

Relační algebra

Jiří Zacpal



KATEDRA INFORMATIKY
UNIVERZITA PALACKÉHO V OLOMOUCI

KMI/DATAB Databáze

Příklad

```
CREATE TABLE osoby1  
(  
    oscislo text PRIMARY KEY,  
    jmeno text,  
    prijmeni text,  
    obor text  
);
```

```
CREATE TABLE osoby2  
(  
    oscislo text PRIMARY KEY,  
    jmeno text,  
    prijmeni text,  
    obor text  
);
```

Řádky



■ osoby1 :

osCislo	jmeno	prijmeni	obor
R20379	Vojtěch	ASSZONYI	AM-DS
R200438	Martin	BRABLÍK	IT
R20647	Jakub	BŘEZINA	IT
R20648	Ondřej	ČECH	IT
R20649	Jiří	DIVIŠ	IT

■ osoby2:

osCislo	jmeno	prijmeni	obor
R20379	Vojtěch	ASSZONYI	AM-DS
R200438	Martin	BRABLÍK	IT
R20652	Jakub	HAMPL	IT
R20653	Oldřich	HORÁK	IT
R20655	Samuel	JAŠÍK	IT

TABLE v1 UNION TABLE v2

- Příklad:

(TABLE osoby1) UNION (TABLE osoby2);

(TABLE osoby2) UNION (TABLE osoby1);

TABLE v1 INTERSECT TABLE v2

- Příklad:

```
(TABLE osoby1) INTERSECT (TABLE osoby2);
```

```
(TABLE osoby2) INTERSECT (TABLE osoby1);
```

TABLE v1 EXCEPT TABLE v2

- Příklad:

(TABLE osoby1) EXCEPT (TABLE osoby2);

(TABLE osoby2) EXCEPT (TABLE osoby1);

Úkol



- Proveďte:
 - sjednocení obou tabulek,
 - průnik obou tabulek,
 - rozdíl první a druhé tabulky.

Složitější výrazy

- Výstup jedné operace můžeme použít na vstup jiné
- Příklad:

```
TABLE osoby1 EXCEPT TABLE osoby2
```

```
UNION
```

```
TABLE osoby2 EXCEPT TABLE osoby1;
```


Výrazy ve FROM klauzuli

Výrazem

`expr AS name`

ve FROM klauzuli určíme tabulku, která bude hodnotou tabulkového výrazu `expr` a bude se jmenovat `name` .

```
SELECT DISTINCT jmeno
```

```
FROM ((TABLE osoby1) union (TABLE osoby2)) as osoby;
```

Příklad

- Máme tabulky:

```
CREATE TABLE studenti (  
    id_student INT,  
    jmeno TEXT,  
    prijmeni TEXT,  
    rocnik INT,  
    email TEXT  
);  
  
CREATE TABLE vyucujici (  
    id_vyucujici INT,  
    jmeno TEXT,  
    prijmeni TEXT,  
    titul TEXT,  
    kancelar TEXT,  
    email TEXT  
);
```

Příklad



- Lze s nimi provádět operace relační algebry?
- Například bychom chtěli zjistit, kteří studenti jsou současně i vyučující:
`SELECT * FROM studenti INTERSECT SELECT * FROM vyucujici;`
each INTERSECT query must have the same number of columns

- Obě tabulky musíme upravit tak, aby měly stejný počet a typ sloupců:
`SELECT jmeno,prijmeni FROM studenti INTERSECT SELECT
jmeno,prijmeni FROM vyucujici;`

Příklad



- Co ale když chceme zobrazit i informaci, jaký kabinet má dotyčná osoba:

```
SELECT jmeno,prijmeni,NULL AS kancelar FROM studenti  
INTERSECT SELECT jmeno,prijmeni,kancelar FROM vyucujici;
```


- Toto není správně. Proč?
- Dotaz musíme upravit:

```
SELECT jmeno,prijmeni,(SELECT kancelar FROM vyucujici AS  
v,studenti AS s WHERE v.jmeno=s.jmeno AND  
v.prijmeni=s.prijmeni AND s.jmeno=studenti.jmeno AND  
s.prijmeni=studenti.prijmeni) as kancelar FROM studenti  
INTERSECT SELECT jmeno,prijmeni,kancelar FROM vyucujici;
```

Bodovaný úkol

- Vytvořte tabulky knihy1 a knihy2 dle uvedeného schématu.



Knihy	
 PK	ISBN
nazev	
vydavatel	
pocet_stran	
rok_vydani	
cena	
zanr	
typ	

Bodovaný úkol



2. Do tabulky knihy1 vložte tyto záznamy:

ISBN	nazev	vydavatel	rok_vydani	pocet_stran	cena	zanr	typ
K003	Stopařův průvodce po galaxii	MF	1985	214	120	scifi	kniha
K045	Pán prstenů - Dvě věže	MF	1948	251	240	fantasy	kniha
K025	Barva kouzel	Talpress	1989	221	358	fantasy	ebook
K026	Strážé! Strážé!	Talpress	2000		214	fantasy	ebook
K027	Lehké fantastično	Talpress	1999	145	415	fantasy	ebook

Do tabulky knihy2 vložte tyto záznamy:

ISBN	nazev	vydavatel	rok_vydani	pocet_stran	cena	zanr	typ
K025	Barva kouzel	Talpress	1989	221	358	fantasy	ebook
K026	Strážé! Strážé!	Talpress	2000		214	fantasy	ebook
K027	Lehké fantastično	Talpress	1999	145	415	fantasy	ebook

Bodovaný úkol

3. Proveďte tyto operace:

- sjednocení obou tabulek,
- průnik obou tabulek,
- tento výraz: $\text{knihy2} - \text{knihy1} \cap \text{knihy1}$