

Jazyk SQL – dotazy nad více tabulkami

Jiří Zácpal



KATEDRA INFORMATIKY
UNIVERZITA PALACKÉHO V OLOMOUCI

KMI/DATAB Databáze

Kartézský součin

Tabulky



Proveďte příkazy ze souboru `datab_03_dotazy_nad_vice_tabulkami_vstup.sql`
vyucujici

id_vyucujici integer	jmeno text	prijmeni text	titul text	kancelar text	email text
1	Jiří	Zacpal	Mgr., Ph.D.	5.071	jiri.zacpal@upol.cz
2	Arnošt	Večerka	RNDr.	5.071	arnost.vecerka@upol.cz

predmety

id_predmet integer	zkratka text	nazev text	semestr text	vyucujici integer	mistnost integer	kredity integer
1	KMI/DATA	Databáze	ZS	1	1	5
2	KMI/WEBA	Webové aplikace	LS	1	2	3
3	KMI/SOFT	Softwarové inženýrství	ZS	2	1	6
4	KMI/ZPP1	Základy programování pro	LS	1	3	4
5	KMI/ALGO1	Algoritmy 1	ZS	2	1	4

Kartézský součin tabulek

- základní syntaxe:

```
SELECT sloupce  
FROM tabulka_1, tabulka_2  
WHERE where_fráze;
```

Příklad

- Chtěli bychom zobrazit jméno vyučujícího pro každý předmět:

```
SELECT vyucujici.jmeno, vyucujici.prijmeni,  
predmety.zkratka, predmety.nazev  
FROM vyucujici,predmety;
```

- Je výsledek správně?
- Potřebujeme ještě vybrat správné řádky.
- Které řádky jsou správně?

Příklad



- Doplníme klauzuli WHERE:

```
SELECT vyucujici.jmeno, vyucujici.prijmeni, predmety.zkratka,  
predmety.nazev  
FROM vyucujici,predmety  
WHERE vyucujici.id_vyucujici=predmety.vyucujici;
```

- V případě, že názvy sloupců jsou unikátní, není nutné určovat, ze které tabulky pochází.

```
SELECT jmeno, prijmeni, zkratka, nazev  
FROM vyucujici,predmety  
WHERE id_vyucujici=vyucujici;
```

Příklad

- Zobrazíme předměty, které vyučuje Jiří Zacpal:

```
SELECT jmeno, prijmeni, zkratka, nazev  
FROM vyucujici,predmety  
WHERE id_vyucujici=vyucujici AND prijmeni='Zacpal' AND  
jmeno='Jiří';
```

- Zobrazíme předměty, které vyučuje Jiří Zacpal nebo za ně lze získat více než 4 kredity:

```
SELECT jmeno, prijmeni, zkratka, nazev  
FROM vyucujici,predmety  
WHERE id_vyucujici=vyucujici AND prijmeni='Zacpal' AND  
jmeno='Jiří';
```

Příklad

- Zobrazíme předměty, které vyučuje Jiří Zacpal nebo za ně lze získat více než 4 kredity:

```
SELECT jmeno, prijmeni, zkratka, nazev  
FROM vyucujici,predmety  
WHERE id_vyucujici=vyucujici AND prijmeni='Zacpal' AND  
jmeno='Jiří' OR kredity>4;
```

- Je to správně?
- Je potřeba dobře napsat podmínku v klauzuli WHERE:

```
SELECT jmeno, prijmeni, zkratka, nazev  
FROM vyucujici,predmety  
WHERE id_vyucujici=vyucujici AND ((prijmeni='Zacpal' AND  
jmeno='Jiří') OR (kredity>4));
```


Příkaz SELECT pro spojení více tabulek

- slouží k výběru záznamů
- základní syntaxe:

SELECT sloupce

FROM tabulka_1, tabulka_2, tabulka_3, ...

WHERE where_fráze;

Příklad

- Zobrazíme seznam předmětů, které mají zapsáni jednotliví studenti:

```
SELECT jmeno, prijmeni, zkratka, nazev  
FROM studenti, predmety, zapsani  
WHERE id_student=student AND id_predmet = předmět;
```

- Zobrazíme všechny studenty, které vyučuje Arnošt Večerka:

```
SELECT jmeno, prijmeni, zkratka, nazev  
FROM vyucujici, predmety  
WHERE id_vyucujici=vyucujici AND ((prijmeni='Zacpal' AND  
jmeno='Jiří') OR (kredity>4));
```

Příklad

- Zobrazíme všechny studenty, které vyučuje Arnošt Večerka:

```
SELECT jmeno, prijmeni  
FROM studenti, predmety, zapsani, vyucujici  
WHERE id_student=student AND id_predmet = predmet AND  
id_vyucujici=vyucujici AND jmeno='Arnošt' AND  
prijmeni='Večerka';
```

- Dotaz je chybný, protože sloupce jmeno a prijmeni jsou ve dvou tabulkách:

```
SELECT studenti.jmeno, studenti.prijmeni  
FROM studenti, predmety, zapsani, vyucujici  
WHERE id_student=student AND id_predmet = predmet AND  
id_vyucujici=vyucujici AND vyucujici.jmeno='Arnošt' AND  
vyucujici.prijmeni='Večerka';
```

Přirozené spojení

Přirozené spojení

- Pro spojení tabulek můžeme použít přirozené spojení:

SELECT sloupce

FROM tabulka_1 NATURAL JOIN tabulka_2

[WHERE where_fráze];

- spojení se provede přes stejné sloupce v tabulce_1 a tabulce_2,
- podmínkou tedy je, že obě dvě tabulky mají stejné sloupce.

Příklad

- Chtěli bychom zobrazit jméno vyučujícího pro každý předmět:

```
SELECT jmeno, prijmeni, zkratka, nazev  
FROM vyucujici NATURAL JOIN predmety;
```

- Je výsledek správně?
- Není, tabulky nemají shodné sloupce.
- Můžeme však použít alias:

```
SELECT jmeno, prijmeni, zkratka, nazev  
FROM vyucujici NATURAL JOIN (SELECT vyucujici AS id_vyucujici,  
zkratka, nazev FROM predmety) AS nove_predmety;
```

Příklad



- Zobrazíme seznam předmětů, které mají zapsáni jednotliví studenti:

```
SELECT jmeno, prijmeni, zkratka, nazev  
FROM studenti NATURAL JOIN (SELECT predmet AS id_predmet,  
student AS id_student FROM zapsani) AS nove_zapsani  
NATURAL JOIN predmety;
```