Duombazės

1 SQL

1.1 Telecom

* Jums duota csv failo nuoroda:

<https://raw.githubusercontent.com/nikipaj1/teaching/main/Analytics/telecom_churn.csv> - užkraukite jį per pandas ir panaudodami pandas .to\_sql funkciją ir sqlite3, sukurkite lokalią duombazę task1.db su lentele telecom, į kurią įkelkite visus duomenis iš telecom dataseto

Atsidarykite duombazę su sqlite database browser arba dbeaver. Viską, ką darysite, SQL queries surašykite po klausimu

* Parašykite SQL query, kad paimtumėte pirmus 5 įrašus iš telecom lentelės

#SELECT \* FROM telecom LIMIT 5

* Parašykite SQL query, kuris patikrintų, ar lentoje esantys stulpeliai neturi NULL reikšmių (arba paskaičiuokite, kiek NULL reikšmių turi kiekvienas stulpelis)

#SELECT \* FROM telecom WHERE State IS NULL

# SELECT count(\*) FROM telecom WHERE State IS NULL;

* Paskaičiuokite, kaip pasiskirsčiusios Churn stulpelio reikšmės

#SELECT count(\*) FROM telecom WHERE Churn == 1;

#SELECT count(\*) FROM telecom WHERE Churn == 0;

* Surikiuokite lentos eilutes pagal "Churn" ir "Total day charge" stulpelius, churn didėjimo tvarka ir total day charge mažėjimo tvarka

SELECT Churn, "Total day charge" FROM telecom ORDER By Churn ASC,"Total day charge" DESC

* Kokios yra vidutinės numerinių stulpelių reikšmės klientams, kurie churnino?

* Kokia maksimali tarptautinių skambučių trukmė tarp lojalių vartotojų (Churn = 0), kurie neturi tarptautinio plano?

SELECT "Total intl minutes" from telecom where Churn = 0 AND "International plan" ='No'

* Sugrupuokite lentą pagal "Churn" stulpelį ir paskaičiuokite ("Total day minutes", "Total eve minutes", "Total night minutes") stulpelių minimalią, maksimalią, vidutinę reikšmes.
* SELECT min("Total day minutes"),min("Total eve minutes"),min("Total night minutes") FROM telecom ORDER by Churn;
* SELECT max("Total day minutes"),max("Total eve minutes"),max("Total night minutes") FROM telecom ORDER by Churn;
* SELECT avg("Total day minutes"),avg("Total eve minutes"),avg("Total night minutes") FROM telecom ORDER by Churn;

1.2

2 SQL in Python

2.1 Airport

Įisivaizduokite, jog turite dalį oro uosto duombazės, kurioje yra dvi lentos: User ir Flight. Abi lentos pavaizduotos apačioje



* Panaudodami python ir sqlite3, sukurkite naują duomabazę task2.db su abiejom lentelėm.
* Parašykite konsolinę programą, kuri leistų pridėti naują informacija apie skrydžius (destination, airplane\_id, max\_num\_people), o num people, kad padarytų 0. Pridėkite kelias skirtingas kryptis, kad Flight lentoje būtų duomenų.
* Parašykite konsolinę programą, kurioje žmogus turi pridėti savo vardą, pavardę, pasport\_id ir flight\_id (patikrinkite, kad flight\_id jau būtų)
* Pridėję naują User įrašą, turite prie atitinkamos Flight eilutės, kuri atitinka User’io pasirinktą flight\_id, pridėti prie num\_people vieną žmogų.
* Jeigu num\_people = max\_num\_people, programa turi uždrausti User’iui imti šį skrydį ir grąžinti informaciją , kad skrydis pilnai užimtas ir jo rinktis jau negalima.

3 ORM and SQLAlchemy

3.1 Apparel Ecommerce

Paveiklėlyje žemiau pavaizduotas duomenų modelis drabužių parduotuvės sandeliavimo aplikacijoje. Naudodamiesi diagrama, atlikite žemiau pateiktas užduotis.

​​

* Aprašykite visus santykius tarp lentų - pvz., many2many/one2many/many2one etc.

apparel size one to one product / categories one to one product categories /

product many to many / collor one to one / product collor one to manny

* Lentelėse, kurios yra tarpinės, trūksta po stulpelį - kokių trūksta, kad tarpinės lentelės turėtų prasmę?

product categories product id /aperal size truksta onother data

* Naudodamiesi SQLalchemy, sukurkite modelį pagal pateiktą diagramą.
* Parašykite programą, kuri naudojantis jūsų sukurtu modeliu leistų:
* Pridėti kategoriją
* Pridėti spalvą
* Pridėti dydį (praignoruokite **sort order** toje lentoje)
* Pridėti produktą - kokie stulpeliai turi būti pridėti prie **other data Product** lentoje, kad ji būtų logiška?

Galbut apparel size

3.1 Organisation



Aukš"čiau pateiktas *workday* įmonės duomenų modelio pavyzdys. Workday dirba su daug skirtingų įmonių ir padeda atlikti tokius veiksmus kaip naujų darbuotojų onboardinimą, darbuotojo prijungimą prie departamento, HR valdymą.

Įsivaizduokite, jog esate workday pradinių dienų vienas pirmųjų inžinierių ir jums reikia sukurti sistemą nuo nulio.

Naudodamiesi SQLalchemy, sukurkite modelį pagal pateiktą diagramą.

Parašykite programą, kuri naudojantis jūsų sukurtu modeliu leistų:

* Pridėti Organizaciją
* Pridėti organizacijos padalinį (department)
* Pridėti Projektą
* Pridėti naujus darbuotojus (person)