

Zadání

S databází <https://is.muni.cz/auth/el/phil/podzim2024/ISKB56/um/01/books.sqlite>

1. Zobrazte všechny knihy vydané po roce 2000.
2. Vypište knihy, kde původní jazyk je angličtina a byly vydány před rokem 1980.
3. Zobrazte knihy u kterých původní název obsahuje slovo "History.". Knihy seřadte podle roku vydání v sestupném pořadí
4. Spočítejte počet knih vydaných v jednotlivých zemích, ale zobrazte ty pouze země, kde bylo vydáno více než 3 knihy. U každé země vypište i počet knih a to ve sloupečku nazvaném pocet_knih
5. Vypište počet knih vydaných jednotlivými vydavateli v 90. letech 20. století. Počet knih vypište ve sloupečku s názvem pocet_knih. Výstup seřadte abecedně dle vydavatelů .
6. Zobrazte seznam vydavatelů, kteří vydali alespoň 2 knihy, přičemž každá kniha má název delší než 15 znaků a její název (ať už přeložený, nebo původní) začíná na **A** . Seřaď vydavatele podle počtu knih sestupně.

S databází

<https://is.muni.cz/auth/el/phil/podzim2024/ISKB56/um/01/movies.sqlite>

databáze má 70mb, takže její stažení i následné nahrání na <https://sqliteonline.com/> může chvíli trvat.

Zdroj dat <https://developer.imdb.com/non-commercial-datasets/> , z datasetu jsem použil pouze klasické filmy, čili žádné seriály, krátké filmy, atd)

Sloupečky:

tconst - ID filmu

primaryTitle – nejpoužívanější název

originalTitle - název v původním jazyce

isAdult - film pro dospělé (0 = Ne, 1 = Ano)

startYear – rok premiéry

runtimeMinutes – délka v minutách

genres – žánry

1. Zobrazte 10 nejdelších filmů žánru "Adventure" začínajících na "N" nebo "P" které měly premiéru mezi lety 1995 a 2015. Výsledek seřadte sestupně podle délky trvání.
2. Vypište průměrnou délku filmu v jednotlivých letech. Výsledek seřadte sestupně dle roku a u každého roku zároveň vypište celkový počet filmů. Do výsledku **nezahrnujte** filmy, u kterých není zadaný rok a filmy, u kterých je rok premiéry v budoucnosti.
 - a. bonus: Průměrnou délku zaokrouhlete na 2 desetinná místa (náповěda: https://www.sqlite.org/lang_corefunc.html#round)
 - b. bonus: zkuste dotaz upravit tak, aby byl univerzální a fungoval třeba i příští rok, kdy "premiéra v budoucnosti" bude rok 2026 (náповěda: https://www.sqlite.org/lang_datefunc.html)

- c. bonus: zkuste si data přenést do excelu a vytvořit vizualizaci trendu

S databází

<https://is.muni.cz/auth/el/phil/podzim2024/ISKB56/um/01/odtahy.sqlite>

1. Vypište počty odtahů dle dny v týdnu.
2. Vypište počty odtahů dle hodiny v rámci dne.

Řešení

books

1.

```
SELECT * FROM books
WHERE year_published > 2000;
```
2.

```
SELECT * FROM books
WHERE language_original = 'English' AND year_published <
1980;
```
3.

```
SELECT * FROM books
WHERE original_title LIKE '%History%'
ORDER BY year_published DESC
```
4.

```
SELECT country, COUNT(*) AS pocet_knih
FROM books
GROUP BY country
HAVING COUNT(*) > 3;
```
5.

```
SELECT publisher, COUNT(*) AS pocet_knih
FROM books
WHERE year_published >= 1990 AND year_published < 2000
GROUP BY publisher
ORDER BY publisher
```
6.

```
SELECT publisher, COUNT(*)
FROM books
WHERE LENGTH(title) > 15 AND (title LIKE 'a%' OR
original_title LIKE 'a%')
GROUP BY publisher
HAVING COUNT() >= 2
ORDER BY COUNT(*) DESC;
```

Movies:

1. SELECT primaryTitle, runtimeMinutes, startYear
FROM movie
WHERE genres LIKE '%Adventure%' AND startYear >= 1995 AND
startYear <= 2015 AND (primarytitle LIKE 'n%' OR primarytitle
LIKE 'p%')
ORDER BY runtimeMinutes DESC
LIMIT 10
pozn: Alternativně lze i startYear BETWEEN 1995 AND 2015
2. SELECT startYear, COUNT(*) AS pocet_filmu,
AVG(runtimeMinutes) AS prumerna_delka
FROM movie
WHERE startYear > 0 AND startYear < 2025
GROUP BY startYear
ORDER BY startYear DESC
 1. SELECT startYear, COUNT(*) AS pocet_filmu,
ROUND(AVG(runtimeMinutes), **2**) AS prumerna_delka
FROM movie
WHERE startYear > 0 AND startYear < 2025
GROUP BY startYear
ORDER BY startYear DESC
 2. SELECT startYear, COUNT(*) AS pocet_filmu,
ROUND(AVG(runtimeMinutes), 2) AS prumerna_delka
FROM movie
WHERE startYear > 0 AND startYear <= **strftime('%Y',
date())**
GROUP BY startYear
ORDER BY startYear DESC

Odtahy

1. SELECT strftime('%w', datum), count(*)
FROM odtahy
GROUP BY strftime('%u', datum)
2. SELECT substr(cas, 1, 2), count(*)
FROM odtahy
GROUP BY substr(cas, 1, 2)