

SQL – úvod

Práce s daty, 9. března 2023

Relační databáze

- Patrně nejpoužívanější způsob ukládání (propojených) dat
- Má základy v matematice, tzv. relačním modelu (kalkulu)
 - selekce, projekce, kartézský součin, rozdíl, sjednocení, spojení
- Data jsou uložena v tabulkách, které lze propojovat
 - Co řádek, to údaje o jedné entitě (osoba, předmět, studijní program, ...)
 - Co sloupec (atribut), to konkrétní údaj (jméno, rodné číslo, pohlaví, e-mail, ...)
 - K propojování slouží sloupce s funkcí klíče – obvykle jednoznačného identifikátoru

Jazyk SQL

- Standardizovaný jazyk pro práci s relačními databázemi
- Zkratka od [Structured Query Language](#)
- Do značné míry univerzální napříč systémy různých výrobců
- Navržen tak, aby co nejvíce připomínal běžné anglické věty:
 - CREATE TABLE tabulka (definice atributů tabulky);
 - SELECT něco FROM tabulka;
 - INSERT INTO tabulka (atribut1, atribut2) VALUES(hodnota1, hodnota2);
 - UPDATE tabulka SET hodnota WHERE podmínka;
 - DELETE FROM tabulka WHERE podmínka;
 - DROP TABLE tabulka;

Přehled hlavních databázových systémů

- Komerční
 - Oracle
 - Microsoft SQL Server (MSSQL)
- Volně dostupné
 - MySQL
 - MariaDB – „fork“ MySQL z roku 2009, kdy Oracle koupil MySQL
 - PostgreSQL
 - SQLite
 - Microsoft SQL Server Express

Příkaz SELECT – projekce

Projekcí vybíráme, které atributy (sloupce) zobrazíme

```
SELECT sloupec1, sloupec2, ... FROM tabulka;
```

- Na velikosti písmen nezáleží, každý příkaz je ale nutno ukončit středníkem

Příklady:

- SELECT * FROM knihy2022;
- SELECT title, author FROM knihy2022;
- SELECT **DISTINCT** item_status FROM knihy2022;
 - Odstraní duplicity
- SELECT min(year) **AS nejstarsi** FROM knihy2022;
 - Vybraný (či vypočítaný) atribut pojmenuje

Agregační funkce jazyka SQL

- COUNT – počet, např. `SELECT COUNT(*) FROM tabulka` = počet řádků
- SUM – součet, např. `SELECT SUM(mzda) FROM zamestnanci`
- MAX – maximum, např. `SELECT MAX(vek) FROM zamestnanci`
- MIN – minimum, např. `SELECT MIN(teplota) FROM mereni`
- AVG – průměr, např. `SELECT AVG(teplota) FROM mereni`
- VARIANCE – rozptyl
- STD – směrodatná odchylka

Příkaz SELECT – řazení a omezení výpisu

Řazení lze zajistit souslovím `ORDER BY` na konci příkazu:

- `SELECT * FROM knihy2022 ORDER BY barcode ASC|DESC;`
 - Celá tabulka seřazená dle čárového kódu vzestupně nebo sestupně
- `SELECT * FROM knihy2022 ORDER BY year, price;`
 - Kombinace atributů jakožto klíčů, podle kterých má být řazeno
- `SELECT * FROM knihy2022 ORDER BY RAND();`
 - Náhodné řazení

Výpis lze omezit slovem `LIMIT` úplně na konci příkazu:

- `SELECT * FROM knihy2022 ORDER BY key LIMIT 10;`

Cvičení I

Nad cvičnou tabulkou knihy2022 zjistěte:

1. Průměrnou cenu knížek v tabulce
2. Celkovou uhrazenou částku dělenou počtem všech řádků v tabulce
 - Proč je výsledek jiný než v bodě 1?
3. Název knihy a její cenu v USD
4. Seznam roků seřazený od nejnovějšího, bez duplicit
5. Název a cenu nejdražšího titulu

<https://sqliteonline.com/>

Cvičení I – řešení

1. Průměrnou cenu knížek v tabulce
 - `select avg(price) from knihy2022;`
2. Celkovou uhrazenou částku dělenou počtem všech řádků v tabulce
 - `select sum(price)/count(*) from knihy2022;`
3. Název knihy a její cenu v USD
 - `select title, price/22 AS cena_v_USD from knihy2022;`
4. Seznam roků seřazený od nejnovějšího, bez duplicit
 - `select distinct year from knihy2022 order by year DESC;`
5. Název a cenu nejdražšího titulu
 - `select title, price from knihy2022 order by price desc limit 1;`

Příkaz SELECT – selekce

Selekcí (ve smyslu relačního kalkulu) vybíráme, které řádky budou zahrnuty do projekce (tj. zobrazeny nebo započítány).

```
SELECT projekce FROM tabulka WHERE podmínka;
```

- Případné řazení a omezení výpisu následuje až za podmínkou

Příklady

- SELECT barcode FROM knihy2022 WHERE call_no='FIL-103';
- SELECT DISTINCT title FROM knihy2022 WHERE no_loans>70;
- SELECT author FROM knihy2022 WHERE author like '%Jiří';
- SELECT id FROM knihy2022 WHERE price IS Null;

Tvorba podmínek v jazyce SQL

- Relační operátory
 - = != < > <= >= (je rovno, je různé od, je menší, je větší, ...)
 - BETWEEN – je mezi, např. `price BETWEEN 200 AND 300`
- Logické spojky
 - AND, && – a zároveň
 - OR, || – nebo
 - () – závorky mění prioritu vyhodnocení podmínek
- Jiné porovnání hodnot
 - LIKE – odpovídá řetězci, např. `open_date LIKE '2004-08%'`
 - RLIKE – odpovídá regulárnímu výrazu, např. `RLIKE '^M.+ová,.*'`
 - IN – nachází se ve vyjmenované množině hodnot, např. `IN (2003, 2017)`

Často používané funkce

- Matematické operátory
 - + - * / DIV % (... , celočíselné dělení, zbytek po dělení)
- Matematické funkce
 - ROUND, FLOOR, CEILING – zaokrouhlování (přirozené, dolů, nahoru)
 - RAND – vrací náhodné reálné číslo v rozsahu od 0 do 1
- Funkce pro práci s řetězci
 - CONCAT, CONCAT_WS – spojování řetězců (se zadaným oddělovačem)
 - LOCATE(substr, str) – vrací pozici podřetězce v řetězci
 - SUBSTR(str, pos, len) – vrací část řetězce dané délky od dané pozice
 - TRIM, LTRIM, RTRIM – maže (obvykle) prázdné místo okolo řetězce

Cvičení II

Nad cvičnou tabulkou knihy2022 vypište:

1. Kolik jednotek knih nebylo ani jednou vypůjčeno
2. Id knih vydaných mezi lety 2015 až 2020 s cenou menší než 200 nebo větší než 2000
3. Posledních pět číslic čárového kódu knih, které v názvu obsahují slovo *marketing*
4. Počet jednotek, jejichž *call_no* začíná na SOC, PRA nebo VÝCH
5. Jména autorů v pořadí *Křestní jméno Příjmení*
6. Dolů zaokrouhlený průměrný rozdíl v letech mezi *open_date* a *update_date* u jednotek se zadanou signaturou (*call_no*)

Cvičení II – řešení, první části

1. Kolik jednotek knih nebylo ani jednou vypůjčeno

```
SELECT count(*) FROM knihy2020 WHERE no_loans=0;
```

2. Id knih vydaných mezi lety 2015 až 2020 s cenou menší než 200 nebo větší než 2000

```
SELECT id, title, price FROM knihy2022 WHERE year  
BETWEEN 2015 AND 2020 AND (price<200 OR price>2000);
```

3. Posledních pět číslic čárového kódu knih, které v názvu obsahují slovo *marketing*

```
SELECT substr(barcode,6,5) FROM knihy2022 WHERE title  
LIKE '%marketing%';
```

Cvičení II – řešení druhé části

1. Počet jednotek, jejichž *call_no* začíná na SOC, PRA nebo VÝCH

```
SELECT count(*) FROM knihy2022 WHERE  
substr(call_no, 1, 3) IN ('SOC', 'PRA', 'VÝC');
```

2. Jména autorů v pořadí *Křestní jméno Příjmení*

```
select concat_ws(' ', substr(author, locate(',', author) +2),  
substr(author, 1, locate(',', author)-1)) from knihy2022;
```

Nebo s pomocí funkce REGEXP_SUBSTR

3. Dolů zaokrouhlený průměrný rozdíl v letech mezi *open_date* a *update_date* u jednotek se zadanou signaturou (*call_no*)

```
select floor(avg(substr(update_date,1,4)-  
substr(open_date,1,4))) from knihy2022 where call_no!='';
```

Zdroje ke samostudiu

- E-kniha [Learning MySQL : get a handle on your data](#)
 - Zejména její třetí kapitola
- [MariaDB Knowledge Base](#)
- Seriál o MySQL na [Linuxsoft.cz](#)
- SQL tutoriál na [W3schools.org](#)
 - Nespočet dalších tutoriálů lze samozřejmě dohledat
- Kniha *Myslete databázově, myslete v SQL!*
 - Snad již brzy v naší knihovně