

Niestety nie miałem nigdy do czynienia z frameworkiem Apache Spark, a ze językiem Scala miałem styczność jedynie podczas kilku ćwiczeń na studiach, więc do pierwszego zadania wybrałem dobrze mi znaną Javę.

Zadanie 1.

Status.java

Napisałem klasę Status reprezentującą model danych z pliku *statuses.json*. Konstruktor przyjmuje z parametru JSONObject i rzutuje dane po etykietach do odpowiednich pól klasy.

Klasa implementuje interfejs `COMPARABLE` w celu porównywania obiektów przy sortowaniu (nadpisana metoda *compareTo*).

Nazwy pól pozostawiłem takie jak w etykiety w pliku z danymi.

Converter.java

Ta klasa zawiera tylko metodę main, w której odbywa się konwersja danych do pliku **.csv*.

Zauważyłem, że plik *statuses.json* zawiera tablicę obiektów, więc zaimportowałem ją do obiektu typu JSONArray.

Stamtąd każdy JSONObject rzutuję do Status i umieszczam na ArrayListie, którą sortuję za pomocą metody *compareTo*.

Każdy obiekt z ArrayListy konkatenuję do Stringa csv według schematu z treści zadania. Stringa zapisuję do pliku *statuses.csv*.

Zadanie 2.

Zadanie2.sql

```
SELECT `kontakt_id`, `klient_id`, `pracownik_id`, `status`, max(`kontakt_ts`)
FROM `statuses` GROUP BY klient_id
```

W tym zapytaniu potrzebuję wybrać ostatni status klienta, więc użyłem funkcji *max()*, która zwraca największą wartość (w tym przypadku chodzi o atrybut *kontakt_ts*).

Dane pogrupowałem po *klient_id*.

Zadanie 3.

Zadanie3.sql

```
ALTER TABLE statuses  
ADD dzien date;
```

```
UPDATE statuses  
SET dzien=kontakt_ts;
```

```
INSERT INTO `f_docieralnosc`(`data`, `sukcesy`, `utrasy`, `do_ponowienia`)  
SELECT dzien,  
COUNT(CASE status WHEN "zainteresowany" THEN 1 ELSE null END),  
COUNT(CASE status WHEN "niezainteresowany" THEN 1 ELSE null END),  
COUNT(CASE status WHEN "poczta_glosowa" THEN 1 WHEN "nie_ma_w_domu" THEN 1 ELSE null  
END)  
FROM statuses  
GROUP BY dzien;
```

To zadanie pozwoliłem sobie rozbić na 3 zapytania.

- 1) Dodałem kolumnę *dzien* typu *date*
- 2) Wypełniłem ją danymi
- 3) Właściwe zapytanie - tabelę uzupełniam unikatowym dniem (dzięki GROUP BY *dzien*), oraz liczbami określającymi odpowiednie wartości miar
- wyjaśnię jedną z nich, pozostałe przebiegają analogicznie

```
COUNT(CASE status WHEN "zainteresowany" THEN 1 ELSE null END)
```

"W przypadku, gdy status przyjmuje wartość "zainteresowany" zaliczamy go 1 raz w przeciwnym pomijamy"

Dostarczone dane nie pozwoliły zweryfikować poprawności zapytania, więc dodałem kilka rekordów do tabeli statuses, aby uzyskać bardziej "miarodajne" informacje.

Do pierwszego zadania wykorzystałem środowisko IntelliJ IDEA, natomiast do zadań 2 i 3 użyłem phpMyAdmin.

Dane z pliku statuses.csv z powodzeniem zaimportowałem do phpMyAdmin.

Wydaje mi się, że nazwa atrybutu `data` może nie być bezpieczna, tym razem się udało.