Описание задачи.

Разработать ETL процесс, получающий ежедневную выгрузку данных (предоставляется за 3 дня), загружающий ее в хранилище данных и ежедневно строящий отчет.

Выгрузка данных.

Ежедневно некие информационные системы выгружают три следующих файла:

1. Список транзакций за текущий день. Формат – CSV.
2. Список терминалов полным срезом. Формат – XLSX.
3. Список паспортов, включенных в «черный список» - с накоплением с начала месяца. Формат – XLSX.

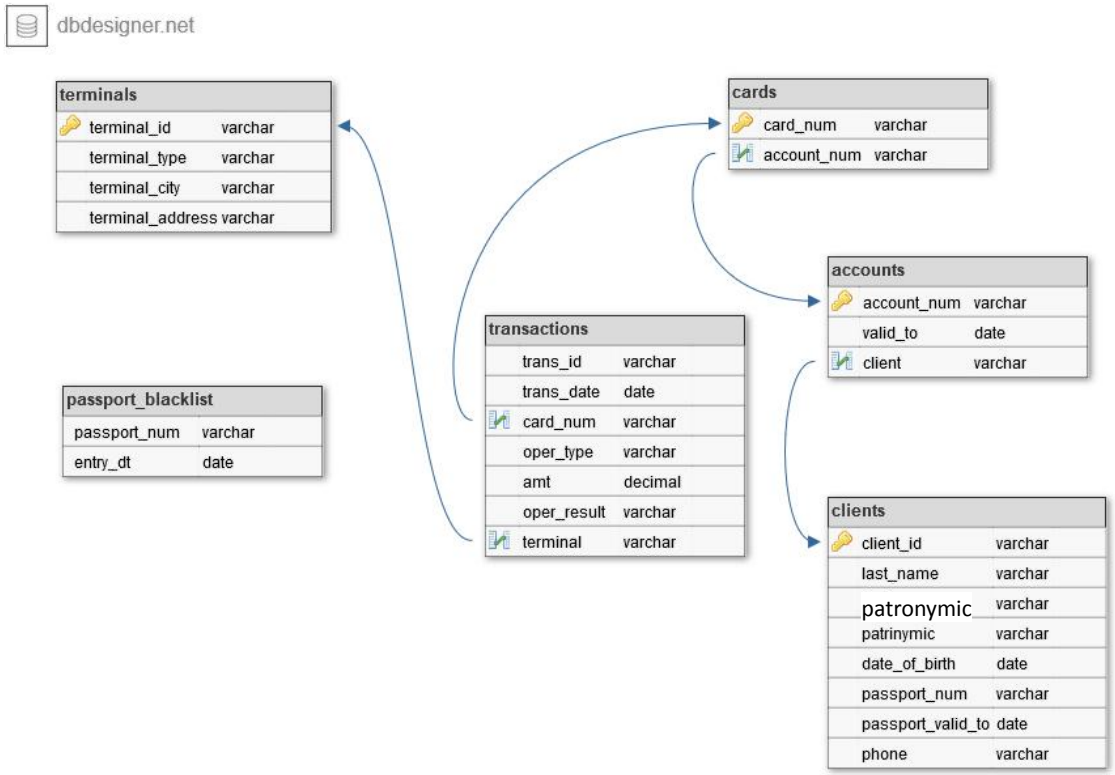
Сведения о картах, счетах и клиентах хранятся в СУБД PostgreSQL.

Вам предоставляется выгрузка за последние три дня, ее надо обработать.

Структура хранилища.

В качестве хранилища выступает ваша учебная база.

Данные должны быть загружены в хранилище со следующей структурой (имена сущностей указаны по существу, без особенностей правил нейминга, указанных далее):



Типы данных в полях можно изменять на однородные если для этого есть необходимость. Имена полей менять нельзя. Ко всем таблицам SCD1 должны быть добавлены технические поля create_dt, update_dt; ко всем таблицам SCD2 должны быть добавлены технические поля effective_from, effective_to, deleted_flg.

Построение отчета.

По результатам загрузки ежедневно необходимо строить витрину отчетности по мошенническим операциям. Витрина строится накоплением, каждый новый отчет укладывается в эту же таблицу с новым report_dt.

В витрине должны содержаться следующие поля:

event_dt	Время наступления события. Если событие наступило по результату нескольких действий – указывается время действия, по которому установлен факт мошенничества.
passport	Номер паспорта клиента, совершившего мошенническую операцию.
fio	ФИО клиента, совершившего мошенническую операцию.
phone	Номер телефона клиента, совершившего мошенническую операцию.

event_type	Описание типа мошенничества (номер).
report_dt	Дата, на которую построен отчет.

Признаки мошеннических операций.

1. Совершение операции при просроченном или заблокированном паспорте.

2. Совершение операции при недействующем договоре.

3. Совершение операций в разных городах в течение одного часа.

4. Попытка подбора суммы. В течение 20 минут проходит более 3х операций со следующим шаблоном – каждая последующая меньше предыдущей, при этом отклонены все кроме последней. Последняя операция (успешная) в такой цепочке считается мошеннической.

Правила именования таблиц.

Необходимо придерживаться следующих правил именования (для автоматизации проверки):

<CODE>_STG_<TABLE_NAME>	Таблицы для размещения стейджинговых таблиц (первоначальная загрузка), промежуточное выделение инкремента если требуется. Временные таблицы, если такие потребуются в расчете, можно также складывать с таким именованием. Имя таблиц можете выбирать произвольное, но смысловое.
<CODE>_DWH_FACT_<TABLE_NAME>	Таблицы фактов, загруженных в хранилище. В качестве фактов выступают сами транзакции и «черный список» паспортов. Имя таблиц – как в ER диаграмме.
<CODE>_DWH_DIM_<TABLE_NAME>	Таблицы измерений, хранящиеся в формате SCD1. Имя таблиц – как в ER диаграмме.
<CODE>_DWH_DIM_<TABLE_NAME>_HIST	Таблицы измерений, хранящиеся в SCD2 формате

	(только для тех, кто выполняет усложненное задание). Имя таблиц – как в ER диаграмме.
<CODE>_REP_FRAUD	Таблица с отчетом.
<CODE>_META_<TABLE_NAME>	Таблицы для хранения метаданных. Имя таблиц можете выбирать произвольное, но смысловое.

<CODE> - 4 буквы вашего персонального кода.

Обработка файлов

Выгружаемые файлы именуются согласно следующему шаблону:

transactions_DDMMYYYY.txt

passport_blacklist_DDMMYYYY.xlsx

terminals_DDMMYYYY.xlsx

Предполагается что в один день приходит по одному такому файлу. После загрузки соответствующего файла он должен быть переименован в файл с расширением .backup чтобы при следующем запуске файл не искался и перемещен в каталог archive:

transactions_DDMMYYYY.txt.backup

passport_blacklist_DDMMYYYY.xlsx.backup

terminals_DDMMYYYY.xlsx.backup

Желающие могут придумать, обосновать и реализовать более технологичные и учитывающие сбои способы обработки (за это будет повышен балл).

Проверка результата.

Проверка задания состоит из нескольких частей, обязательных к одновременному выполнению.

1. Загрузка в anytask.

В anytask выкладывается zip-архив, содержащий следующие файлы и каталоги:

main.py	Файл, обязательный		Основной процесс обработки.

файлы с данными	Файл, обязательный	Те файлы, которые вы получили в качестве задания. Просто скопируйте все 9 файлов.
main.ddl	Файл, обязательный	Файл с SQL кодом для создания всех необходимых объектов в базе.
main.cron	Файл, обязательный	Файл для постановки вашего процесса на расписание, в формате crontab
archive	Каталог, обязательный	Пустой, сюда должны перемещаться отработанные файлы
sql_scripts	Каталог, необязательный	Если вы включаете в main.py какие-то SQL скрипты, вынесенные в отдельные файлы – помещайте их сюда.
py_scripts	Каталог, необязательный	Если вы включаете в main.py какие-то python скрипты, вынесенные в отдельные файлы – помещайте их сюда.

Имя архива – 4 буквы вашего кода с расширением .zip. Например, SIND.zip.

2. Данные в таблицах на сервере.

Данные в ваших таблицах должны быть загружены за все три дня. Данные в таблицах будут проверены автоматически исходя из правил наименования. Будьте внимательны, если имя таблицы не соответствует выставленным требованиям – проверка не происходит, считается что вы не отловили ни один из случаев.

3. Код в вашем гит репозитарии.

На гите должны быть

выложены точно те же файлы и каталоги, которые вы прислали на проверку в anytask.

Критерии оценки.

Проект будет оцениваться экспертной оценкой. Оценка выставляется аргументировано и может обсуждаться, но не изменяться. После объявления оценки, если не прошел контрольный срок, можно доработать индивидуальное задание и сдать его на повторную проверку.

У преподавателя есть право добавить дополнительные баллы за сложные решения в проекте (не сложное решение простой задачи, а именно решение сложной задачи).

Минимальные требования для проекта (макс 4 балла)

1. Данные загружены в таблицу фактов (transactions).
2. Создана и заполнена хотя бы одна таблица измерений (terminals, blacklist, accounts, cards, clients)
3. Структурированный код: отступы, табуляции, комментирование, разделение на отдельные файлы логических блоков.
4. Форма SCD 1.
5. Выделен хотя бы один тип мошеннических транзакций в отчете

Проект на максимальный балл (10 баллов)

1. Таблицы приведены к форме SCD 2.
2. Выявлены все типы мошеннических транзакций.
3. Грамотно настроен ETL обработки данных. Процесс полностью автоматизирован.

Доп. Баллы:

+1 за внедрение airflow в проект

+1 балл за существенные улучшения и интересные решения (например, за обработку ошибок в данных) на усмотрение преподавателя

Баллы от 4 до 10 будут ставится в зависимости от **качества** решения. Качество будет определяться экспертно.

К оцениванию проекта невозможно применить некую объективную шкалу оценки (например, 50 строк кода это лучше чем 20 строк кода, или пять таблиц в отчете лучше чем три). Поэтому проект будет оцениваться экспертной оценкой по пяти показателям. В качестве эксперта выступает преподаватель. Оценка выставляется аргументировано и может обсуждаться, но не изменяться. После объявления оценки, если не прошел контрольный срок, можно доработать индивидуальное задание и сдать его на повторную проверку.

У преподавателя есть право добавить дополнительные баллы за сложные решения в проекте (не сложное решение простой задачи, а именно решение сложной задачи).

Критерии выставления оценки:

1. Структурированность кода – восприятие кода (отступы, табуляции), комментирование, разделение на отдельные файлы логических блоков. **До 10%.**

2. Качество обработки инкремента. Инкремент должен выделяться правильно, максимально эффективно и без лишних операций, контроль проводится в том числе автоматически по нескольким операциям. **До 15%.**

3. Общая сложность процесса обработки данных. При выполнении задания необходимо придерживаться стандартов, изученных в курсе. Необоснованное ухудшение процесса обработки будет снижать балл. Приветствуется использование изученных алгоритмов загрузки данных в хранилище,
использование метаданных. **До 40%, причем если вы используете только SCD1 – то до 15%.**

4. Качество получаемого результата. Необходимо найти все типы мошеннических операций (см. **Признаки мошеннических операций)**. За каждый выявленный тип операций до 9%. Итого **до 35%.**

5. Дополнительные баллы за сложность. Проверяющий оставляет за собой право добавлять **до 25%** дополнительных баллов за дополнительное полезное улучшение (и усложнение) проекта.

Минимальные требования, для того чтобы мы считали проект успешно выполненным — успешная загрузка одной фактовой таблицы и одной таблицы измерений, отлов хотя бы одного случая мошенничества в отчете и минимальный балл за все задание 35%.