

Úvod do databázových systémů

5. cvičení

Ing. Petr Lukáš
petr.lukas@nativa.cz

Ostrava, 2013

- **K čemu se používají následující konstrukce?**
 1. **IN**
 2. **EXISTS**
 3. **ALL**
 4. **ANY**
- **Co je výsledkem IN a EXISTS ?**

Opakování

- **K čemu se používají následující konstrukce?**
 1. **IN** ptáme se, jestli hodnota je v nějakém výčtu
 2. **EXISTS** testujeme existenci záznamu
 3. **ALL** testujeme, zda určité porovnání platí pro všechny z výčtu
 4. **ANY** testujeme, zda určité porovnání platí alespoň jednou pro nějaký výčet
- **Co je výsledkem IN a EXISTS ?**

Booleovská hodnota, tzn. IN a EXISTS jsou vždy součástí nějakého logického výrazu (např. v klauzuli WHERE)

Klauzule příkazu SELECT

Klauzule příkazu SELECT

SELECT	<i>seznam sloupců na výstupu</i>
FROM	<i>vstupní tabulky</i>
WHERE	<i>podmínka</i>
GROUP BY	<i>seskupení</i>
HAVING	<i>podmínka skupiny</i>
ORDER BY	<i>seřazení výsledku</i>

Ne všechny klauzule jsou povinné, ale jejich pořadí je striktně dané!

Klauzule příkazu SELECT

SELECT	<i>seznam sloupců na výstupu</i>
FROM	<i>vstupní tabulky</i>
WHERE	<i>podmínka</i>
GROUP BY	<i>seskupení</i>
HAVING	<i>podmínka skupiny</i>
ORDER BY	<i>seřazení výsledku</i>

Ne všechny klauzule jsou povinné, ale jejich pořadí je striktně dané!

- **Spojování tabulek**
- **Agregační funkce**
- **Vnořené dotazy a poddotazy**

Spojování tabulek

1. Příklad – vnitřní spojení

Student

login	jmeno	skola
nov078	Jan	1
luk194	Petr	1
lak065	Luboš	2

Skola

id	nazev
1	VŠB – TUO
2	Ostravská univerzita



Vyber jména studentů a k nim názvy škol, ve kterých studují

1. Příklad – vnitřní spojení

Student

login	jmeno	skola
nov078	Jan	1
luk194	Petr	1
lak065	Luboš	2

Skola

id	nazev
1	VŠB – TUO
2	Ostravská univerzita



Vyber jména studentů a k nim názvy škol, ve kterých studují

```
SELECT jmeno, nazev  
FROM Student, Skola  
WHERE Student.skola = Skola.id
```



jmeno	nazev
Jan	VŠB – TUO
Petr	VŠB – TUO
Luboš	Ostravská univerzita

1. Příklad – vnitřní spojení

Student

login	jmeno	skola
nov078	Jan	1
luk194	Petr	1
lak065	Luboš	2

Skola

id	nazev
1	VŠB – TUO
2	Ostravská univerzita

↓ ↓
Vyber jména studentů a k nim názvy škol, ve kterých studují

```
SELECT
  jmeno, nazev
FROM
  Student, Skola
WHERE
  Student.skola = Skola.id
```

=

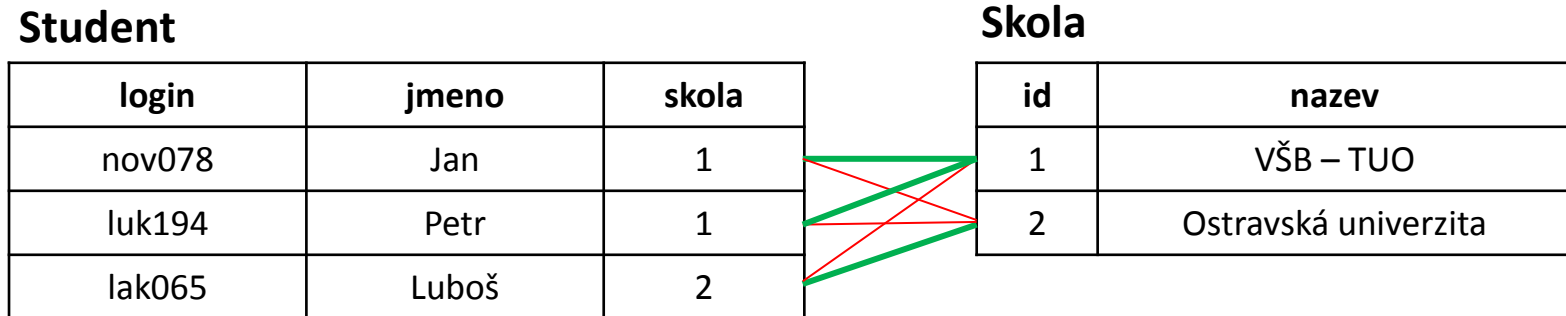
```
SELECT
  jmeno, nazev
FROM
  Student
JOIN Skola ON Student.skola = Skola.id
```

↓

jmeno	nazev
Jan	VŠB – TUO
Petr	VŠB – TUO
Luboš	Ostravská univerzita

1. Příklad – vnitřní spojení

- Stále platí představa, jako bychom nejprve udělali kartézský součin a následně „profiltrovali“ jen smysluplné kombinace záznamů.



```
SELECT jmeno, nazev  
FROM Student JOIN Skola ON Student.skola = Skola.id
```

jmeno	nazev
Jan	VŠB – TUO
Petr	VŠB – TUO
Luboš	Ostravská univerzita

2. Příklad – vnější spojení

Student

login	jmeno	skola
nov078	Jan	1
luk194	Petr	1
lak065	Luboš	2

Skola

id	nazev
1	VŠB – TUO
2	Ostravská univerzita



Vyber názvy všech škol a jména studentů, kteří v nich studují

2. Příklad – vnější spojení

Student

login	jmeno	skola
nov078	Jan	1
luk194	Petr	1
lak065	Luboš	2

Skola

id	nazev
1	VŠB – TUO
2	Ostravská univerzita



Vyber názvy všech škol a jména studentů, kteří v nich studují

```
SELECT nazev, jmeno  
FROM Skola JOIN Student ON Skola.id = Student.skola
```



nazev	jmeno
VŠB – TUO	Jan
VŠB – TUO	Petr
Ostravská univerzita	Luboš

- **Je výsledek správně?** – V tuto chvíli ano, ale...

2. Příklad – vnější spojení

Student

login	jmeno	skola
nov078	Jan	1
luk194	Petr	1
lak065	Luboš	2

Skola

id	nazev
1	VŠB – TUO
2	Ostravská univerzita
3	ČVUT v Praze



Vyber **názvy všech škol** a jména studentů, kteří v nich studují

```
SELECT nazev, jmeno  
FROM Skola JOIN Student ON Skola.id = Student.skola
```



nazev	jmeno
VŠB – TUO	Jan
VŠB – TUO	Petr
Ostravská univerzita	Luboš

- Je výsledek správně? – **NE**, ve výsledku není ČVUT.

2. Příklad – vnější spojení

Student

login	jmeno	skola
nov078	Jan	1
luk194	Petr	1
lak065	Luboš	2

Skola

id	nazev
1	VŠB – TUO
2	Ostravská univerzita
3	ČVUT v Praze



Vyber **názvy všech škol** a jména studentů, kteří v nich studují

```
SELECT nazev, jmeno  
FROM Skola LEFT JOIN Student ON Skola.id = Student.skola
```



nazev	jmeno
VŠB – TUO	Jan
VŠB – TUO	Petr
Ostravská univerzita	Luboš
ČVUT v Praze	NULL

Hodnota **NULL**

- Vyjadřuje prázdnou hodnotu (nevyplněno)
- Pro test na hodnotu **NULL** se používá speciální operátor **IS NULL**.
- Jakékoli jiné porovnání s hodnotou **NULL** vrací výsledek „nepravda“.

Vyber všechny osoby, kde není vyplněna adresa.

```
SELECT *  
FROM Osoba  
WHERE adresa = NULL
```

```
SELECT *  
FROM Osoba  
WHERE adresa IS NULL
```

3. Příklad – spojovací podmínka

Student

login	jmeno	skola
nov078	Jan	1
luk194	Petr	1
lak065	Luboš	2

Skola

id	nazev
1	VŠB – TUO
2	Ostravská univerzita
3	ČVUT v Praze



Vyber názvy všech škol a jména studentů, kteří je navštěvují a jejichž login začíná na „L“

```
SELECT nazev, jmeno  
FROM Skola LEFT JOIN Student ON Skola.id = Student.skola  
WHERE login LIKE 'L%'
```



nazev	jmeno
VŠB – TUO	Petr
Ostravská univerzita	Luboš

- **Pozor, filtrování WHERE probíhá až po spojení tabulek.**
- Podmínka WHERE vyloučí záznam o ČVUT, protože jediný login pro ČVUT po spojení tabulek bude NULL.

3. Příklad – spojovací podmínka

Student

login	jmeno	skola
nov078	Jan	1
luk194	Petr	1
lak065	Luboš	2

Skola

id	nazev
1	VŠB – TUO
2	Ostravská univerzita
3	ČVUT v Praze



Vyber názvy všech škol a jména studentů, kteří je navštěvují a jejichž login začíná na „L“

```
SELECT nazev, jmeno
FROM
  Skola
LEFT JOIN Student ON (Skola.id = Student.skola AND login LIKE 'L%')
```



nazev	jmeno
VŠB – TUO	Petr
Ostravská univerzita	Luboš
ČVUT v Praze	<i>NULL</i>

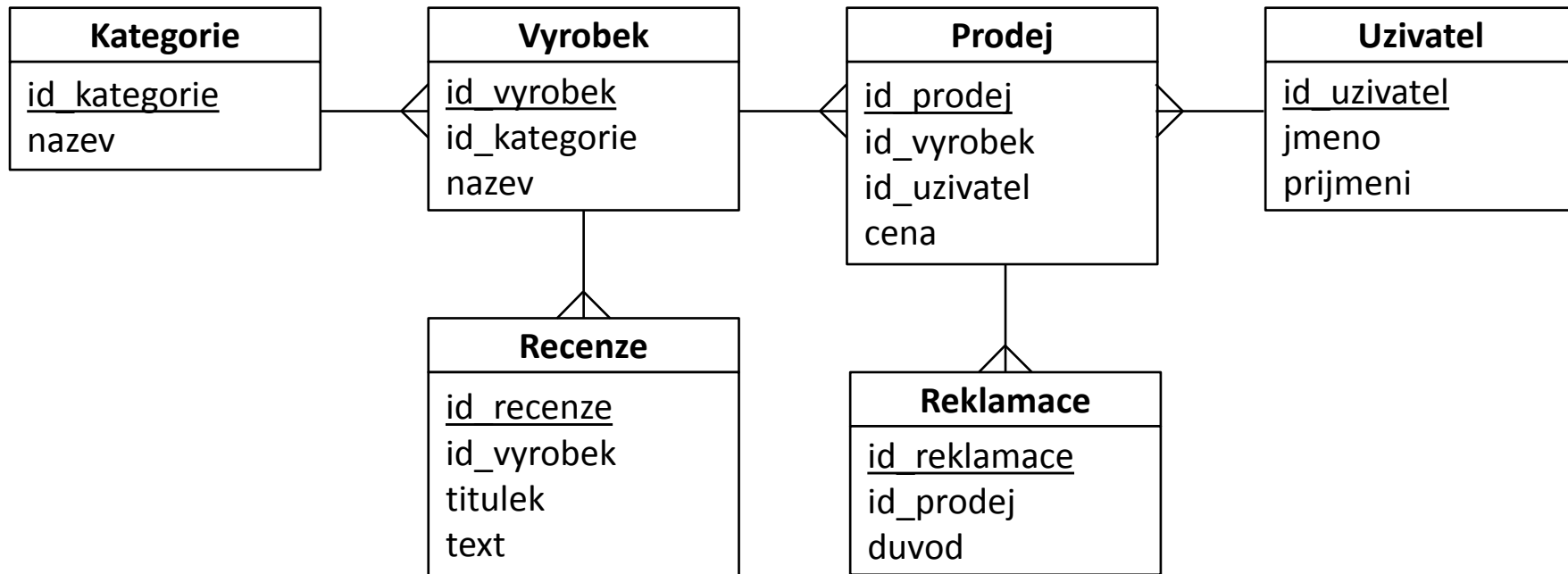
INNER JOIN

Vnitřní spojení – ekvivalent spojování pomocí podmínky WHERE

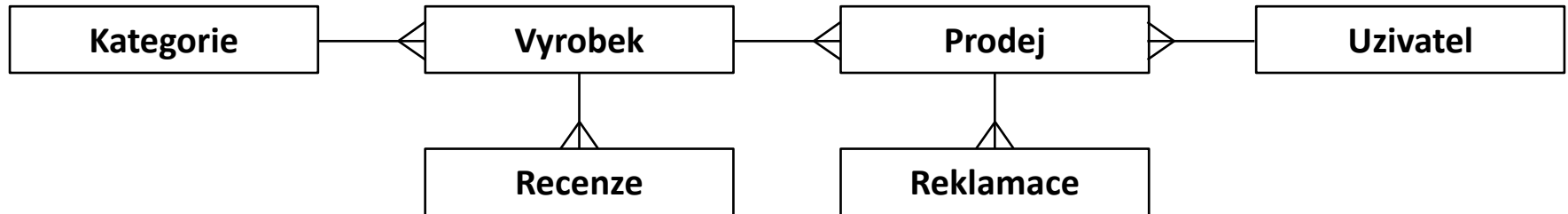
OUTER JOIN

Vnější levé nebo pravé spojení, které navíc zachovává všechny záznamy z levé nebo pravé tabulky.

Spojování tabulek



4. Příklad – spojování více tabulek



Vypište nadpisy recenzí, recenzované produkty a jejich kategorie.

```
SELECT
```

```
  Recenze.Nadpis, Vyrobek.Nazev, Kategorie.Nazev
```

```
FROM
```

```
  Recenze
```

```
JOIN Vyrobek ON Recenze.id_vyrobek = Vyrobek.id_vyrobek
```

```
JOIN Kategorie ON Vyrobek.id_kategorie = Kategorie.id_kategorie
```

Agregační funkce

5. Příklad - minimum

Vyrobyky

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

ceny z www.alfacomp.cz ke dni 15.10.2013



Vyberte cenu nejlevnějšího notebooku.

5. Příklad - minimum

Vyrobky

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

ceny z www.alfacomp.cz ke dni 15.10.2013



Vyberte cenu nejlevnějšího notebooku.

```
SELECT MIN(cena) AS [nejmensi_cena]  
FROM Vyrobky
```



nejmensi_cena
8 990 Kč

6. Příklad – minimum se seskupením

Vyrobyky

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

ceny z www.alfacomp.cz ke dni 15.10.2013



Vyberte ceny nejlevnějších notebooků jednotlivých značek.

6. Příklad – minimum se seskupením

Vyrobyky

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

ceny z www.alfacomp.cz ke dni 15.10.2013



Vyberte ceny nejlevnějších notebooků jednotlivých značek.

```
SELECT znacka, MIN(cena) AS [nejmensi_cena]  
FROM Vyrobyky  
GROUP BY znacka
```



znacka	nejmensi_cena
Acer	10 490 Kč
HP	8 949 Kč
Apple	33 836 Kč

**Co je součástí klauzule SELECT a není v
agregační funkci, musí být uvedeno v
GROUP BY!**

7. Příklad – podmínka skupiny

Vyrobyky

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

ceny z www.alfacomp.cz ke dni 15.10.2013



Vypište značky, které prodávají nejlevnější notebook za více než 30 000 Kč.

7. Příklad – podmínka skupiny

Vyrobky

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

ceny z www.alfacomp.cz ke dni 15.10.2013



Vypište značky, které prodávají nejlevnější notebook za více než 30 000 Kč.

```
SELECT znacka
FROM Vyrobky
GROUP BY znacka
HAVING MIN(cena) > 30000
```



znacka
Apple



Agregační funkce

COUNT (<i>attr</i>)	Počet
COUNT (DISTINCT <i>attr</i>)	Počet různých
SUM (<i>attr</i>)	Součet
AVG (<i>attr</i>)	Průměr
MIN (<i>attr</i>)	Minimum
MAX (<i>attr</i>)	Maximum

Agregační funkce přeskakují NULL hodnoty

Postup zpracování SQL dotazu

Vypište značky začínající na „A“, které prodávají nejlevnější notebook za více než 30 000 Kč.
Setřídte tyto značky podle abecedy vzestupně.

```
SELECT
    značka
FROM
    Vyrobyky
WHERE
    značka LIKE 'A%'
GROUP BY
    značka
HAVING
    MIN(cena) > 30000
ORDER BY
    značka
```

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

Postup zpracování SQL dotazu

Vypište značky začínající na „A“, které prodávají nejlevnější notebook za více než 30 000 Kč.
Seřadíte tyto značky podle abecedy vzestupně.

```
SELECT
    značka
FROM
    Vyrobyky
WHERE
    značka LIKE 'A%'
GROUP BY
    značka
HAVING
    MIN(cena) > 30000
ORDER BY
    značka
```

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

Postup zpracování SQL dotazu

Vypište značky začínající na „A“, které prodávají nejlevnější notebook za více než 30 000 Kč.
Setřídte tyto značky podle abecedy vzestupně.

```
SELECT
    značka
FROM
    Vyrobyky
WHERE
    značka LIKE 'A%'
GROUP BY
    značka
HAVING
    MIN(cena) > 30000
ORDER BY
    značka
```

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč



id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

Postup zpracování SQL dotazu

Vypište značky začínající na „A“, které prodávají nejlevnější notebook za více než 30 000 Kč.
Setřídte tyto značky podle abecedy vzestupně.

```
SELECT
  značka
FROM
  Vyrobyky
WHERE
  značka LIKE 'A%'
GROUP BY
  značka
HAVING
  MIN(cena) > 30000
ORDER BY
  značka
```

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč



id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

Postup zpracování SQL dotazu

Vypište značky začínající na „A“, které prodávají nejlevnější notebook za více než 30 000 Kč.
Setřídte tyto značky podle abecedy vzestupně.

```
SELECT
  značka
FROM
  Vyrobyky
WHERE
  značka LIKE 'A%'
GROUP BY
  značka
HAVING
  MIN(cena) > 30000
ORDER BY
  značka
```

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč



id_vyrobku	nazev	znacka	cena
5	Apple MacBook Air 13"	Apple	33 836 Kč

Postup zpracování SQL dotazu

Vypište značky začínající na „A“, které prodávají nejlevnější notebook za více než 30 000 Kč.
Setřídte tyto značky podle abecedy vzestupně.

```
SELECT
  značka
FROM
  Vyroby
WHERE
  značka LIKE 'A%'
GROUP BY
  značka
HAVING
  MIN(cena) > 30000
ORDER BY
  značka
```

id_vyrobku	nazev	znacka	cena
5	Apple MacBook Air 13"	Apple	33 836 Kč



id_vyrobku	nazev	znacka	cena
5	Apple MacBook Air 13"	Apple	33 836 Kč

Postup zpracování SQL dotazu

Vypište značky začínající na „A“, které prodávají nejlevnější notebook za více než 30 000 Kč.
Setřídte tyto značky podle abecedy vzestupně.

```
SELECT
  značka
FROM
  Vyroby
WHERE
  značka LIKE 'A%'
GROUP BY
  značka
HAVING
  MIN(cena) > 30000
ORDER BY
  značka
```

id_vyrobku	nazev	znacka	cena
5	Apple MacBook Air 13"	Apple	33 836 Kč



znacka
Apple

Vnořené dotazy a poddotazy

8. Příklad – vnořené dotazy

Vyrobyky

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

ceny z www.alfacomp.cz ke dni 15.10.2013



Vyberte maximální průměrnou cenu notebooků různých značek.

8. Příklad – vnořené dotazy

Vyrobky

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

ceny z www.alfacomp.cz ke dni 15.10.2013



Vyberte maximální průměrnou cenu notebooků různých značek.

Vyberte průměrnou cenu notebooků různých značek.

```
SELECT znacka, AVG(cena) AS [prumer]  
FROM Vyrobky  
GROUP BY znacka
```



znacka	prumer
Acer	15 240 Kč
HP	10 469 Kč
Apple	33 836 Kč

8. Příklad – vnořené dotazy

Vyrobky

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

ceny z www.alfacomp.cz ke dni 15.10.2013



Vyberte maximální průměrnou cenu notebooků různých značek.

Vyberte průměrnou cenu notebooků různých značek.

```
SELECT znacka, AVG(cena) AS [prumer]
FROM Vyrobky
GROUP BY znacka
```



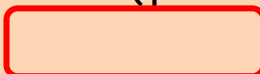
znacka	prumer
Acer	15 240 Kč
HP	10 469 Kč
Apple	33 836 Kč

Vyberte maximální průměrnou cenu notebooků různých značek.

```
SELECT MAX(prumer) AS [maximum]
FROM 
```



maximum
33 836 Kč



8. Příklad – vnořené dotazy

Vyrobky

id_vyrobku	nazev	znacka	cena
1	Acer TravelMate P253-E	Acer	10 490 Kč
2	HP 650	HP	8 949 Kč
3	HP ProBook 4540s	HP	11 990 Kč
4	Acer Aspire V7-581G-53334G52akk	Acer	19 990 Kč
5	Apple MacBook Air 13"	Apple	33 836 Kč

ceny z www.alfacomp.cz ke dni 15.10.2013



Vyberte maximální průměrnou cenu notebooků různých značek.

```
SELECT MAX(prumer) AS [maximum]
FROM
(
  SELECT znacka, AVG(cena) AS [prumer]
  FROM Vyrobky
  GROUP BY znacka
) prumery
```

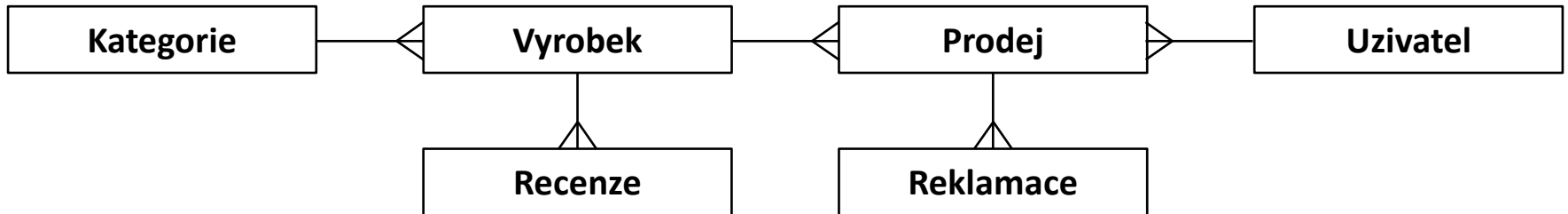


maximum
33 836 Kč

Vnořené dotazy

- Místo tabulky jako zdrojové relace použijeme celý vnořený SELECT.
- Vnořený SELECT **musí být uzávorkovaný a pojmenovaný**. Všechny jeho **sloupce musí být pojmenované**.
- Výsledkem vnořeného selectu je relace.

9. Příklad – poddotazy



Vypište všechny výrobky a k nim počty prodejů a recenzí.

```
SELECT
```

```
    Vyrobek.Nazev, COUNT(Recenze.id_recenze), COUNT(Prodej.id_prodej)
```

```
FROM
```

```
    Vyrobek
```

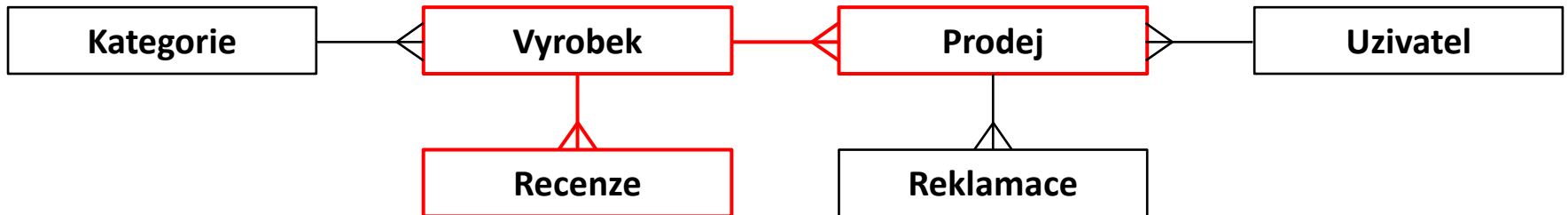
```
    LEFT JOIN Recenze ON Recenze.id_vyrobek = Vyrobek.id_vyrobek
```

```
    LEFT JOIN Prodej ON Vyrobek.id_vyrobek = Prodej.id_vyrobek
```

```
GROUP BY
```

```
    Vyrobek.Nazev
```

9. Příklad – poddotazy

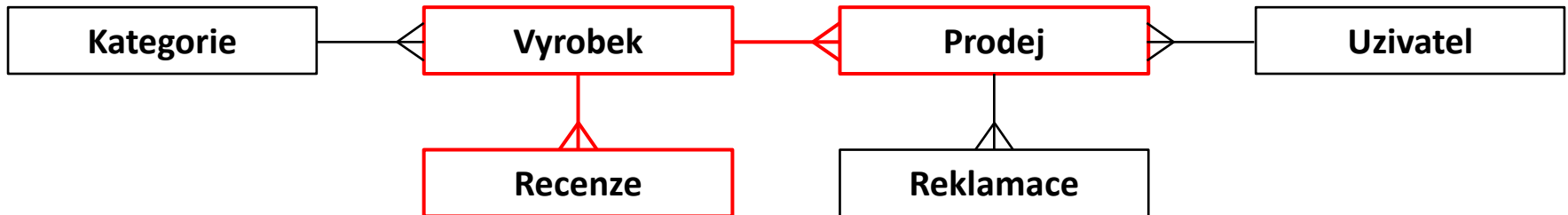


Vypište všechny výrobky a k nim počty prodejů a recenzí.

```
SELECT
  Vyrobek.Nazev, COUNT(Recenze.id_recenze), COUNT(Prodej.id_prodej)
FROM
  Vyrobek
  LEFT JOIN Recenze ON Recenze.id_vyrobek = Vyrobek.id_vyrobek
  LEFT JOIN Prodej ON Vyrobek.id_vyrobek = Prodej.id_vyrobek
GROUP BY
  Vyrobek.Nazev
```

Při spojování nesmíme jedinou tabulku vícekrát navázat vztahem směrem 1:N.

9. Příklad – poddotazy



Vypište všechny výrobky a k nim počty prodejů a recenzí.

```
SELECT
```

```
  Vyrobek.Nazev,
```

```
  (
```

```
    SELECT COUNT(Recenze.id_recenze)
```

```
    FROM Recenze
```

```
    WHERE Recenze.id_vyrobek = Vyrobek.id_vyrobek
```

```
  ) AS [pocet_recenzi],
```

```
  (
```

```
    SELECT COUNT(Prodej.id_prodej)
```

```
    FROM Prodej
```

```
    WHERE Prodej.id_vyrobek = Vyrobek.id_vyrobek
```

```
  ) AS [pocet_prodeju]
```

```
FROM
```

```
  Vyrobek
```

Výpočet počtu
recenzí

Výpočet počtu
prodejů

Poddotazy

- Musí **vracet jedinou hodnotu** (jeden řádek, jeden sloupec)
- Lze je použít kdekoli, např. v klauzuli SELECT, v WHERE, ve spojovací podmínce za JOIN ... ON, v HAVING.
- Velice užitečné, pokud počítáme agregace nad dvěma nazávislými tabulkama – tzn. takovýma, kde si je nemůžeme dovolit jednoduše spojit.
- V podstatě jimi lze simulovat „funkci“

Vnořené dotazy vs. Poddotazy

Vnořené dotazy

```
SELECT T.nazev, SUM(T.pocet)
FROM
  (
    SELECT nazev, pocet
    FROM ...
    WHERE ...
  ) T
WHERE ...
```

- Vnořené dotaz v klauzuli FROM
- Sám o sobě představuje tabulku
- Všechny sloupce musí být pojmenovány
- Dotaz jakožto „zdroj dat“ musí mít název (alias)

Poddotazy

```
SELECT
  (
    SELECT SUM(pocet)
    FROM ...
    WHERE ...
  ) AS hodnota
FROM ...
WHERE ...
```

- Poddotaz v klauzuli SELECT
- Poddotaz můžeme použít kdykoli potřebujeme spočítat jednu jedinou hodnotu
- Poddotaz musí vracet jediný řádek a jediný sloupec

1. Spojování tabulek

- Používání vnitřního spojení
- Používání vnějšího spojení
- Využití spojovací podmínky
- Pozor na spojování nezávislých tabulek

2. Agregční funkce

- COUNT, SUM, AVG, MIN, MAX
- Význam klauzule GROUP BY
- Význam klauzule HAVING

3. Vnořené dotazy a poddotazy

- Používání vnořených dotazů místo tabulek
- Používání poddotazů pro „doptání“ na určitou hodnotu

www.dbedu.cs.vsb.cz

- Přihlášení přes **jednotný login a heslo**
- Vpravo sloupec -> *České kurzy* -> *UDBS*

Příprava na test

- V archivu kurzů 2012/2013 na 6. cvičení naleznete loňský skript databáze z testu
- V letošním ročníku naleznete zadání testů
- **Procvičovat!**