

Databázové systémy a SQL

Lekce 3

Daniel Klimeš

Spojení sloupců = JOIN

ID pacienta	Jmeno	Prijmeni
1	Jan	Novák
2	Jana	Nová
3	Karel	Starý

+

ID pacienta	Datum vysetreni	Vysledek vysetreni
1	12.1.2011	39,5
1	15.3.2011	36,8
2	2.2.2011	37,5

+

ID lekare	Jmeno	Prijmeni
10	Petr	Šikovný
20	Jana	Levá
30	Karel	Starý

Spojení řádků – množinové operace

Operace s dotazy, které vrací stejnou datovou strukturu (stejně sloupce)

- **UNION** **Sjednocení množin – duplicitní řádky vyloučeny**
- **UNION ALL** **Sjednocení množin včetně duplicit**
- **INTERSECT** **Průnik množin – pouze shodné řádky**
- **MINUS** **Rozdíl množin**

SELECT sloupec FROM tabulka

UNION

SELECT sloupec FROM tabulka2

Počet sloupců prvního a druhého dotazu musí být stejný
a musí být stejného datového typu

ID pacienta	Jmeno	Prijmeni
1	Jan	Novák
2	Jana	Nová
3	Karel	Starý

ID pacienta	Datum vysetreni	Vysledek vysetreni
1	12.1.2011	39,5
1	15.3.2011	36,8
2	2.2.2011	37,5

Spojování tabulek = join

Druhy spojení:

- vnitřní - **inner join** - jen spojitelné řádky
- vnější - **outer join** - **left join**, **right join**, **full join**
všechny řádky jedné tabulky + napojitelné řádky druhé tabulky

Vnitřní spojení

```

SELECT * FROM tabulka1, tabulka2 WHERE tabulka1.sloupec (PK) = tabulka2.sloupec (FK)
SELECT * FROM pacient, vysetreni WHERE pacient.id_pacienta = vysetreni.id_pacienta
    
```

ID pacienta	Jmeno	Prijmeni	ID_pacienta	Datum vysetreni	Vysledek vysetreni
1	Jan	Novák	1	12.1.2011	39,5
1	Jan	Novák	1	15.3.2011	36,8
2	Jana	Nová	2	2.2.2011	37,5

Vnější spojení

SELECT * FROM tabulka1 **LEFT JOIN** tabulka2 ON tabulka1.sloupec = tabulka2.sloupec
 SELECT * FROM pacient **LEFT JOIN** vysetreni ON pacient.id_pacienta = vysetreni.id_pacienta

ID pacienta	Jmeno	Prijmeni	ID_pacienta	Datum vysetreni	Vysledek vysetreni
1	Jan	Novák	1	12.1.2011	39,5
1	Jan	Novák	1	15.3.2011	36,8
2	Jana	Nová	2	2.2.2011	37,5
3	Karel	Starý			

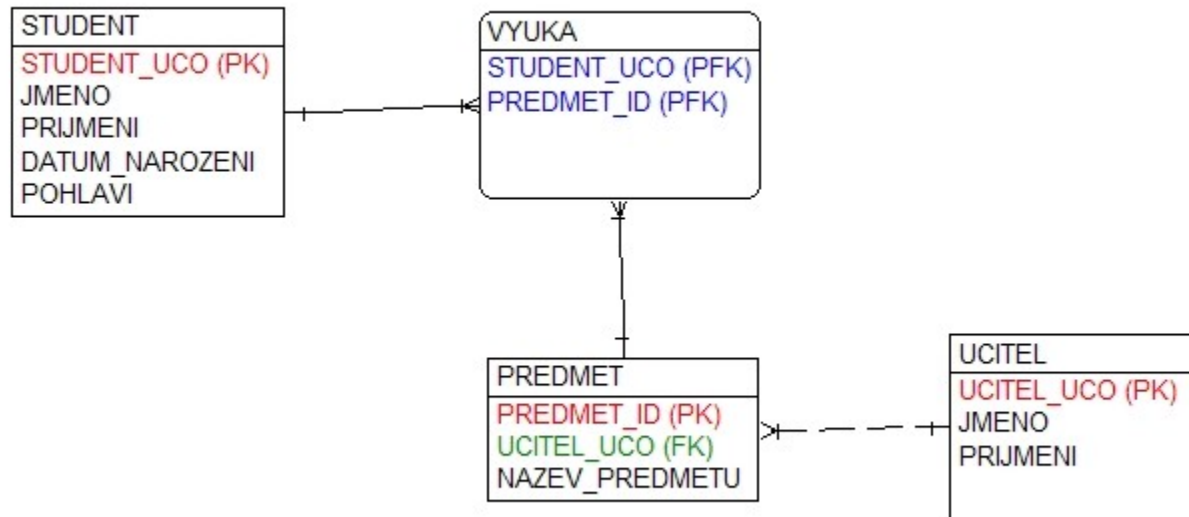
ORACLE varianta

SELECT * FROM tabulka1, tabulka2 WHERE tabulka1.sloupec = tabulka2.sloupec(+)

SELECT * FROM pacient, vysetreni WHERE pacient.id_pacienta = vysetreni.id_pacienta(+)

Vazba student - predmet m-n => „mezitabulka“ VYUKA

[1.1]

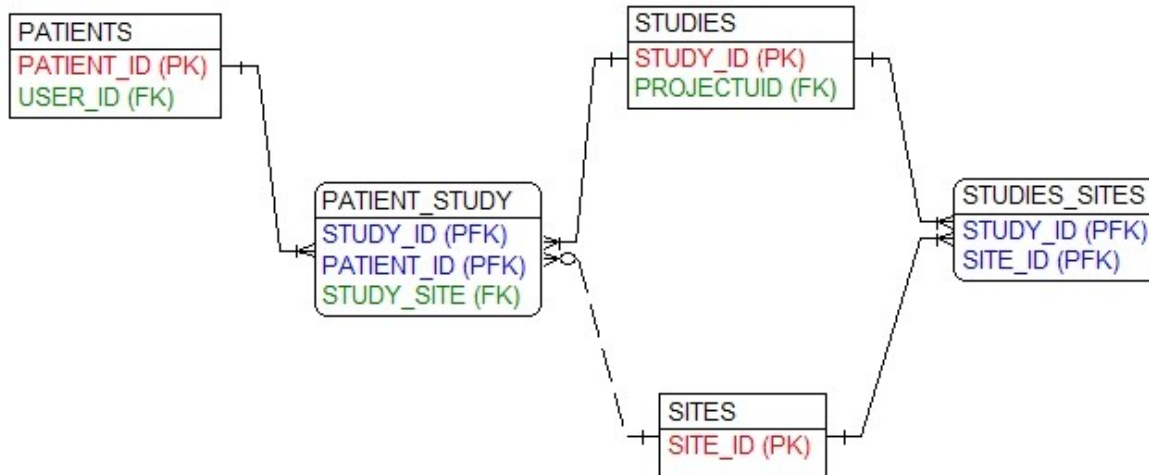


[2.1]

- 1) Zapište své jméno do tabulky STUDENT
- 2) Zapište si vybraný předmět/předměty do tabulky VYUKA
- 3) Vypište studenty zapsané do alespoň jednoho předmětu
- 4) Vypište všechny studenty s vybraným předmětem/předměty
- 5) Vypište všechny předměty a k nim počet zapsaných studentů
- 6) Vypište učící učitele a jeho studenty
- 7) Odhlašte **sebe** ze všech předmětů
- 8) Přihlaste se jedním příkazem do všech předmětů

Vazba pacienti – studie m-n => „mezitabulka“ PATIENT_STUDY
 Vazba studie – pracoviště m-n => „mezitabulka“ STUDIES_SITES

[1,1]



Zjistěte počet pacientů v jednotlivých studiích
STUDY_NAME, počet pacientů

Zjistěte počet pacientů dle pohlaví v jednotlivých studiích
STUDY_NAME, pohlaví, počet pacientů

Zjistěte počet zapojených pracovišť do jednotlivých studií
STUDY_NAME, počet pracovišť

Vypište pracoviště zapojená do více studií
SITE, počet studií

Vypište všechny studie a počet zařazených pacientů v jednotlivých letech
STUDY_NAME, rok(DATE_OF_ENROLLMENT)

Zjistěte počet pacientů v jednotlivých studiích po pracovištích
STUDY_NAME, SITE, počet pacientů

Vypište všechny pacienty ze studií study_id 3 a 23

Vypište všechny unikátní pacienty ze studií study_id 3 a 23

Vypište společné pacienty ze studií study_id 3 a 23

Kolika studií se účastní centra z Brna?

Kolika studií se účastní centra z Prahy (všechny obvody)?

Seznam studií, které se účastní Brno, ale nikdo z Prahy

Zanořené dotazy

- uzavřené v kulatých závorkách ()
- vložení:

místo názvu sloupce

místo názvu tabulky

v sekci WHERE

Místo sloupce:

```
SELECT COUNT(patient_id), (select count (*) FROM patients)
FROM patient_study
WHERE study_id = 3 ;
```

- vnořený dotaz na pozici sloupce musí vrátit právě jeden řádek

Subdotaz na pozici FROM nahrazuje tabulku

```
SELECT COUNT(*) FROM (
  SELECT study_id, COUNT(*)
  FROM patient_st udy GROUP BY study_id
)
```

Varianty:

- WHERE sloupec = (SELECT sloupec FROM...
- WHERE sloupec = ANY (SELECT sloupec FROM...
- WHERE sloupec IN (SELECT sloupec FROM ...
- WHERE sloupec > ALL (SELECT sloupec FROM ...
- WHERE EXISTS (SELECT * FROM....
- WHERE NOT EXISTS (SELECT * FROM...

Nejmladší pacient:

```
SELECT * FROM patients WHERE date_of_birth = (
    SELECT MAX(date_of_birth) FROM patients);
```

```
SELECT * FROM patients WHERE date_of_birth IS NOT NULL
    AND date_of_birth >= ALL (
    SELECT date_of_birth FROM patients);
```

```
SELECT * FROM patients tab1 WHERE date_of_birth IS NOT NULL
    AND NOT EXISTS (
    SELECT * FROM patients tab2 WHERE tab2. date_of_birth > tab1. date_of_birth );
```

Napište 3 varianty, které zobrazí řádek s nejstarším pacientem

Napište dotaz, který vrátí všechny pacienty kromě nejstaršího a nejmladšího

Vypište všechny studie a počet zařazených pacientů v jednotlivých letech a u každé nejmladšího a nejstaršího pacienta v daném roce

STUDY_NAME, rok(DATE_OF_ENROLLMENT),
min(date_of_birth), max(date_of_birth)

Zjistěte počet pacientů ve studiích, kde počet pacientek není větší než 10

STUDY_NAME, počet pacientů

Registr/studie se skládá z 1 až n formulářů, které se vyplňují v určité fázi péče o pacienta

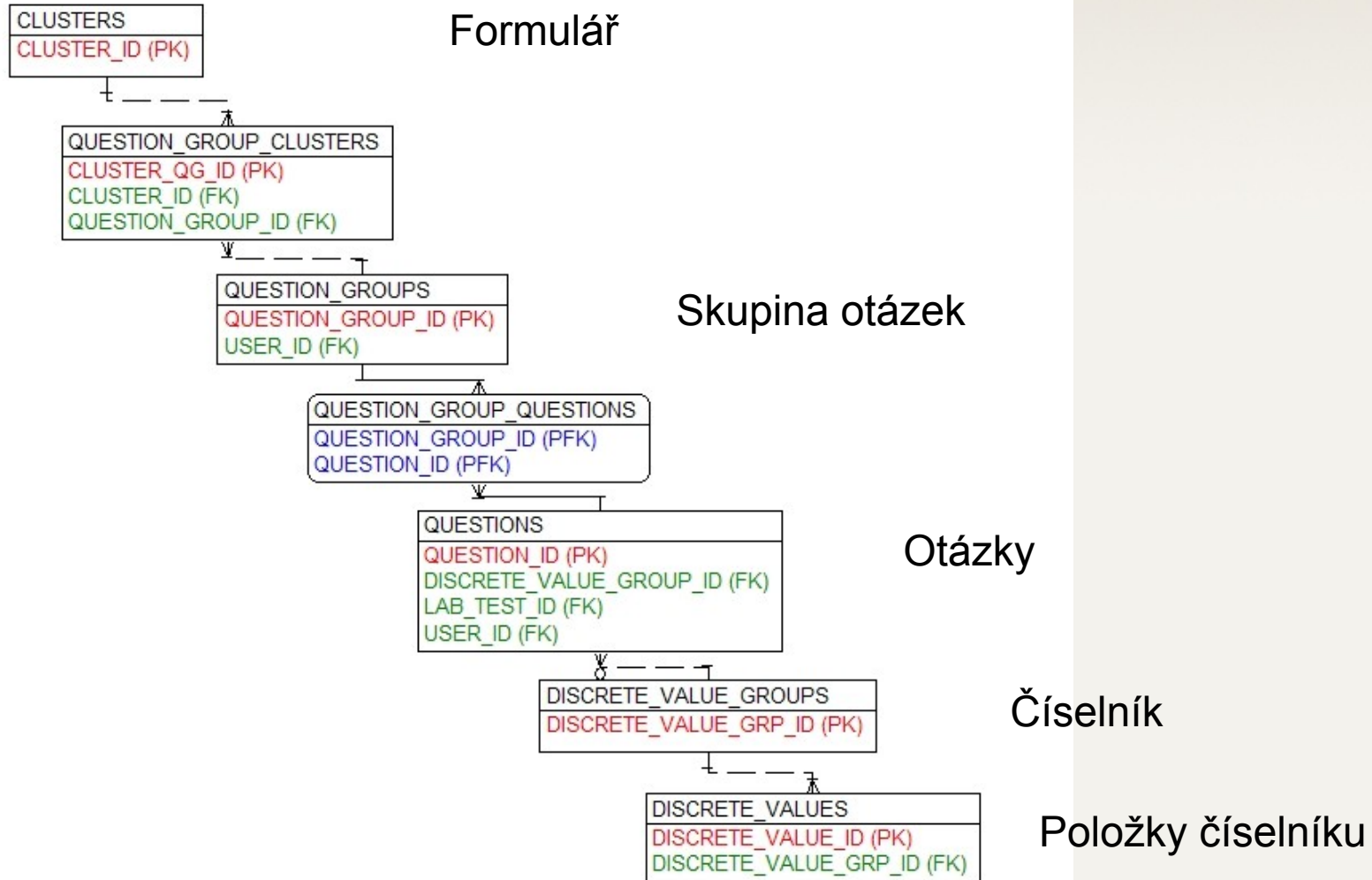
Formulář se skládá z 1 až n skupin otázek

Skupina otázek je tvořena 1 až n otázkami

Otázky mohou být různého datové typu, (číslo, text, datum, číselník)

Číselník je sada povolených odpovědí na danou otázku (výběr z nabídky „roletka“, combo box)

[1,1]



- Najděte formulář bez definované skupiny otázek
- Najděte formulář s největším počtem skupin
- Najděte formulář s největším počtem otázek
- Najděte nejčastěji používaný číselník