Название и общее описание проекта:

Название - Клиенты и счета.

Есть база с данными о клиентах и счетах в формате csv.

Цели проекта с описанием бизнес-задачи и требованиями:

Необходимо построить на основе существующих данных витрины на дату 2020-11-01

1. Витрина _corporate_payments_. Строится по каждому уникальному счету (AccountDB и AccountCR) из таблицы Operation.

Поле	Описание
ClientId	ИД клиента (PK)
ClientName	Наименование клиента
Type	Тип клиента (ФЛ, ЮЛ)
Form	Организационно-правовая форма (ООО, ИП и т.п.)
RegisterDate	Дата регистрации клиента
TotalAmt	Сумма операций по всем счетам клиент. Считается как сумма corporate_account.total_amt по всем счетам.
CutoffDt	Дата операции

2. Витрина _corporate_account_. Строится по каждому уникальному счету из таблицы Operation на заданную дату расчета.

Поле	Описание
AccountId	ИД счета
ClientId	Ид клиента счета
PaymentAmt	Сумма операций по счету, где счет клиента указан в дебете проводки
EnrollementAmt	Сумма операций по счету, где счет клиента указан в кредите проводки
TaxAmt	Сумму операций, где счет клиента указан в дебете, и счет кредита 40702
ClearAmt	Сумма операций, где счет клиента указан в кредите, и счет дебета 40802
CarsAmt	Сумма операций, где счет клиента указан в дебете проводки и назначение платежа не содержит слов по маскам Списка 1
FoodAmt	Сумма операций, где счет клиента указан в кредите проводки и назначение платежа содержит слова по Маскам Списка 2
FLAmt	Сумма операций с физ. лицами. Счет клиента указан в дебете проводки, а клиент в кредите проводки – ФЛ.
CutoffDt	Дата операции;

3. Витрина _corporate_info_. Строится по каждому уникальному клиенту из таблицы Operation.

Поле	Описание
AccountID	ИД счета
AccountNum	Номер счета
DateOpen	Дата открытия счета
ClientId	ИД клиента
ClientName	Наименование клиента
TotalAmt	Общая сумма оборотов по счету. Считается как сумма PaymentAmt и EnrollementAmt
CutoffDt	Дата операции

Список 1:

%а/м%, %а\м%, %автомобиль %, %автомобили %, %транспорт%, %трансп%средс%, %легков%, %тягач%, %вин%, %vin%,%vin:%, %ford%, %форд%,%kia%, %кия%, %киа%%mitsubishi%, %мицубиси%, %nissan%, %ниссан%, %scania%, %bmw%, %бмв%, %audi%, %ауди%, %jeep%, %джип%, %volvo%, %вольво%, %тоуота%, %тойота%, %тоиота%, %hyundai%, %хендай%, %renault%, %рено%, %peugeot%, %пежо%, %lada%, %лада%, %datsun%, %додж%, %mercedes%, %мерседес%, %volkswagen%, %фольксваген%, %skoda%, %шкода%, %самосвал%, %rover%, %ровер%

Список 2:

%сою%, %соя%, %зерно%, %кукуруз%, %масло%, %молок%, %молоч%, %мясн%, %мясо%, %овощ%, %подсолн%, %пшениц%, %рис%, %с/х%прод%, %с/х%товар%, %с\х%прод%, %с\х%товар%, %сахар%, %сельск%прод%, %сельск%товар%, %сельхоз%прод%, %сельхоз%товар%, %семен%, %семечк%, %сено%, %соев%, %фрукт%, %яиц%, %ячмен%, %картоф%, %томат%, %говя%, %свин%, %курин%, %куриц%, %рыб%, %алко%, %чаи%, %кофе%, %чипс%, %напит%, %бакале%, %конфет%, %колбас%, %морож%, %с/м%, %с\м%, %консерв%, %пищев%, %питан%, %сыр%, %макарон%, %лосос%, %треск%, %саир%, % филе%, % хек%, %хлеб%, %какао%, %кондитер%, %пиво%, %ликер%

Описание исходных данных:

1. Таблица клиентов 10 000 записей

Поле	Описание
ClientId	ИД клиента (PK)
ClientName	Наименование клиента
Type	Тип клиента (ФЛ, ЮЛ)
Form	Организационно-правовая форма (ООО, ИП и т.п.)
RegisterDate	Дата регистрации клиента

2. Таблица счетов – 20 000 записей

Поле	Описание
AccountId	ИД счета (PK)
AccountNum	Двадцатизначный номер счета
ClientId	ИД клиента владельца счета (FK)
DateOpen	Дата открытия счета

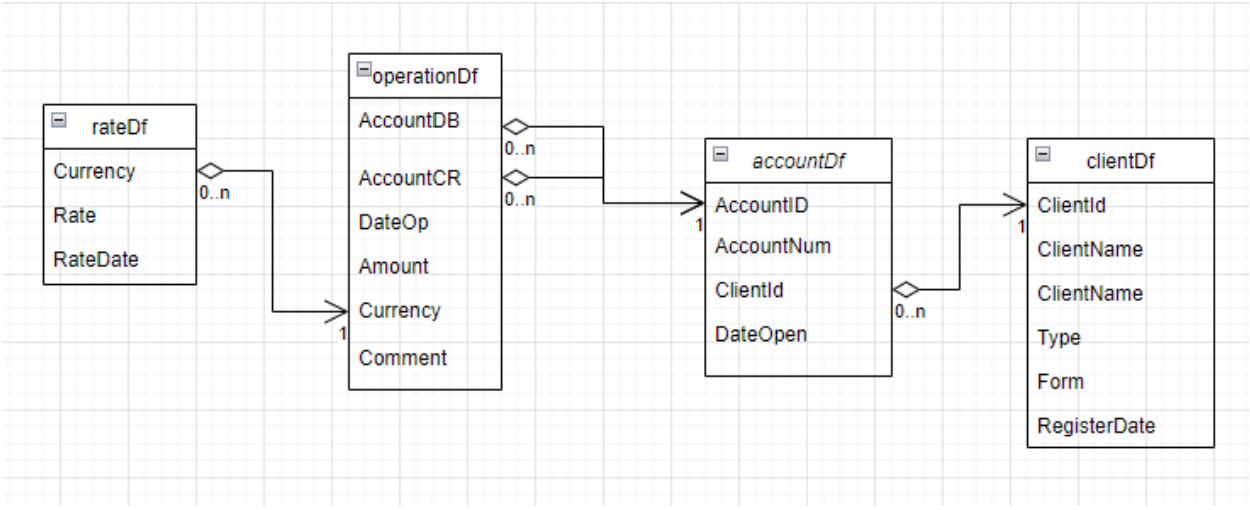
3. Операции по счетам – 100 000 записей

Поле	Описание
AccountDB	Счет дебета проводки (FK)
AccountCR	Счет кредита проводки (FK)
DateOp	Дата операции
Amount	Сумма операции
Currency	Валюта операции
Comment	Назначение платежа

4. Курсы валют по отношению к рублю

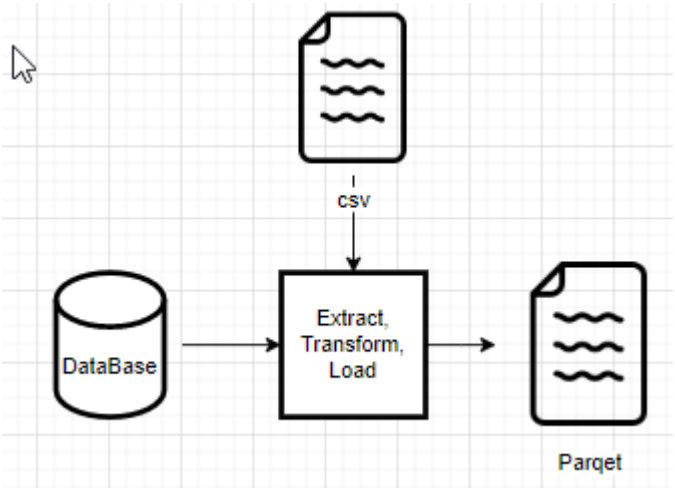
Поле	Описание
Currency	Валюта
Rate	Курс
RateDate	Дата курса.

Схема связей исходных данных:



План реализации:

1) Загрузка данных из файлов в формате csv
2) Загрузка данных со списками из БД
3) Суммы операций из csv перевести на актуальный курс
4) Каждую витрину преобразовать в файл в формате Parqet



Используемые технологии:

Технологический стек – spark, scala, sql.

Spark используем так как предполагается что исходных данных будет очень много.