

## Zvolené téma:

- 04: COVID-19 (dr. Rýchly)

## Riešitelia:

- Sabína Gregušová (xgregu02)
- Peter Hamran (xhamra00)
- Adrián Tulušák (xtulus00)

## Úvod

Cieľom tohto projektu bolo zoznámenie sa so spracovaním neštruktúrovaných dát, s ich prípravou a spracovaním pre ďalšie využitie. Následne vďaka nadobudnutým vedomostiam navrhnuť nástroje, ktoré umožnia automatické spracovanie a nahranie dát do NoSQL databáze. Z NoSQL databáze následne vybrať vhodné data, ktoré je po úprave možné uložiť do vhodne navrhutej relačnej databázy. Poslednou časťou projektu je návrh a implementácia prostredia na zodpovedanie dvoch zadaných dotazov a jedného vlastného k danej téme.

Témy tohto projektu boli vopred zadané a pre riešenie nášho projektu sme si vybrali tému COVID-19. Dáta využité na zodpovedanie dotazov pochádzajú z verejne dostupných zdrojov. Na návrh systémov sme použili viacero voľne dostupných technológií a programovacích jazykov, ktoré spolu vytvárajú celok.

## Príprava dát

Tento projekt sa zameriava na situáciu v Českej republike (dotazy A, B) a na situáciu v Európskej únii (dotaz C) od vypuknutia pandémie COVID-19. Keďže sa jedná o stále aktuálne téma, existuje k nemu pomerne veľké množstvo dát. Podarilo sa nám získať dátové sady, ktoré sú dostačujúce pre zodpovedanie všetkých troch dotazov. Jedná sa o:

- pre A: COVID-19: Přehled osob s prokázanou nákazou dle hlášení krajských hygienických stanic (v2)
- pre B: COVID-19: Přehled epidemiologické situace dle hlášení krajských hygienických stanic podle okresu
- pre C: Testing for COVID-19 by week and country

Tieto dátové sady obsahujú najpodstatnejšie informácie a pre ich lepšiu prezentáciu boli vytvorené a pridané 2 dátové sady. Dotazy by bolo možné zodpovedať aj bez nich, avšak v tomto prípade ide o zlepšenie čitateľnosti výsledku. Jedná sa o:

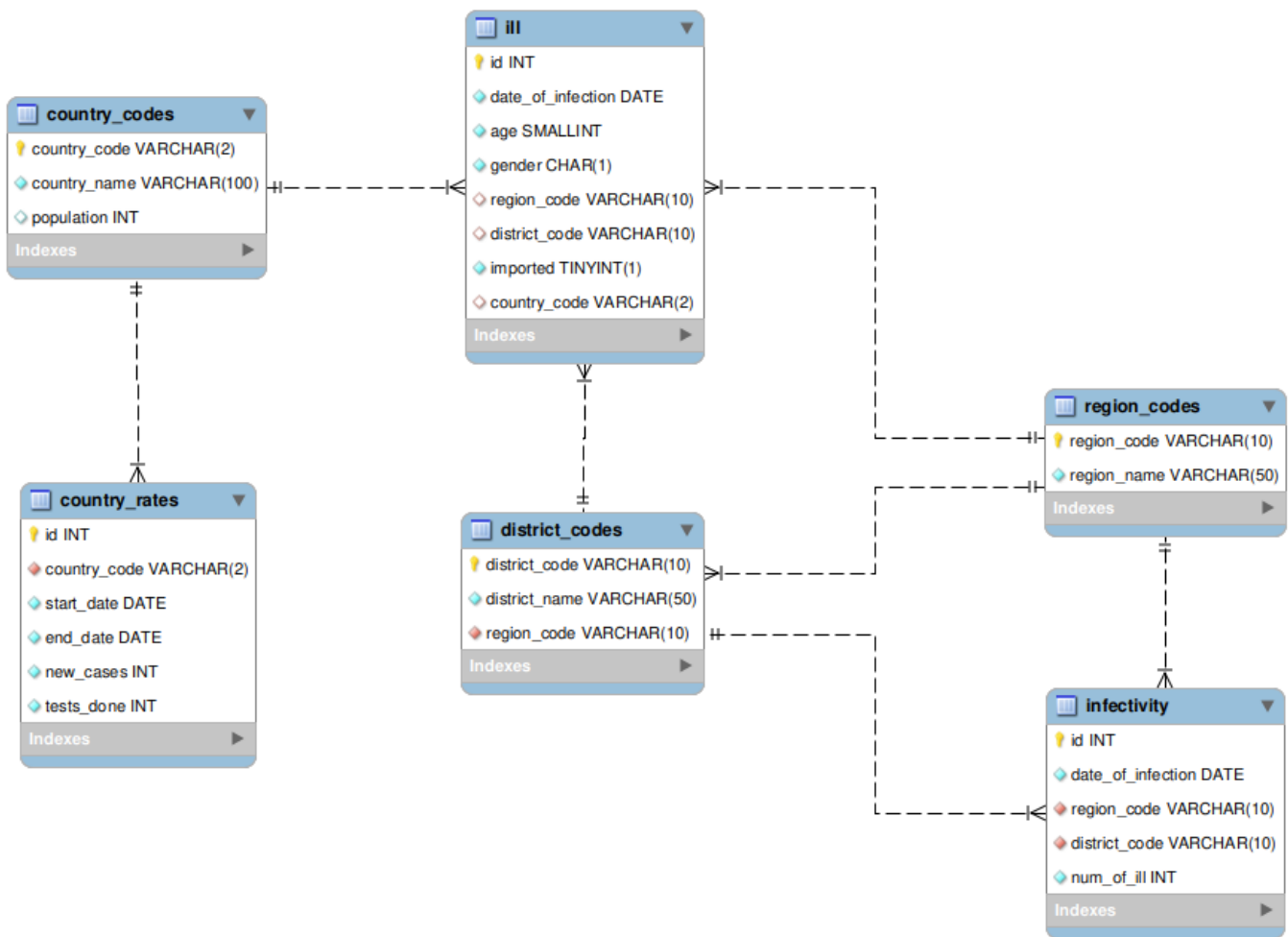
- `countries.json`: obsahuje unikátne skratky krajín a ich český ekvivalent
- `nuts_lau.json`: obsahuje kódy NUTS (kraj) a LAU (okres) a ich český ekvivalent

Dátové sady sú importované pomocou príkazu `mongoimport`. Dátové sady A a B vyžadujú dodatočné formátovanie pre prácu s podstatnými dátami. Tento proces je automatizovaný pomocou skriptu `prepData.sh`, ktorý:

- Zmaže a vytvorí novú (čistú) databázu s názvom "corona"
- Postupne stiahne, a ak je potrebné aj naformátuje, dátové sady vo formáte JSON
- Naimportuje každú dátovú sadu ako kolekciu; kolekcie sú pomenované podľa hlavnej dátovej sady (A, B, C) a novo pridané dátové sady sú pomenované ako `countries` a `nuts_lau`

Samotnú úpravu, prípravu a konverziu dát vykonáva skript `convertData.py`, ktorý sa spúšťa s Python verzou 3. Tento skript bol navrhnutý tak, aby bol ľahko modifikovateľný a rozšíriteľný, preto využíva princípy objektovo orientovaného návrhu. Na začiatku sa pripojí ku NoSQL a MySQL databázi. Vytvorí relačnú databázu v jazyku SQL - kód vytvorenia relačnej databáze sa nachádza v samostatnom súbore s názvom `Corona.sql`, ktorý je preparovaný a jednotlivé príkazy sú vykonané v rámci skriptu `convertData.py`. Toto riešenie zaisťuje, že samotný skript obsahuje minimum „natvrdo“ napísaných príkazov pre vytvorenie relačnej databáze.

Samotná relačná databáza obsahuje dokopy 6 tabuliek - 3 hlavné tabuľky pre zodpovedanie dotazov a 3 pomocné. Teoreticky by bolo možné použiť iba 3 hlavné tabuľky, ale mnohé dáta by boli redundantné a nevedlo by to k vhodnému návrhu. Slová ako `district` a `region` sú často zameniteľné, no v celom tomto projekte je slovo `district` považované za okres a `region` za kraj.



Obrázek 1: Schéma relačnej databázy

V skripte `convertData.py` sme navrhli všeobecnú funkciu `obtainImportantData`, ktorá chystá a upravuje všetky dáta. Táto funkcia dostane na vstupe dáta z MongoDB, zoznam záhlaví, ktoré chceme extrahovať a zoznam obmedzení pre jednotlivé záhlavia.

Demonštrácia použitia:

```
self.obtainImportantData(C, ['population', 'country_code'], ['', 'UPPER_C']),
```

kde `C` je kolekcia všetkých dát pre dotaz `C` z `mongoDB`, druhý parameter obsahuje zoznam záhlaví (dát), ktoré chceme v tomto prípade extrahovať (populácia a kód krajiny), a na záver je zoznam obmedzení rovnakej dĺžky ako bol zoznam záhlaví. Prázdna položka v zozname obmedzení značí "žiadne obmedzenie" a obmedzenie "UPPER\_C" prekonvertuje všetky kódy krajín do veľkých písmen. Vďaka tomuto návrhu spracovania dát je veľmi jednoduché modifikovať a rozširovať relačnú databázu a samotnú úpravu dát.

## Dotazy

Dotazy sú v projekte realizované pomocou `mysql` konektoru navrhnutého v skriptovacom jazyku Python. Tento konektor je implementovaný v súbore `query.py` a využíva princípy objektovo orientovaného návrhu z dôvodu zvýšenej prehľadnosti a zjednodušenia znovupoužiteľnosti. Takto navrhnutý objekt je následne využívaný na pozadí grafického rozhrania pre získavanie dát z relačnej databázy.

Hlavný objekt obaluje inicializáciu spojenia s relačnou databázou. Jednotlivé SQL doazy sú realizované formou metód tohto objektu, ktorých vstupy sú parametrami SQL dotazov a výstupmi je hodnota alebo pole hodnôt na ktoré sa dotazujeme.

## **Grafické rozhranie**