Dokumentacja

Projekt

Pimp My Wheels

Bazy danych

Prowadzący kurs: dr Tomasz Stroiński

Grupa VII

28 czerwca 2024

1 Użyte technologie

1. Python w wersji 3.12.2
2. Jupyter Notebook w wersji 2024.5.0
3. Pakiety do Pythona:
    - mysql_connector w wersji 2.2.9
    - numpy w wersji 2.0.0
    - pandas w wersji 2.2.2
    - scipy w wersji 1.14.0
    - SQLAlchemy w wersji 2.0.31
    - Unidecode w wersji 1.3.8
    - openpyxl w wersji 3.1.3
4. R w wersji 4.3.1.
5. RStudio w wersji 2023.09.0.
6. Pakiety do R:
    - RMariaDB w wersji 1.3.1,
    - tidyverse w wersji 2.0.0,
    - ggplot2 w wersji 3.4.3,
    - eeptools w wersji 1.2.5,
    - scales w wersji 1.2.1,
    - moments w wersji 0.14.1,
    - DescTools w wersji 0.99.54.
7. ERD Editor w wersji 1.0.20

2 Spis plików

2.1 Dokumentacja

1. projekt.vuerd.json - schemat bazy danych stworzony w programie ERD Editor.
2. schemat_bazy.png - zrzut ekranu schematu bazy danych.
3. dokumentacja.pdf - aktualnie czytany plik zawierający dokumentację projektu.

2.2 Wypełnianie

1. inputs - folder zawierający zbiór wszystkich danych pobranych z internetu w celu generowania losowych wartości do tabel.
2. outputs - folder zawierający zbiór wynikowych tabel wkładanych do bazy danych.
3. osoby.py - pilk zawierający funkcje służące do generowania danych osobowych klientów i pracowników.

- Imiona[1] i nazwiska[2] generowane są na podstawie danych osób występujących w rejestrze PESEL, im popularniejsze imię/nazwisko tym większa szansa na jego wystąpienie.
- PESEL i data urodzenia losowane są równolegle, bo są ze sobą powiązane.
- Miasta losowane są z danych[3] GUS'u z 2021 roku. Z największym prawdopodobieństwem wylosowany jest Wrocław. Jeśli nie jest to Wrocław to losowane jest województwo (im dalsze od Dolnego Śląska tym mniejsza szansa), a następnie losowane jest miasto z województwa (im więcej mieszkańców tym większe prawdodpodobieństwo)
- Ulica losowana jest ze zbioru ulic[4], a numer domu i potencjalnie mieszkania to losowe liczby naturalne.
- Numer telefonu ma z dużym prawdopodobieństwem polski numer kierunkowy, a numer osobowy jest całkiem losowy.
- Adres e-mail generowany jest na bazie imienia, nazwiska i daty urodzenia osoby, chociaż nie zawsze wszystkie dane są użyte.

4. daty_transakcji.py - plik zawierający funkcje służące do generowania takich dat transakcji, które nie są ustawowo dniami wolnymi od pracy[5]. Data wielkanocy w danym roku liczona jest na podstawie algorytmu Gauss'a[6].

5. wypelnianie_tabel.ipynb - plik wypełniający wszystkie tabele w folderze outputs, tak aby dane w tabelach miały sens.

6. wypelnianie_bazy.ipynb - plik wypełniający bazę danych na serwerze danymi z tabel w folderze outputs.

7. czesci.py - plik służący do pobrania danych dotyczących części z pliku czesci_pl.csv [7]. Oryginalny plik został przetłumaczony oraz zmodyfikowany na potrzeby projektu.

8. samochody.py - plik zawierający funkcje generujące dane dotyczące samochodów oraz motocykli z plików samochody.csv oraz motocykle.csv.
   - Marka i model samochodu zostały wygenerowane na podstawie danych dotyczących sprzedanych przez portal AutoScout24 samochodów[8] oraz popularnych marek i modeli motocykli uzyskanych przez webscrapping[9].
   - Rok produkcji został wygenerowany z rozkładu normalnego $\mathcal{N}(2008, 4.5)$ a następnie ograniczony z góry przez 2023 i skorygowany (odjęte 10 lat), w przypadku braku zgodności z datami transakcji dotyczącymi pojazdu.
   - Dla samochodów dane dotyczące skrzyni biegów, mocy oraz typu paliwa były losowane z odpowiednimi wagami zależnymi od wylosowanego modelu. Natomiast dla motocykli, ze względu na brak danych były losowane przy użyciu rozkładu jednostajnego. W przypadku motocykli typ paliwa to zawsze "Benzyna".
   - Pojemność silnika została wyznaczona na podstawie mocy pojazdu z dodaniem losowego czynnika z rozkładu normalnego.

[1] https://dane.gov.pl/pl/dataset/1667,lista-imion-wystepujacych-w-rejestrze-pesel-osoby-zyjacel

[2] https://dane.gov.pl/pl/dataset/1681,nazwiska-osob-zyjacych-wystepujace-w-rejestrze-pesel

[3] https://stat.gov.pl/obszary-tematyczne/ludnosc/ludnosc/powierzchnia-i-ludnosc-w-przekroju-terytorialnym-w-2021-roku,7,18.html

[4] https://eteryt.stat.gov.pl/eTeryt/rejestr_teryt/udostepnianie_danych/baza_teryt/uzytkownicy_indywidualni/pobieranie/pliki_pelne.aspx?contrast=default

[5] https://www.portalkadrowy.pl/czas-pracy/ustawa-z-18-stycznia-1951-r.-o-dniach-wolnych-od-pracy-tekst-jedn.-dz.u.-z-2020-r.-poz.-1920-6882.html

[6] https://en.wikipedia.org/wiki/Date_of_Easter#Algorithms

[7] https://github.com/atduskgreg/hypecyclist/tree/master/lists/Listofautoparts.csv

[8] https://www.kaggle.com/datasets/ander289386/cars-germany

[9] https://www.kaggle.com/datasets/nehalbirla/motorcycle-dataset

- Typy nadwozia dla samochodów oraz kolor[10] dla wszystkich pojazdów były generowane losowo z ustalonymi wagami.
- Powypadkowość przyjmowała wartość True z prawdopodobieństwem 0.15.
- Dane dotyczące liczby miejsc i liczby drzwi były generowane losowo dla samochodów, natomiast dla motocykli przyjmowały wartości odpowiednio 1 i 0.

2.3 Raport

1. raport.Rnw - plik zawierający analizę danych w formie raportu. Analizowane tematy to:
   - Odsetek naprawianych marek pojazdów.
   - Liczba naprawianych pojazdów w każdym miesiącu pracy warsztatu.
   - Sprzedaż których pojazdów przyniosła najwięcej zysku? (Tabela najlepszych okazji)
   - Profil klienta.
   - Jak wybrane cechy pojazdów wpływają na zysk warsztatu?
   - Kim są najlepszy mechanik i sprzedawca w warsztacie?
   - Analiza bilansu.
2. raport.pdf - wygenerowany wcześniej raport.

3 Kolejność i sposób uruchamiania plików

1. (Tylko przy pierwszym włączeniu) w celu zainstalowania odpowiednich pakietów do Pythona, należy wykonać w terminalu (będąc w folderze zawierającym pliki projektu) następującą komendę:

   ```
   pip install -r requirements.txt
   ```

2. W celu wygenerowania tabel do postaci plików csv należy wejść do folderu **wypełnianie**, następnie otworzyć plik **wypelnianie_tabel.ipynb** i przycisnąć przycisk `Run All` w panelu sterowania notatnika. Pliki csv pojawią się w folderze **outputs**.
3. W celu wypełnienia bazy danych wartościami z wcześniej wygenerowanych tabel należy włączyć plik **wypelnianie_bazy.ipynb** i przycisnąć przycisk `Run All`.
4. (Tylko przy pierwszym włączeniu) w celu zainstalowania pakietu do R można w RStudio w konsoli wpisać `install.packages(nazwa_pakietu)` zastępując `nazwa_pakietu` odpowiednim pakietem.
5. Aby wygenerować raport należy otworzyć plik **raport.Rnw** w RStudio i nacisnąć przycisk `Compile PDF`. W pliku są dane potrzebne do połączenia się z bazą danych, co następuje automatycznie podczas kompilacji.

4 Schemat projektu bazy danych

Zrzut ekranu schematu bazy danych oraz sam schemat bazy danych dostępne są w plikach kolejno **schemat_bazy.png** i **projekt.vuerd.json** w folderze dokumentacja.

[10] `https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcROPUxK6q2XjUMfVl_3vjCopTyqd-WzKG_RuQ&s`

5 Relacje i zależności funkcyjne

5.1 klienci

Klucze kandydujące:

- id,
- telefon,
- email.

Brak nietrywialnych zależności funkcyjnych niezaczynających się od nadkluczy.

5.2 pracownicy

Klucze kandydujące:

- id,
- telefon,
- email.

Brak nietrywialnych zależności funkcyjnych niezaczynających się od nadkluczy.

Mimo że adres email w tabeli jest postaci imię.nazwisko@kuba.pl, to nie ma zależności funkcyjnej od imienie i nazwiska, ponieważ w przypadku, gdyby imię i nazwisko się powtórzyło do adresu dodawana jest liczba (na przykład imię.nazwisko2@kuba.pl), żeby nie było powtórzeń emaila. Zależności w drugą stronę też nie ma z uwagi na utratę polskich (i nie tylko) znaków, więc nie rozróżniamy przykładowo nazwiska Święs i Swięs.

5.3 samochody

Klucze kandydujące:

- id.

Brak nietrywialnych zależności funkcyjnych nie zaczynających się od nadkluczy.

5.4 sprzedaż_samochodu

Klucze kandydujące:

- id.

Brak nietrywialnych zależności funkcyjnych nie zaczynających się od nadkluczy.

5.5 użyte_części

Klucze kandydujące:

- id.

Brak nietrywialnych zależności funkcyjnych nie zaczynających się od nadkluczy.

5.6 wyposażenie

Klucze kandydujące:

- id.

Brak nietrywialnych zależności funkcyjnych nie zaczynających się od nadkluczy.

5.7 zakup_samochodu

Klucze kandydujące:

- id.

Brak nietrywialnych zależności funkcyjnych nie zaczynających się od nadkluczy.

5.8 usługi

Klucze kandydujące:

- id.

Brak nietrywialnych zależności funkcyjnych nie zaczynających się od nadkluczy.

6 Czy baza jest EKNF?

Wszystkie tabele opisują jeden obiekt. Nie zawierają powtarzających się informacji. Wartości w kolumnach w tabelach są elementarne. Te cechy świadczą o tym że relacje są pierwszej postaci normalnej. To i brak zależności funkcyjnych nie zaczynających się od nadkluczy mówi o tym, że baza jest EKNF.

7 Co było najtrudniejsze podczas realizacji projektu?

Podczas wykonywania projektu zmierzyliśmy się z wieloma wyzwaniami, zarówno na etapie projektowania bazy, jej uzupełniania i pisania raportu. Mimo tego, że dokładnie przemyśleliśmy strukturę bazy ostatecznie okazało się, że musieliśmy ją zmodyfikować podczas uzupełniania jej danymi, tak aby nie przechowywać danych, które sprawiłyby, że baza nie jest EKNF. Najtrudniejszą częścią okazało się to, by być w stanie pogodzić nasze ambicje z tym, że mamy na wykonanie projektu określony czas. Chcieliśmy, by baza miała jak najwięcej wspólnego z rzeczywistością i jednocześnie nie tworzyć projektu, który będzie wymagał od nas rozpatrywania irracjonalnie dużej ilości warunków.

Dużym technicznym problemem okazało się ładowanie tabel na serwer baz danych. Metoda wgrywania table na serwer, którą wybraliśmy wymagała identycznej struktury tabel w plikach csv i schemacie bazy danych. Było to jednak przydatne, bo pozwoliło na dużą kontrolę w budowaniu bazy i zapobieganiu niezgodności między różnymi częściami projektu. Dużym problemem również było naprawienie błędów, ponieważ informacje o błędach nie były czytelne. Zapisywanie i odczytywanie typów danych też przysporzyło wiele trudności, na przykład numer PESEL odczytujący się jako liczba zmiennoprzecinkowa, mimo że zapisywano ją jako napis.