

# **VIKMB55**

# **SQL pro pokročilé a**

# **datové modelování**

Hodina č. 9

# Test

**Vypište počet objednávek dle hodin v rámci dne. Tj. kolik objednávek bylo v 8 ráno, kolik v 9,... Ve výpisu vždy uveďte hodinu a počet objednávek.**

```
SELECT strftime("%H", vytvoreno) hodina, count(id)  
pocet_objednavek FROM objednavky GROUP BY hodina
```

**Vypište jména zákazníků, kteří udělali svoji první objednávku v lednu 2014 a poslední v prosinci 2014**

```
SELECT z.jmeno, z.prijmeni  
FROM objednavky o  
JOIN zakaznici z ON o.id_zakaznik=z.id_zakaznik  
GROUP BY o.id_zakaznik  
HAVING  
MIN(vytvoreno) LIKE "2014-01%" AND  
MAX(vytvoreno) LIKE "2014-12%"
```

**Vypište všechny knížky na “B”, které byly alespoň jednou  
objednány. K každé vypište její název, autory a žánry**

```
SELECT k.nazev,  
GROUP_CONCAT(DISTINCT a.jmeno || " " ||a.prijmeni),  
GROUP_CONCAT(DISTINCT z.zanr)  
FROM polozky_objednavky po  
JOIN knihy k ON po.id_kniha=k.id  
JOIN autori_knihy ak ON k.id=ak.knihy_id  
JOIN autori a ON ak.atori_id=a.id  
JOIN knihy_zanry kz ON k.id=kz.knihy_id  
JOIN zanry z ON kz.zanry_id=z.id  
WHERE k.nazev LIKE "B%"  
GROUP BY k.id
```

**Vypište jména zákazníka/zákazníků, kteří si objednali největší množství knih (=kusů). Připravte univerzální řešení, tedy takové, které dopředu neví, jestli takový člověk je jeden, či má stejný počet knih třeba 10 zákazníků**



```
SELECT z.jmeno, z.prijmeni
FROM polozky_objednavky po
JOIN objednavky o ON po.id_objednavka=o.id
JOIN zakaznici z ON o.id_zakaznik=z.id_zakaznik
GROUP BY o.id_zakaznik
HAVING SUM(pocet)=
(
  SELECT SUM(pocet) knih
  FROM polozky_objednavky po
  JOIN objednavky o ON po.id_objednavka=o.id
  GROUP BY o.id_zakaznik
  ORDER BY knih DESC LIMIT 1
)
```

**Vypište nakladatele, kteří vydali alespoň jednu knihu, která získala alespoň 10 hodnocení a její průměrná známka byla nižší než průměrné hodnocení pro všechny knížky v e-shopu**

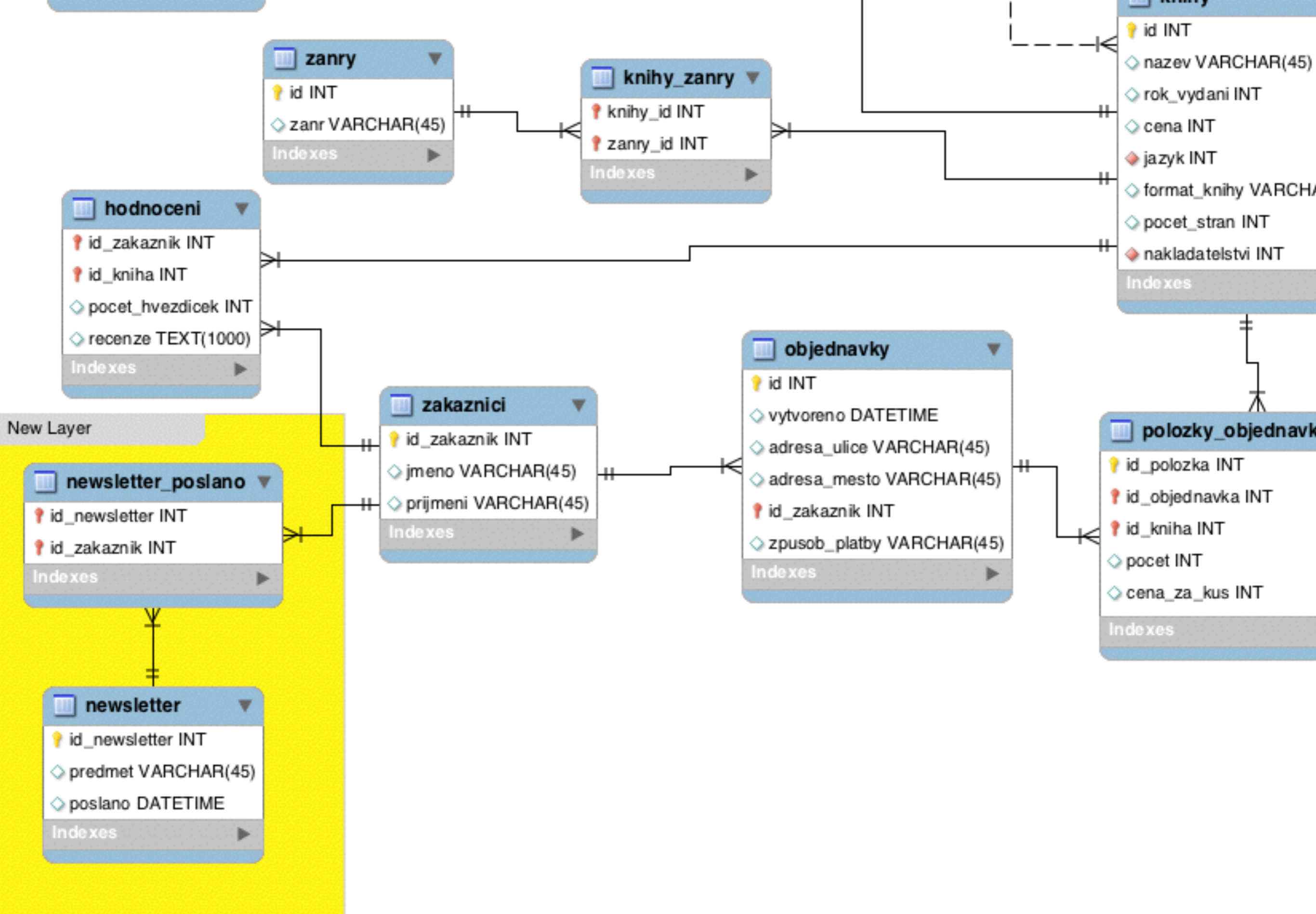
```
SELECT DISTINCT n.nazev
FROM hodnoceni h
INNER JOIN knihy k ON k.id = h.id_kniha
INNER JOIN nakladatelstvi n ON nakladatelstvi = n.id
GROUP BY id_kniha
HAVING
COUNT(*) > 10 AND
AVG(pocet_hvezdicek) <
    (SELECT AVG(pocet_hvezdicek) FROM hodnoceni)
```

**Vypište kolika zákazníkům se poslaly jednotlivé newslettery, kolik objednávek se od nich uskutečnilo do dvou dnů od jejich rozeslání (tj. v den rozeslání a den poté) a jaká byla tržba z těchto objednávek**

**Vypište kolika zákazníkům se poslaly jednotlivé newsletters,** kolik objednávek se od nich uskutečnilo do dvou dnů od jejich rozeslání (tj. v den rozeslání a den poté) a jaká byla tržba z těchto objednávek

```
SELECT np.id_newsletter,  
COUNT(DISTINCT np.id_zakaznik) as pocet  
FROM newsletter n  
LEFT JOIN newsletter_poslan np  
    ON n.id_newsletter=np.id_newsletter  
GROUP BY np.id_newsletter
```

Vypište kolika zákazníkům se poslaly jednotlivé newslettery, **kolik objednávek se od nich uskutečnilo do dvou dnů od jejich rozeslání (tj. v den rozeslání a den poté)** a jaká byla tržba z těchto objednávek





```
SELECT np.id_newsletter,  
COUNT(o.id) as objednavek,  
SUM(pocet*cena_za_kus) as trzba  
FROM newsletter n  
LEFT JOIN newsletter_poslan np  
  ON n.id_newsletter=np.id_newsletter  
LEFT JOIN objednavky o  
  ON np.id_zakaznik=o.id_zakaznik  
LEFT JOIN polozky_objednavky po  
  ON o.id=po.id_objednavka  
WHERE  
  date(o.vytvoreno) = date(n.poslano)  
  OR  
  date(o.vytvoreno) = date(n.poslano, "+1 day")  
GROUP BY np.id_newsletter
```

```
SELECT * FROM (dotaz1) LEFT JOIN (dotaz2)  
ON dotaz1.id_newsletter = dotaz2.id_newsletter
```

# **Vztažené/nevztažené poddotazy**

correlated/noncorrelated subselects

<http://produkty.jdem.cz/>

**Vypište počet produktů, které stojí méně než je průměrná  
cena.**

```
SELECT COUNT(*)  
FROM produkt  
INNER JOIN produkt_cena ON id = id_produk  
WHERE cena <  
    (SELECT AVG(cena) FROM produkt_cena)
```

**Vypište počet produktů, které stojí méně než je průměrná cena.**

```
SELECT COUNT(*)  
FROM produkt  
INNER JOIN produkt_cena ON id = id_produk  
WHERE cena <  
    (SELECT AVG(cena) FROM produkt_cena)
```

“Nevztažený poddotaz” (noncorrelated) - neobsahuje žádný odkaz na “vnější” dotaz. Tj. funguje sám o sobě a vykoná se jen jednou.

**Vypište 50 produktů, které stojí méně než 4000 korun**

Dobré řešení č. 1

```
SELECT *  
FROM produkt  
INNER JOIN produkt_cena ON id = id_produk  
WHERE cena < 4000  
LIMIT 50
```

# Vypište 50 produktů, které stojí méně než 4000 korun

Dobré řešení č. 2

```
SELECT *  
FROM produkt  
WHERE id IN (SELECT id_produk FROM produkt_cena  
WHERE cena < 4000 LIMIT 50)
```

“Nevztažený poddotaz” (noncorrelated) - neobsahuje žádný odkaz na “vnější” dotaz. Tj. funguje sám o sobě a vykoná se jen jednou.



## Vypište 50 produktů, které stojí méně než 4000 korun

Nedobré řešení

```
SELECT *  
FROM produkt  
WHERE (SELECT cena FROM produkt_cena WHERE  
id_produk = produkt.id) < 4000  
LIMIT 50
```

“Vztažený poddotaz” (correlated) - vnitřní dotaz používá políčko z vnějšího (produkt.id), musí se proto vykonat pro každý řádek výsledků zvlášť, čili 50x

# Vypište 50 produktů, které stojí méně než 4000 korun

Nedobré řešení

```
SELECT *  
FROM produkt  
WHERE (SELECT cena FROM produkt_cena WHERE  
id_produk = produkt.id) < 4000  
LIMIT 50
```

Zkuste si změnit LIMIT na 500, 5000, 10000... a koukejte se na čas, za který je dotaz vykonán (vpravo dole). Bez limitu to raději ani nezkoušejte :)

<http://rozhlas-auta.jdem.cz/>

Stáhli jsme si data set <https://samizdat.cz/data/odtahy-2016/data/bkom.xlsx>, převedli ho do CSV, importovali do SQLITE a zkoušeli najít v kolik hodin proběhlo kolik odtahů. Za pomocí <http://openrefine.org/> jsme si vyzkoušeli “vyčistit” překlepy v názvech ulic - tohle byl takový bonus, protože s SQL to nemá nic moc společného :)

# Problém s kódováním

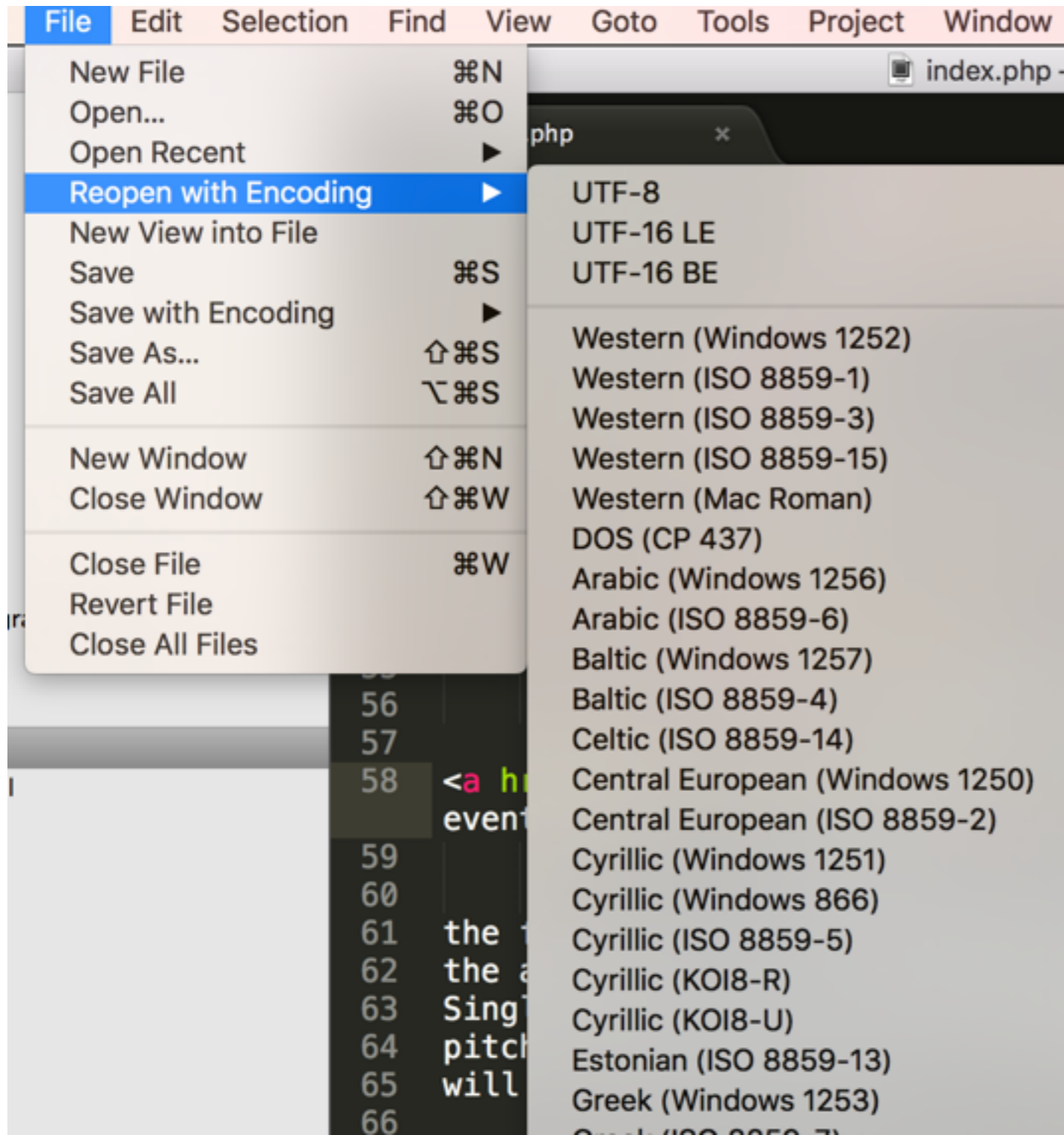
SQLite pracuje s kódováním UTF-8 (resp. umí i některá další, ale ty nepodporují české znaky), čili když se do něj snažíte nacpat data (např. CSV), která jsou v jiném kódování, tak pak máte “rozbitou češtinu” ...

08:51	TYLOVA	
08:52	TYLOVA	
08:53	TYLOVA	
08:53	GALANDAUEROVA	
08:54	BO❖ET❖CHOVA	
08:55	HRUB❖HO	
08:55	METOD❖JOVA	
08:56	GALANDAUEROVA	
08:57	MOJM❖ROVO N❖M.	

# Problém s kódováním

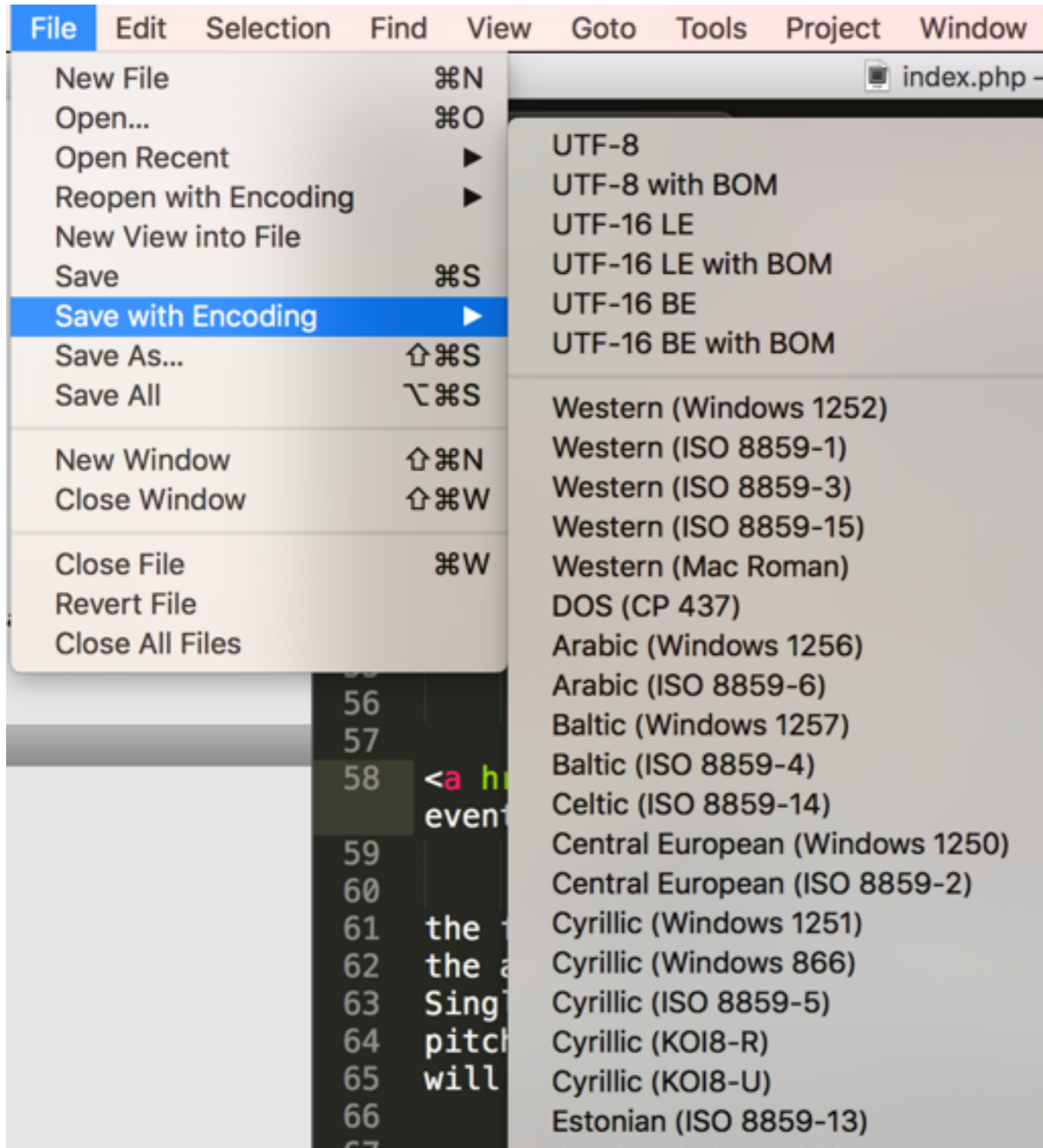
Microsoft Excel data často exportuje s kódováním windows-1250, čili musíte najít nějaký převodník do UTF. Bud' googlete "cp1250 to utf 8 online" a nebo si nainstalujte (třeba) editor <https://www.sublimetext.com/>, který umí soubor otevřít v daném kódování a v jiném ho pak uložit. Viz screenshoty na dalších slajdech

# Problém s kódováním



Můžete vyzkoušet  
obě ze sekce  
Central European,  
dokud se vám  
soubor nezobrazí  
správně

# Problém s kódováním



...a pak si to  
uložíte jako UTF-8