

# Databázové systémy a SQL

## Lekce 4

Daniel Klimeš



---

# PRÁCE S VÍCE TABULKAMI

PatientID	Firstname	Lastname
1	Jan	Novák
2	Jana	Nová
3	Karel	Starý

+

PatientID	ExamDate	ExamResult
1	12.1.2011	39,5
1	15.3.2011	36,8
2	2.2.2011	37,5

ENTITY = tabulky/tables

RELATIONSHIP = vazba

E-R diagramy = datové modely (data models)

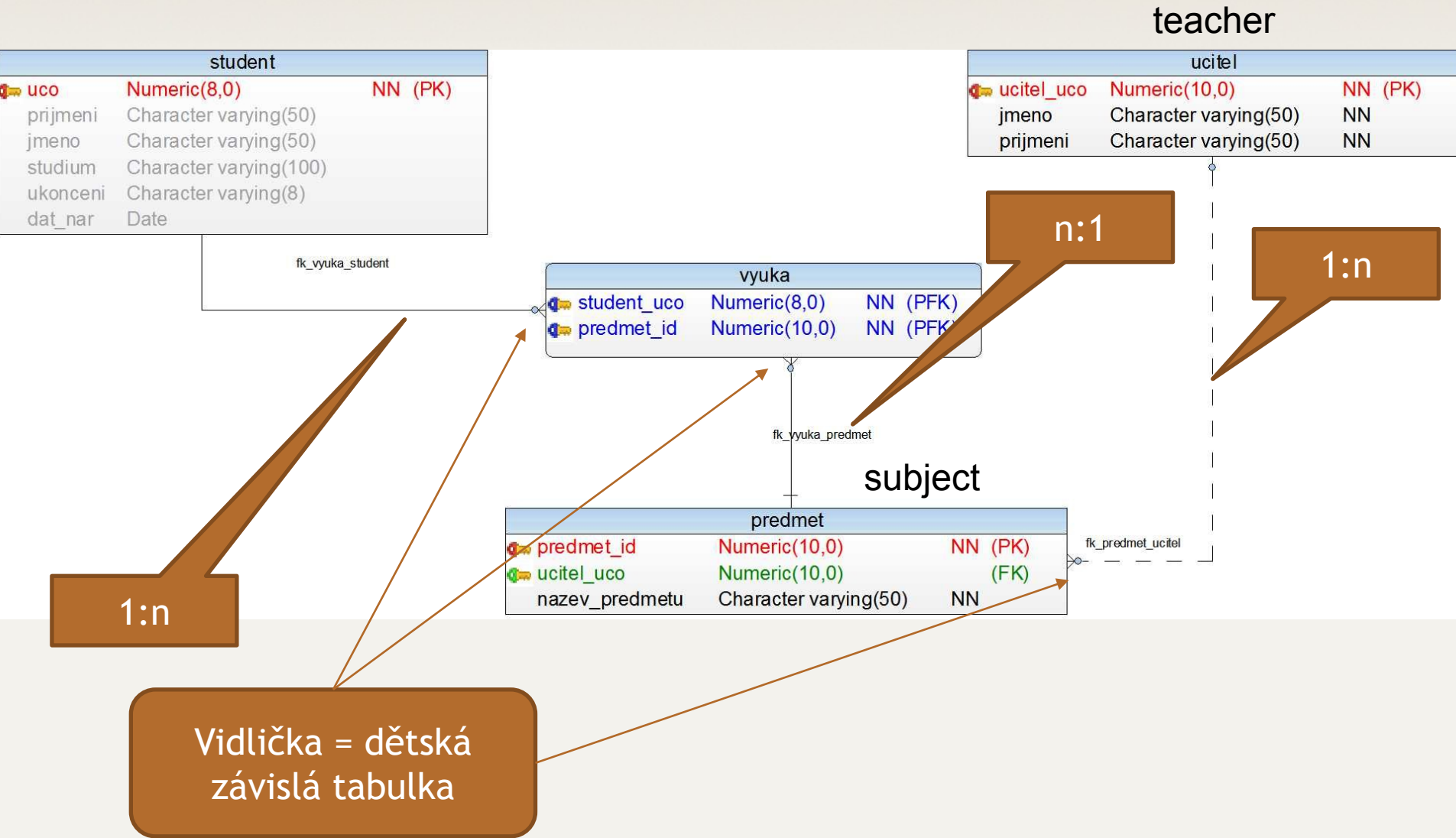
Typy vazeb:

1:1 – jeden řádek tabulky A má vazbu s jedním řádkem tabulky B

**1:n – k jednomu řádku tabulky A se váže 0 až N řádků tabulky B**

m:n – k jednomu řádku tabulky A se váže 0 až N řádků tabulky B  
**ale zároveň** k jednomu řádku z B se váže 0 až N řádků A

# ER diagram



- Definice entit (tabulek)
- Stanovení primárních klíčů všech tabulek
- Tvorba vazeb
  - **Migrace primárního klíče rodičovské tabulky do dětské tabulky**
  - **Cizí klíč může, ale nemusí být součástí primárního klíče dětské tabulky**

Spustte skript2.sql

- Vytvořte si vlastní předmět (řádek v tabulce předmět)
- Zkuste vytvořit předmět s neexistujícím UCO\_teacher
- Přihlaste se do zvolených předmětů
- Odhlašte se ze všech předmětů
- Přihlaste se do **všech** dostupných předmětů
- Zkuste smazat všechny učitele



---

# DOTAZOVÁNÍ VÍCE TABULEK



Spojení sloupců = JOIN

PatientID	Firstname	Lastname
1	Jan	Novák
2	Jana	Nová
3	Karel	Starý

+

PatientID	Date_of_exam	Result
1	12.1.2011	39,5
1	15.3.2011	36,8
2	2.2.2011	37,5

+

PhysicianID	Firstname	Lastname
10	Petr	Šikovný
20	Jana	Levá
30	Karel	Starý

Spojení řádků – množinové operace

PatientID	Firstname	Lastname
1	Jan	Novák
2	Jana	Nová
3	Karel	Starý

PatientID	Date_of_exam	Result
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 / Inner join

```
SELECT * FROM pacient JOIN vysetreni ON pacient.id_pacienta = vysetreni.id_pacienta
```

PatientID	Firstname	Lastname	PatientID	Date_of_exam	Result
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

### Alternativní varianta

```
SELECT * FROM tabulka1, tabulka2 WHERE tabulka1.sloupec = tabulka2.sloupec
```

```
SELECT * FROM pacient, vysetreni WHERE pacient.id_pacienta = vysetreni.id_pacienta
```

## 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

PatientID	Firstname	Lastname	PatientID	Date_of_exam	Result
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(+)

Vypište studenty zapsané do alespoň jednoho předmětu

*Select students with one or more registered subjects*

Vypište všechny studenty s vybraným předmětem/předměty

*Select all students with a given registered subject*

Vypište své jméno a své předměty

*Select your name with your subjects*

Vypište všechny předměty a k nim počet zapsaných studentů

*Select all subject with number of registered students*

Vypište učící učitele a jeho předměty

*Select teachers and their subjects*

Vypište učící učitele a jeho studenty

*Select teachers and their students*

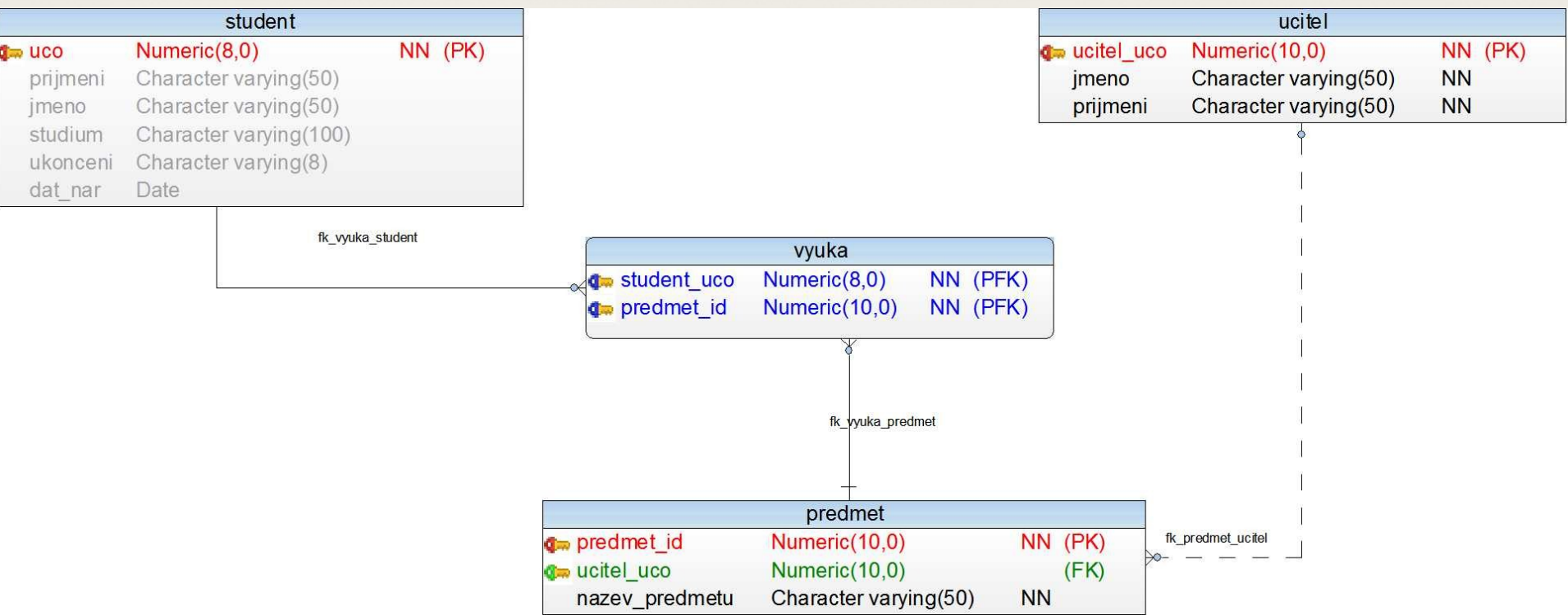
Vypište všechny učitele a počet jeho studentů

*Select all teachers and their number of students*

Vypište učitele, kteří neučí žádný předmět / studenty, kteří nemají zapsaný žádný předmět

Vypište studenty, kteří mají zapsané víc jak 2 předměty

# ER diagram



Spojení sloupců = JOIN

PatientID	Firstname	Lastname
1	Jan	Novák
2	Jana	Nová
3	Karel	Starý

+

PatientID	Date_of_exam	Result
1	12.1.2011	39,5
1	15.3.2011	36,8
2	2.2.2011	37,5

+

PhysicianID	Firstname	Lastname
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**
- **EXCEPT**                 **Rozdíl množin**
- **MINUS**                     **Rozdíl množin (ORACLE)**

SELECT sloupec FROM tabulka

**UNION**

SELECT sloupec FROM tabulka2

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



Vypište seznam všech studentů a učitelů (jméno, příjmení)

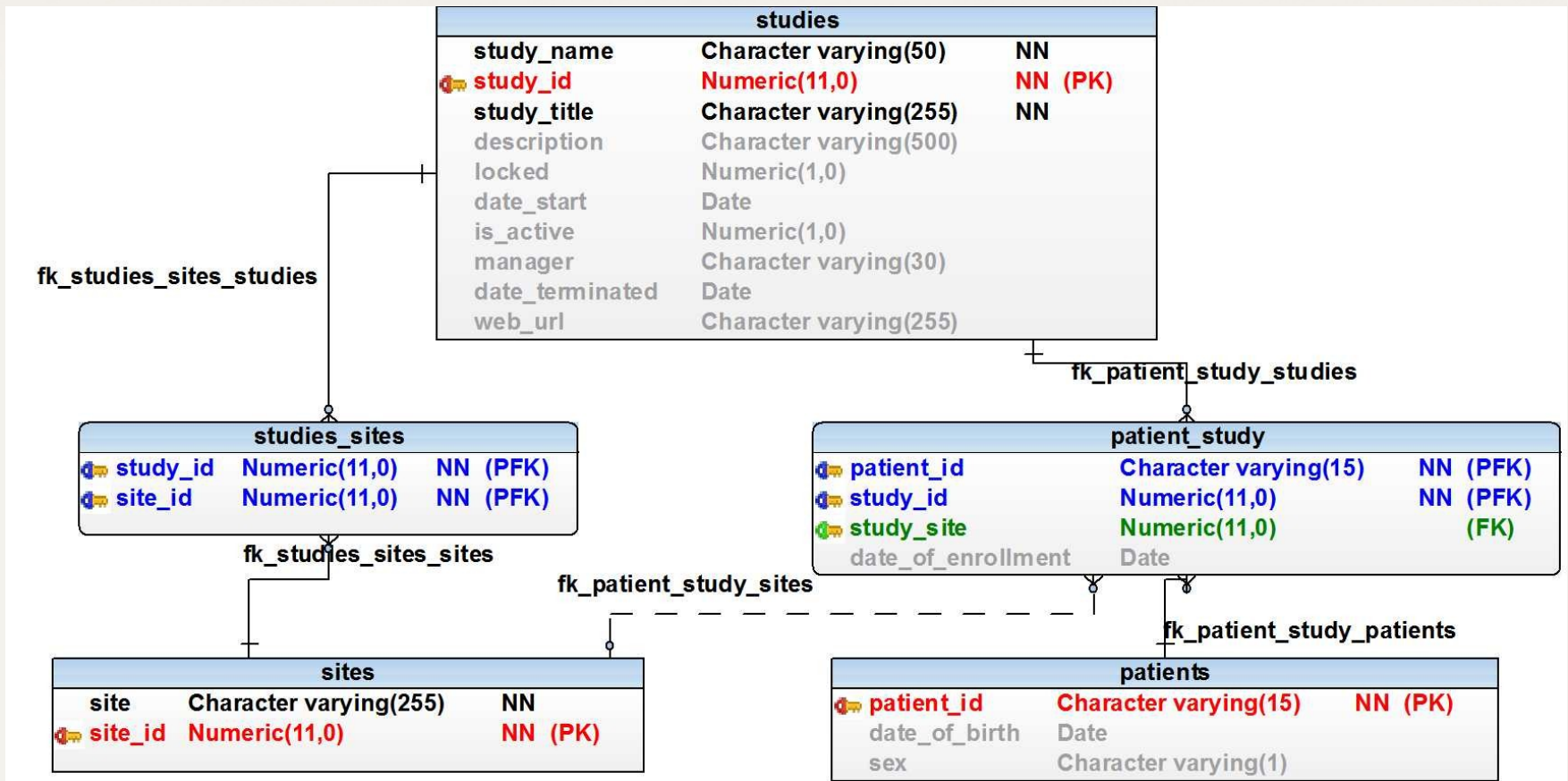
*Select firstname and lastname of students and teachers*

Přidejte jednoho učitele mezi studenty a vyzkoušejte všechny množinové operace (průnik, rozdíl)

*Add a copy of one row from table teacher to student and try all set functions*

# Another data model

patients – studies m-n => „mezitabulka“ PATIENT\_STUDY  
 studies – sites m-n => „mezitabulka“ STUDIES\_SITES



- Rozbalte skript3\_data.zip
- Spust'te skript3.sql
- Podívejte se na následující cvičení

## Zjistěte počet pacientů v jednotlivých studiích

*How many patients are enrolled in each study*

Result: STUDY\_NAME, number of patients

## Zjistěte počet pacientů dle pohlaví v jednotlivých studiích

*How many patients are enrolled in each study grouped by sex*

Result: STUDY\_NAME, sex, number of patients

## Zjistěte počet zapojených pracovišť do jednotlivých studií

*How many sites participate in each study?*

Result: STUDY\_NAME, number of sites

## Vypište pracoviště zapojená do více studií

*Select all sites, which participate in more than 1 study*

SITE, počet studií

## Vypište všechny studie a počet zařazených pacientů v jednotlivých letech

*Select all studies and number of enrolled patients in each year*

STUDY\_NAME, rok(DATE\_OF\_ENROLLMENT), počet pacientů

```

SELECT s.study_name, to_char(date_of_enrollment, 'yyyy'), count(*)
FROM patient_study ps
JOIN studies s ON ps.study_id = s.study_id
WHERE study_name = 'IKARUS'
GROUP BY s.study_id,s.study_name, to_char(date_of_enrollment, 'yyyy')
ORDER BY s.study_name, to_char(date_of_enrollment, 'yyyy')

```

```

SELECT * FROM roky

```

```

SELECT study_name , a FROM studies s
JOIN roky r ON 1=1
WHERE study_name = 'IKARUS'
ORDER BY s.study_name , r.a

```

```

SELECT study_name , a, count(ps.patient_id) FROM studies s
JOIN roky r ON 1=1
LEFT JOIN patient_study ps ON ps.study_id = s.study_id
AND to_char(ps.date_of_enrollment, 'yyyy')::integer = r.a
WHERE study_name = 'IKARUS'
GROUP BY s.study_id,s.study_name, r.a
ORDER BY s.study_name, r.a

```