

Úvod do databázových systémů

7. cvičení

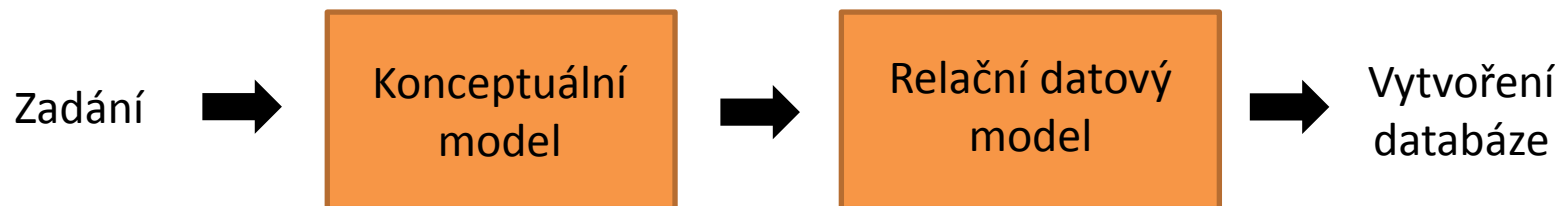
Ing. Petr Lukáš

petr.lukas@nativa.cz

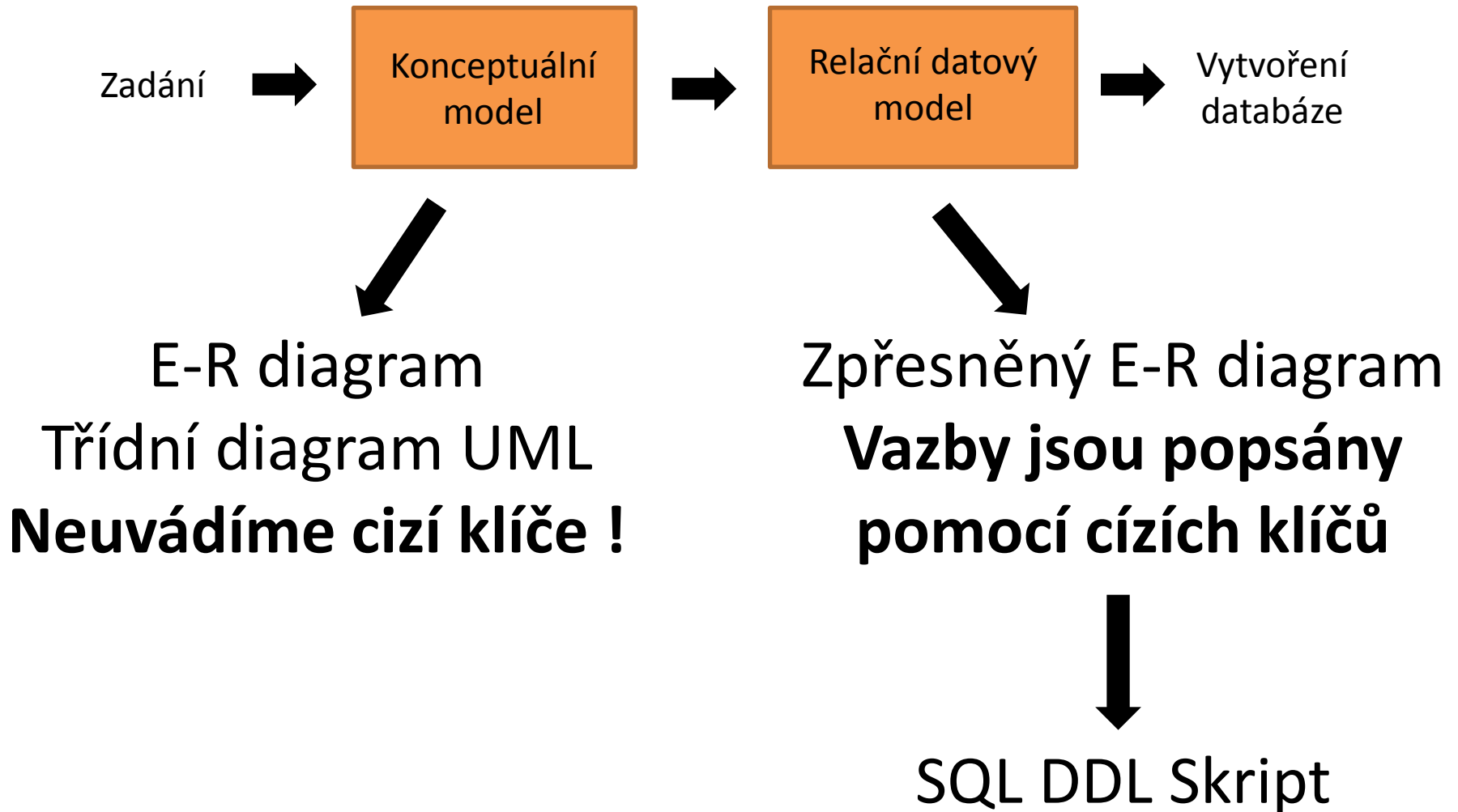
Ostrava, 2012

Modelování databází [1]

- **Datový model** – množina konceptů, které mohou být použity pro popis struktury databáze
 - **Konceptuální model** – logický popis struktury databáze
 - **Databázové schéma** (např. relační datový model) – popis struktury databáze pro konkrétní databázový systém



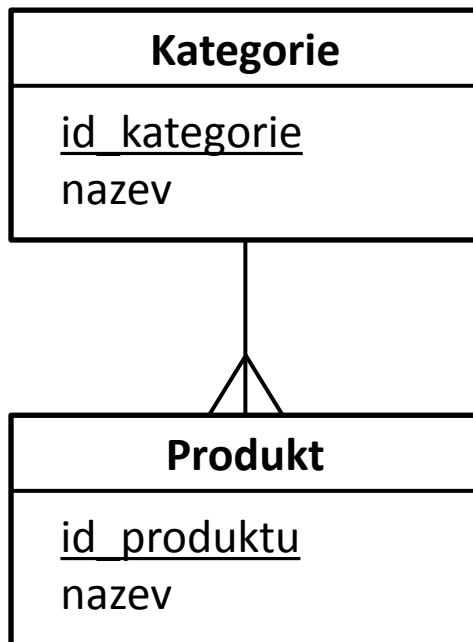
Modelování databází



Řešení kardinality

Vztah 1:N

Konceptuální D. M.

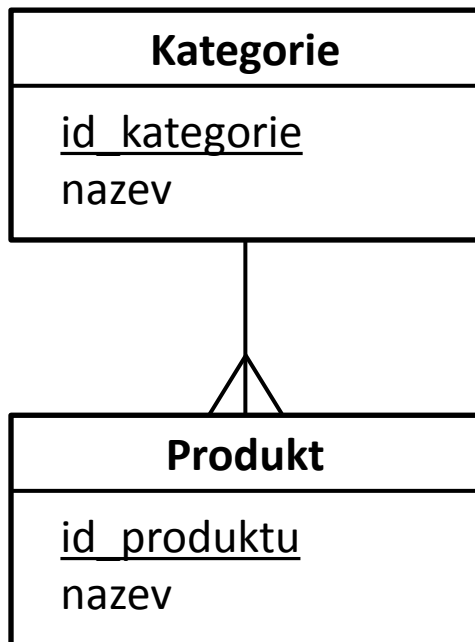


Relační D. M.

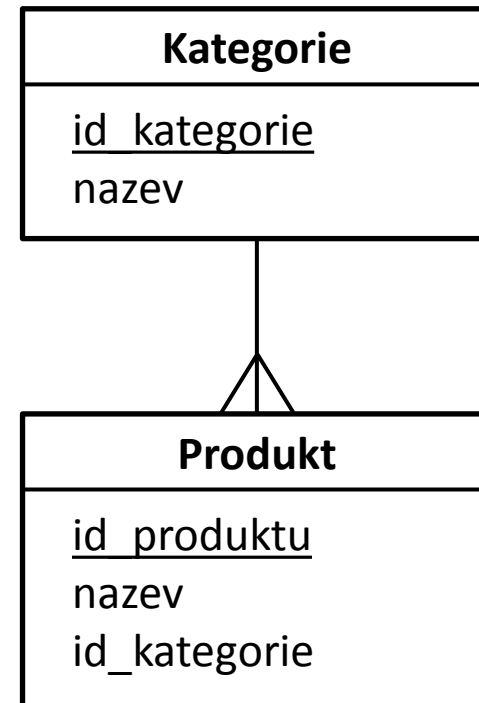
Řešení kardinality

Vztah 1:N

Konceptuální D. M.



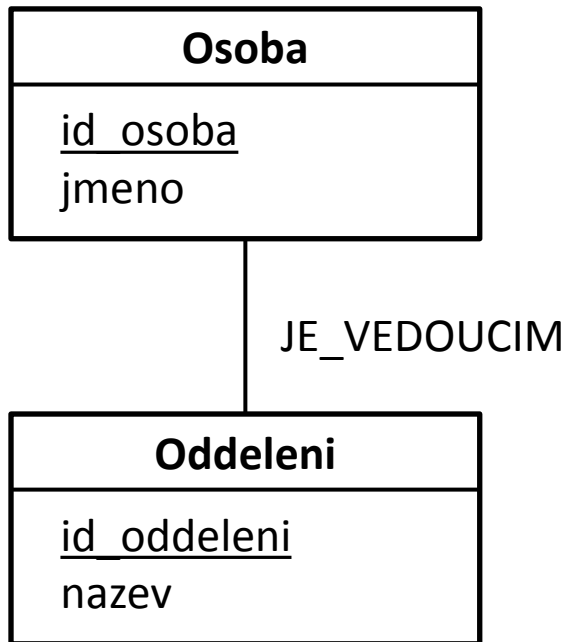
Relační D. M.



Řešení kardinality

Vazba 1:1

Konceptuální D. M.

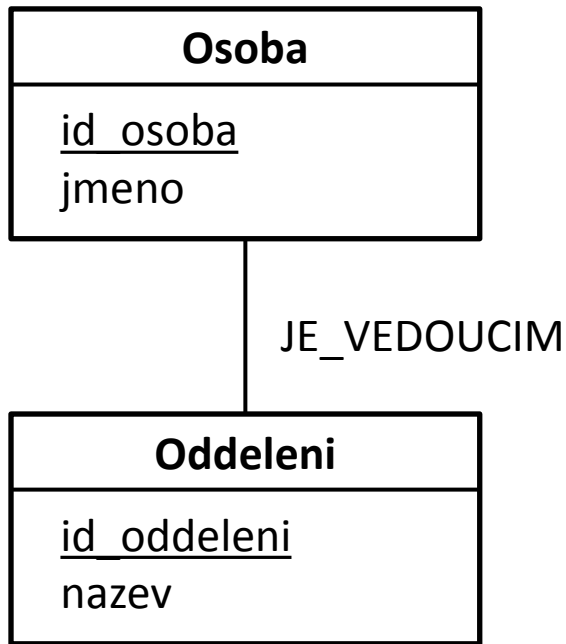


Relační D. M.

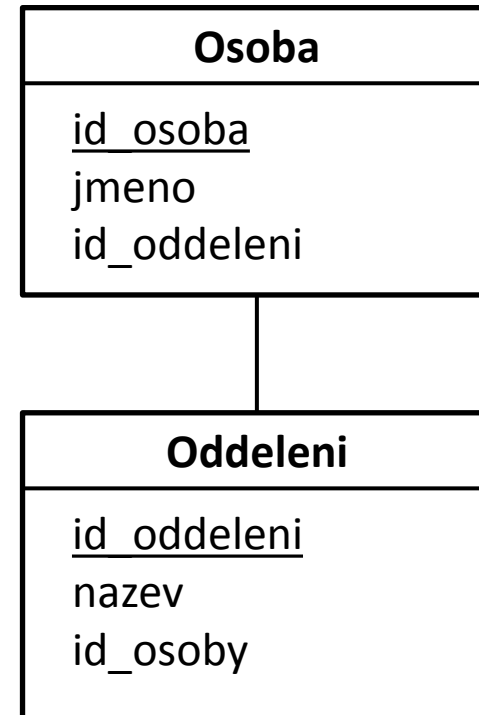
Řešení kardinality

Vazba 1:1

Konceptuální D. M.



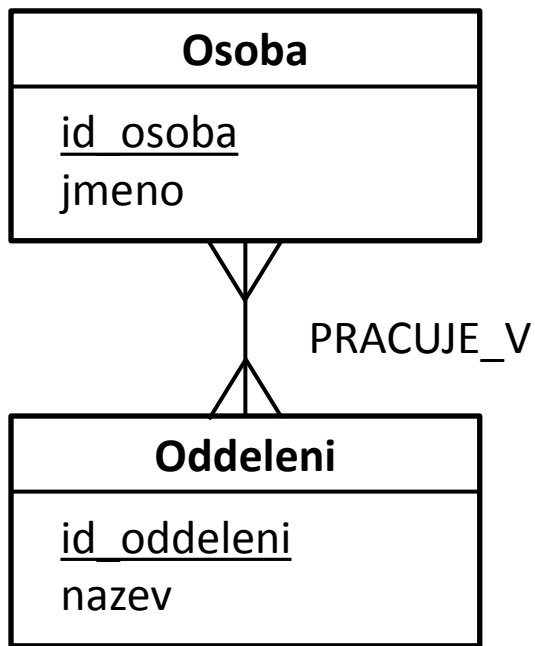
Relační D. M.



Řešení kardinality

Vazba M:N

Konceptuální D. M.

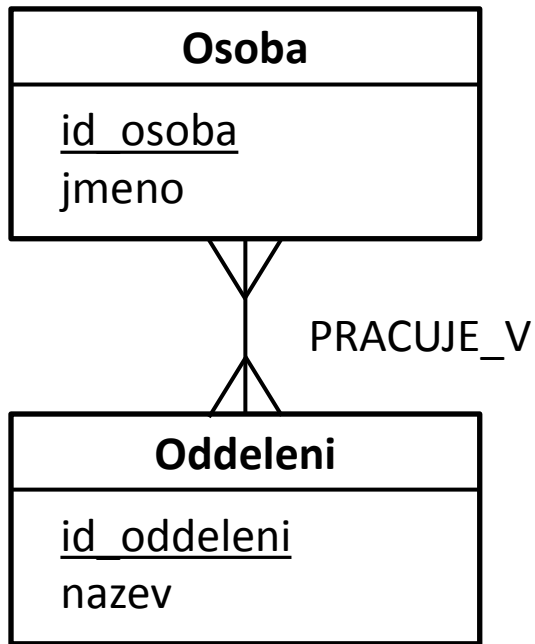


Relační D. M.

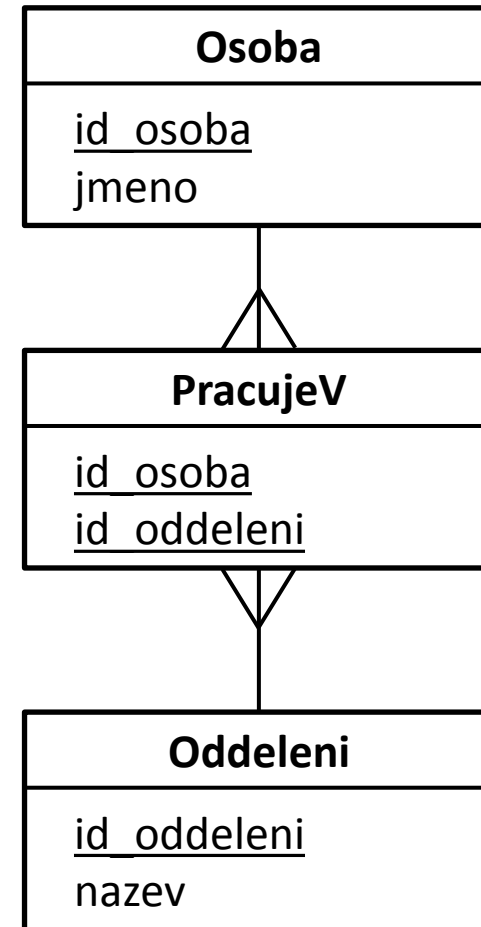
Řešení kardinality

Vazba M:N

Konceptuální D. M.



Relační D. M.



Datový slovník

Datový slovník

Osoba

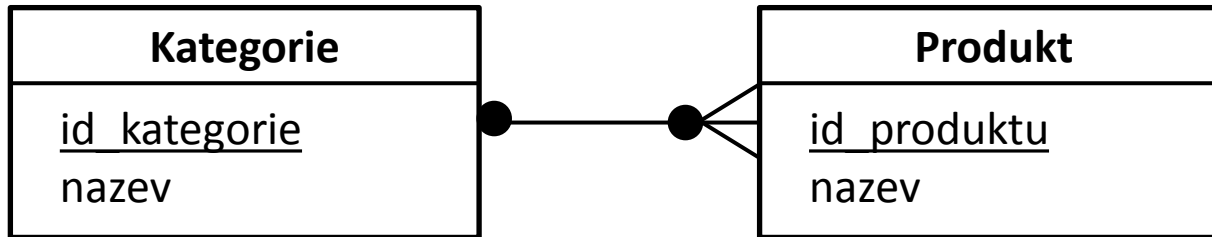
Atribut	Datový typ	Rozsah	Klíč	Null	Další IO
id_osoba	INT	4	ANO	NE	
jmeno	NVARCHAR	50	NE	NE	
prijmeni	NVARCHAR	50	NE	NE	
e_mail	NVARCHAR	40	NE	ANO	Musí obsahovat znak ,@‘

Datové typy

- **INT** Celé číslo
- **FLOAT** Desetinné číslo (plovoucí čárka)
- **DECIMAL(n,p)** Desetinné číslo (celkový počet míst a počet míst za des. oddělovačem)
- **CHAR(n)** Přesný počet znaků
- **VARCHAR(n)** Řetězec s proměnnou délkou
- **BIT** Hodnota 1 nebo 0
- **NCHAR(n)** To samé jako CHAR, ale UNICODE
- **NVARCHAR(n)** To samé jako VARCHAR, ale UNICODE
- **DATETIME** Datum a čas (razítko)

... a další

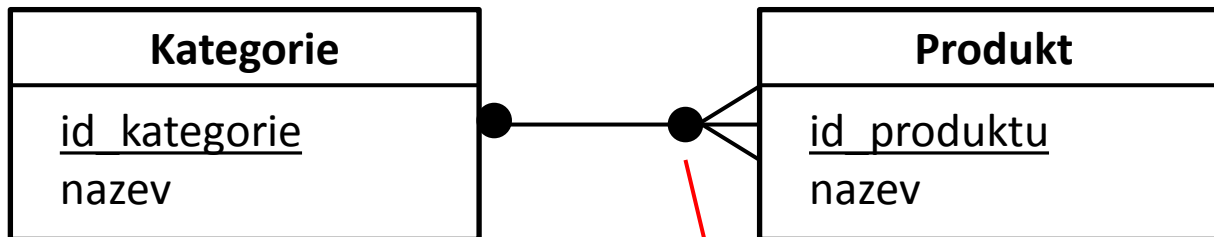
Řešení povinnosti členství



Produkt

Atribut	Datový typ	Rozsah	Klíč	Null	Další IO
id_produktu	INT	4	ANO	NE	
nazev	NVARCHAR	50	NE	NE	
id_kategorie	NVARCHAR	50	NE	NE	

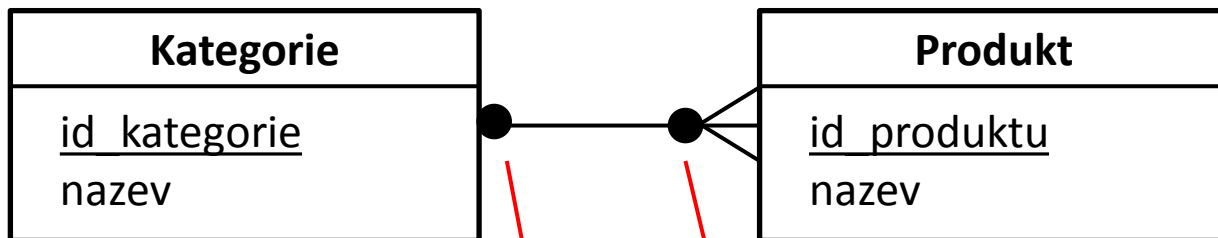
Řešení povinnosti členství



Produkt

Atribut	Datový typ	Rozsah	Klíč	Null	Další IO
id_produktu	INT	4	ANO	NE	
nazev	NVARCHAR	50	NE	NE	
id_kategorie	NVARCHAR	50	NE	NE	

Řešení povinnosti členství



V relačních DB nelze jednoduše vyřešit

Produkt

Atribut	Datový typ	Rozsah	Klíč	Null	Další IO
id_produktu	INT	4	ANO	NE	
nazev	NVARCHAR	50	NE	NE	
id_kategorie	NVARCHAR	50	NE	NE	

Rozdělení jazyků pro práci s daty

- **DDL** (Data Definition Language)
Vytváření datových struktur (prázdných tabulek)
- **DML** (Data Manipulation Language)
Vkládání, úprava a mazání dat
- **DQL** (Data Query Language)
Dotazování nad daty



SQL

Rozdělení jazyků pro práci s daty

DDL / DML / DQL

- Vyhledej nejmladšího zaměstnance
- Vytvoř tabulku s definovanými atributy
- Vrať všechny produkty
- Vlož záznam do tabulky
- Smaž obsah tabulky
- Smaž tabulku
- Přidej sloupec do tabulky

Rozdělení jazyků pro práci s daty

DDL / DML / DQL

- Vyhledej nejmladšího zaměstnance (**DQL**)
- Vytvoř tabulku s definovanými atributy (**DDL**)
- Vrať všechny produkty (**DQL**)
- Vlož záznam do tabulky (**DML**)
- Smaž obsah tabulky (**DML**)
- Smaž tabulku (**DDL**)
- Přidej sloupec do tabulky (**DDL**)

SQL jako DDL – vytvoření tabulky

SQL jako DDL – vytvoření tabulky

```
CREATE TABLE Osoba  
(  
    id_osoby INT PRIMARY KEY NOT NULL,  
    jmeno NVARCHAR(50) NOT NULL,  
    prijmeni NVARCHAR(50) NOT NULL,  
    zamestnanec BIT NOT NULL DEFAULT 0,  
    cislo_oddeleni INT NULL  
    FOREIGN KEY REFERENCES  
    Oddeleni(cislo_oddeleni)  
)
```

SQL jako DDL – vytvoření tabulky

```
CREATE TABLE Osoba
```

```
(
```

```
    id_osoby INT PRIMARY KEY NOT NULL,
```

```
    jmeno NVARCHAR(50) NOT NULL,
```

```
    prijmeni NVARCHAR(50) NOT NULL,
```

```
    zamestnanec BIT NOT NULL DEFAULT 0,
```

```
    cislo_oddeleni INT NULL
```

```
    FOREIGN KEY REFERENCES
```

```
    Oddeleni(cislo_oddeleni)
```

```
)
```

SQL jako DDL – vytvoření tabulky

Datový typ Primární klíč Povinný atribut Výchozí hodnota

```
CREATE TABLE Osoba  
(  
  id_osoby INT PRIMARY KEY NOT NULL,  
  jmeno NVARCHAR(50) NOT NULL,  
  prijmeni NVARCHAR(50) NOT NULL,  
  zamestnanec BIT NOT NULL DEFAULT 0,  
  cislo_oddeleni INT NULL  
  FOREIGN KEY REFERENCES  
  Oddeleni(cislo_oddeleni)  
)
```

Cizí klíč

SQL jako DDL – další užitečné příkazy

Odstranění tabulky

Přidání sloupce

Úprava sloupce (např. rozsahu)

SQL jako DDL – další užitečné příkazy

Odstranění tabulky

```
DROP TABLE Osoba
```

Přidání sloupce

```
ALTER TABLE Osoba
```

```
ADD datum_narozeni DATETIME
```

Úprava sloupce (např. rozsahu)

```
ALTER TABLE Osoba
```

```
ALTER COLUMN jmeno NVARCHAR(100)
```


SQL jako DDL

- CREATE ...** Vytvářím něco nového (tabulku, sloupec, proceduru, funkci)
- ALTER...** Upravuji něco, co existuje
- DROP...** Odstraňuji něco, co existuje

DDL příkazy za nás ve většině případů řeší vhodný manager (např. SQL Management Studio).

SQL jako DML – vkládání dat

Příkaz pro jednotlivé vkládání záznamů.

- Pozor, počet atributů uvedený za názvem tabulky musí odpovídat počtu uvedených hodnot.

```
INSERT INTO Osoba (id_osoby, jmeno, prijmeni)  
VALUES (1, 'Petr', 'Lukáš')
```

SQL jako DML – úprava dat

Uvádíme název tabulky, jeden nebo více atributů, jejichž hodnotu chceme změnit (oddělujeme čárkami) a podmínku, pro které záznamy se má update provést.

```
UPDATE Osoba  
SET jmeno = 'Jakub', prijmeni = 'Pokusný'  
WHERE osoba_id = 1
```

SQL jako DML – úprava dat

Uvádíme název tabulky, jeden nebo více atributů, jejichž hodnotu chceme změnit (oddělujeme čárkami) a podmínku, pro které záznamy se má update provést.

```
UPDATE Osoba
SET jmeno = 'Jakub', prijmeni = 'Pokusný'
WHERE osoba_id = 1
```

 **Pozor, nezapomenout na WHERE**

SQL jako DML – mazání dat

Uvádíme název tabulky, nesmíme zapomenout na omezující podmínku.

```
DELETE FROM Osoba  
WHERE osoba_id = 1
```

 **Pozor, nezapomenout na WHERE**

Úvod do databázových systémů

www.dbedu.cs.vsb.cz

jméno: **student**

heslo: **tuodbedu**

Vpravo sloupec -> České kurzy -> UDBS -> Cvičení – Relaçní algebra

Připojit se k databázi

Reference

[1] Radim Bača, *Konceptuální model – Úvod do databázových systémů*, 2012