse levého spojení: všechny zleva i když nejsou vpravo. Spojení zprava, diskuse. Spojení vnitřní, diskuse.

Aktualizovatelnost dotazu: bez klíče v ODDELENI, s primárním klíčem KOD v ODDELENI.

4.3.9 Více datových zdrojů, spojení (join)

Jen popis, bez příkladu.

4.3.10 Predikáty výběru

Jen popis, bez příkladu. ALL, DISTINCT, DISTINCTROW, TOP n, TOP n PERCENT.

4.3.11 Více datových zdrojů, sjednocení (union)

Jen popis, bez příkladu.

4.4 Příkaz Transform

Popis, účel, pojem "Kontingenční tabulka".

Z13: Počty žen a mužů v jednotlivých odděleních a celkem.

```
transform count(OSCIS) as POCET
select POHLAVI, count(OSCIS) as CELKEM
from ZAMESTNANCI
group by POHLAVI
pivot PRACOVISTE
```

4.5 Poddotazy

4.5.1 Operátor In

Nejprve jen popis, in (10, 12, 18). Pak rozšíření na in (poddotaz). Negace s Not In.

4.5.2 Operátory All, Any

Nejprve jen popis, diskuse < (10, 12, 18). Pak rozšíření na < All | Any (poddotaz). Diskuse výsledku dotazu:

U14: Co zobrazí dotaz

```
select DATUM, KC, CO
from VYDAJE
where KC < All (select KC from VYDAJE where OBCHOD="K")
```

4.5.3 Operátor Exists

Negace s Not Exists.

Z15: Ve kterých odděleních ještě nejsou zaměstnanci?

```
select * from ODDELENI P where not Exists (select * from ZAMESTNANCI Z
where P.KOD=Z.PRACOVISTE)
```

U15: Ve kterých obchodech se nenakupovalo?

```
select *
from OBCHODY K
where not Exists
(select * from VYDAJE V where K.KOD=V.OBCHOD)
```

4.5.4 Vnoření poddotazů

Z16: Kteří zaměstnanci pracují v největším okrese (okresech)?

```
select *
```

```

from ZAMESTNANCI
where PRACOVISTE in (
    select KOD from ODDELENI where OKRES in (
        select SPZ from OKRESY where OBYVATEL=(select max (OBYVATEL)
from OKRESY)
    )
)

```

Výsledek bude prázdný, největší je KI a tam nepracuje nikdo. Proto změnit FM na 300000, tam je jeden zaměstnanec. Vyzkoušet 300000 i pro OP.

U16: Co jsme koupili v obchodě (obchodech), kterých je v republice nejvíc?

```

select *
from VYDAJE
where OBCHOD in (
    select KOD from OBCHODY where OBCHODU=(select max (OBCHODU) from
OBCHODY)
)

```

Výsledek bude prázdný, nejvíc je A a tam se nenakupovalo. Proto změnit OBCHODU u K na 300, tam je nákupů hodně. Vyzkoušet 300 i pro L.

Z17: Kolik je třeba na platy zaměstnanců z nejmenšího okresu?

```

select sum (PLAT)
from ZAMESTNANCI
where PRACOVISTE in (
    select KOD from ODDELENI where OKRES in (
        select SPZ from OKRESY where OBYVATEL=(select min (OBYVATEL)
from OKRESY)
    )
)

```

4.6 Příkazy pro práci se strukturou tabulek

Vytvoření tabulky zaměstnanců s osobním číslem, jménem, datem nástupu a platem:

```

create table ZAMEST (OS_CIS text(3), JMENO text(30), NASTUP date, PLAT
long)

```

Vytvoření primárního klíče zaměstnanců tvořeného osobním číslem:

```

create index KLIC on ZAMEST (OS_CIS) with primary

```

Vytvoření tabulky zaměstnanců přímo s osobním číslem jakožto primárním klíčem:

```

create table ZAMEST (OS_CIS text(3) primary key, JMENO text(30),
NASTUP date, PLAT long)

```

Přidání sloupce s dvouznačkovým kódem oddělení do tabulky zaměstnanců:

```

alter table ZAMEST add column KOD_ODD text(2)

```

Přidání sloupce s pětiznačkovým PSČ jakožto primárním klíčem do tabulky obcí:

```

alter table OBCE add column PSC text(5) primary key

```

Změna sloupce s dvouznačkovým kódem oddělení v tabulce zaměstnanců na čtyřznačkový:


```
alter table ZAMEST alter column KOD_ODD text(4)
```

Vypuštění sloupce s kódem oddělení z tabulky zaměstnanců:

```
alter table ZAMEST drop column KOD_ODD
```

Vypuštění klíče z tabulky zaměstnanců:

```
drop index KLIC on ZAMEST
```

Vypuštění celé tabulky zaměstnanců z databáze:

```
drop table ZAMEST
```

4.7 Příkazy pro práci s daty tabulek

Přidání nového řádku s osobním číslem, jménem, datem nástupu a platem (tj. všech datových polí v pořadí polí stanoveným při vytvoření tabulky) do tabulky zaměstnanců:

```
insert into ZAMEST values ("NOV", "Novák", #1/2/2010#, 9500)
```

Přidání nového řádku s platem a jménem (tj. jen některých datových polí v pořadí polí stanoveným v příkaze) do tabulky zaměstnanců:

```
insert into ZAMEST (PLAT, JMENO) values (9500, "Novák")
```

Přidání nových řádků do tabulky zaměstnanců z jiného datového zdroje (jména VŠECH sloupců ve zdroji nově přijatých MUSÍ být v tabulce zaměstnanců, ale tabulka zaměstnanců může mít více sloupců než nově přijatí - na pořadí nezáleží):

```
insert into ZAMEST select * from NOVE_PRIJATI
```

Vytvoření nové tabulky (třeba jen s některými sloupci) z existujícího datového zdroje:

```
select JMENO, PLAT, NASTUP into ZALOHA_ZAMEST from ZAMEST
```

Změna dat (některých sloupců) tabulky (třeba jen v některých řádcích):

```
update ZAMEST set PLAT=PLAT*1.05, NASTUP=NASTUP+14 where PLAT<9000
```

Vypuštění některých (při "where true" všech) řádků:

```
delete from ZAMEST where PLAT<5000
```

5 Databáze - programování

5.1 Objektový model MDB - ACCDB

5.1.1 Knihovny DAO

Pro 32-bitové systémy: Microsoft DAO 3.6 Object Library (DAO360.DLL). I pro 64-bitové systémy: Microsoft Office 12.0 Access Database Engine Object Library (ACEDAO.DLL), 12 je přitom číslo odpovídající verzi MS Office (např. 15 pro Office 2013, 14 pro Office 2010). Po zpřístupnění mají obě knihovny ve vytvářené aplikaci identifikátor DAO.

5.1.2 Základní objektové třídy

DBEngine, Database. Vytvoření nové prázdné databáze, otevření existující databáze. Základní kolekce objektové třídy Database.

TableDef, Recordset. Logika práce s tabulkami (struktura, data). Field a kolekce Fields. Vlastnosti datového pole, hodnota uložená v datech.

5.2 Zpracování databáze z ne-databázového programu pomocí VBA

5.2.1 Vytvoření a naplnění pomocí SQL

Zpracování programy v modulech Excelu. Cvičné téma: Zaměstnanci firmy. Metoda Execute objektové třídy Database.

5.2.2 Zpracování vytvořené databáze

Otevření instance Recordset, procházení záznamů, získávání hodnot z datového pole a jejich zpracování, zavření Recordset.

Na tomto základě zpracování pomocí příkazu SQL odevzdávajícího Recordset (Select) a přístup k výsledné hodnotě.

5.2.3 Vytvoření a naplnění pomocí instancí tříd z DAO

Vytvoření instance TableDef, naplnění její kolekce Fields přidáváním instancí Field (2 způsoby vytvoření instance Field), přidání TableDef do kolekce TableDefs databáze.

Naplnění daty pomocí instance Recordset, její metody AddNew a Update. Metoda Edit.

5.2.4 Třída Property

Kolekce Properties a její význam pro databázový program obecně a databázový program Access konkrétně. Podprogram pro přidání - změnu instance Property v kolekci Properties.

Použití podprogramu pro nastavení formátu a počtu desetinných míst akceptovaný programem Access při zobrazení dat. Poukaz na způsob zjištění relevantních názvů instancí Property pro program Access.