

Домашняя работа 1.

1. Краткая инструкция:

- а. подготовительные действия для работы с данными (где и в каком виде хранить данные, какой софт установить и т.д.) - установить необходимо PostgreSQL, данные брались из таблиц, чей формат был изменен на csv. Также в колонке с названием date был изменен формат с числового на "дата".
- б. краткое содержания процедур, заложенных в скриптах/запросах, а также порядок их запуска;
- в. описание ожидаемых результатов каждого этапа.

2. Скрипты с запросами в формате sql

1 задание: Запросами SQL загрузить в БД данные из файлов ratings_task.xlsx, credit_events_task.xls и scale_EXP_task.csv. Выбрать форматы полей, создать таблицы (наложить разумные ограничения на поля), импортировать данные. (1 балл)

1 таблица

#Сначала удаляем предыдущую таблицу, которая получалась из-за работы над кодом в процессе его написания. Затем создаем новую.

```
DROP TABLE if exists public.actions
```

```
;
```

```
CREATE TABLE public.actions
```

```
(
```

#Устанавливаем первичный ключ, это понадобится в следующих заданиях

```
Rating_record_id serial primary key,
```

#выбираем форматы полей, соответствующие содержанию и назначению данных, также ставим ограничение по размеру. Если необходимо пишем, что в ячейке не может отсутствовать значение (NOT NULL).

```
rat_id varchar(25) NOT NULL,
```

```
grade varchar(25) NOT NULL,
```

```
outlook varchar(25),
```

```
change varchar(25) NOT NULL,
```

```
"date" char(15) NOT NULL,
```

```
ent_name varchar(200) NOT NULL,
```

```
okpo varchar(10) NOT NULL,
```

```
ogrn varchar(15),
```

```
inn varchar(10),
```

```
finst varchar(10),
```

```
agency_id varchar(10) NOT NULL,
```

```
rat_industry varchar(20),  
rat_type varchar(10) NOT NULL,  
horizon varchar(10),  
scale_typer varchar(50),  
currency varchar(10),  
backed_flag varchar(30)  
)
```

```
WITH (
```

```
OIDS = FALSE
```

```
)
```

```
TABLESPACE pg_default;
```

```
ALTER TABLE public.actions
```

```
OWNER to postgres;
```

#устанавливаем разумные ограничения на поля, чтобы при использовании неподходящего по логике таблицы значения данные не вводились, данные ограничения взяты из описания данных в предложенных для работы файлах.

```
ALTER TABLE public.actions ADD CONSTRAINT agency_id_con CHECK (agency_id='AKM' OR agency_id  
='EXP' OR agency_id='FCH' OR agency_id='MDS' OR agency_id='NRA' OR agency_id='RUS' OR  
agency_id='SNP');
```

```
ALTER TABLE public.actions ADD CONSTRAINT horizon_con CHECK (horizon='LT' OR horizon='ShT');
```

```
ALTER TABLE public.actions ADD CONSTRAINT scale_typer_con CHECK (scale_typer='lsc' OR position('sc'  
in scale_typer) = char_length(scale_typer) - 1);
```

```
ALTER TABLE public.actions ADD CONSTRAINT currency_con CHECK (currency='lcur' OR  
currency='Ncur');
```

#Импортируем данные в таблицу, учитывая правильный путь файла, его формат и необходимость в первой строке выводить данные, а заголовок помещать отдельно primary key)

```
copy public.actions (rat_id, grade, outlook, change, "date", ent_name, okpo, ogrn, inn, finst, agency_id,  
rat_industry, rat_type, horizon, scale_typer, currency, backed_flag) FROM  
E'C:\\Users\\Public\\Data\\Data\\Risks\\ratings_task.csv' (format csv, delimiter ';', header True)
```

Ожидаемые результаты: ожидается получить описанную выше таблицу с 38252 значениями.

2 таблица

#Описание процесса аналогично пункту «а».

```
DROP TABLE if exists public.credit_events_task
```

```

;
CREATE TABLE public.credit_events_task
(
    "inn" varchar(10) NOT NULL,
    "date" date NOT NULL,
    event varchar(3) NOT NULL
)
WITH (
    OIDS = FALSE
)
TABLESPACE pg_default;
ALTER TABLE public.credit_events_task
    OWNER to postgres;
ALTER TABLE public.credit_events_task ADD CONSTRAINT event_con CHECK (event='dft' OR
event='lqd');
copy public.credit_events_task FROM
'C:\\Users\\Public\\Data\\Data\\Risks\\credit_events_task.csv'(format csv, delimiter ';', header True)

```

Ожидаемые результаты: ожидается получить описанную выше таблицу с 188 значениями.

3 таблица

#Описание процесса аналогично пункту «а», однако в конце появляется следующий код: encoding 'WIN 1251' – он помогает работать с кириллицей в ячейках таблицы.

```

DROP TABLE if exists public.scale_EXP_task
;
CREATE TABLE public.scale_EXP_task
(
    grade varchar(15) NOT NULL UNIQUE,
    grade_id integer NOT NULL PRIMARY KEY
)
WITH (
    OIDS = FALSE
)
TABLESPACE pg_default;
ALTER TABLE public.scale_EXP_task

```

OWNER to postgres;

```
copy public.scale_EXP_task FROM 'C:\\Users\\Public\\Data\\Data\\Risks\\scale_EXP_task.csv'(format
csv, delimiter ';', header True,encoding 'WIN 1251')
```

Ожидаемые результаты: ожидается получить описанную выше таблицу с 13 значениями.

2 номер: Запросами SQL вынести из таблицы actions информацию о рейтингах (тип, срочность и т.д.) и информацию о рейтингуемом лице (текстовое название и лишние числовые коды) в отдельные таблицы. (1 балл)

#Удаляем данные от предыдущей версии в процессе работы и создаем новую таблицу

```
DROP TABLE if exists rat_info
```

```
;
```

```
CREATE TABLE rat_info AS
```

Выбираем из таблицы actions информацию о рейтингах и выносим ее в отдельную таблицу, группируя записи по набору параметров

```
SELECT distinct agency_id, rat_industry, rat_type, horizon, scale_typer, currency,
backed_flag,rating_record_id, rat_id
```

```
FROM public.actions GROUP BY agency_id, rat_industry, rat_type, horizon, scale_typer, currency,
backed_flag,rating_record_id, rat_id;
```

#устанавливаем внешний ключ для дальнейшей работы

```
ALTER TABLE rat_info ADD PRIMARY KEY (rating_record_id)
```

#Удаляем данные от предыдущей версии в процессе работы и создаем новую таблицу

```
DROP TABLE if exists info_comp
```

```
;
```

```
CREATE TABLE info_comp AS
```

Выбираем из таблицы ratings_task информацию о компаниях и выносим ее в отдельную таблицу, группируя записи по набору параметров

```
SELECT ent_name, okpo, ogrn, inn, finst
```

```
FROM ratings_task GROUP BY ent_name, okpo, ogrn, inn, finst;
```

#устанавливаем внешний ключ для дальнейшей работы

```
ALTER TABLE info_comp ADD PRIMARY KEY (ent_name)
```

Ожидаемые результаты: ожидается получить описанные выше таблицы.

3 номер: Связать таблицы в базе, задав внешние ключи, где это целесообразно. (1 балл)

#Внешний ключ связывает поле подчинённой таблицы с уникальным полем главной таблицы
По тому, как получены таблицы info_comp и rat_info, очевидно, что из таблицы actions можно удалить соответствующие столбцы, оставив ссылки на новые таблицы. Для этих ссылок нужны внешние ключи, которые соединяют содержащиеся в actions поля ent_name и rating_record_id с одноимёнными первичными ключами таблиц. Логично было бы, если бы первичным ключом rat_info был rat_id, но он не обладает уникальностью. Не удалось подобрать набор полей, который не содержал бы искусственного ключа rating_record_id и был бы уникальным для данной таблицы. Попытка создания внешнего ключа на поле grade, ссылающегося на таблицу scale_EXP_task, не работает, поскольку значения grade в таблице actions принимают не только значения, имеющие id и находящиеся в scale_EXP_task, но и многие другие. Таблица credit_events, содержащая inn в роли идентификатора фирм, не может быть соединена с actions или info_comp по внешнему ключу inn, потому что даже в info_comp поле inn не уникально.

```
ALTER TABLE actions ADD CONSTRAINT entfk FOREIGN KEY (ent_name) REFERENCES info_comp (ent_name);
```

```
ALTER TABLE actions ADD CONSTRAINT ratfk FOREIGN KEY (rating_record_id) REFERENCES rat_info (rating_record_id);
```

```
ALTER TABLE actions ADD CONSTRAINT gradefk FOREIGN KEY (grade) REFERENCES scale_EXP_task (grade)
```

4 номер: В сформированной базе данных составьте запрос, который выводит для выбранных вида рейтинга (rat_id) и даты (date) в таблице actions, все актуальные рейтинги. Актуальным рейтингом признается значение, присвоенное или подтвержденное последним рейтинговым действием до исследуемой даты, при условии, что это действие было не снятием и не приостановкой рейтинга. Таблица должна содержать поля ent_name, grade и assign_date (дату присвоения актуального рейтинга). (2 балла)

#Делаем вложенный запрос, который для каждой пары значений rat_id и date находит требуемые последние рейтинги. И если они актуальны (не закрыты), то присоединяет их к выборке.

```
SELECT distinct ent_name, grade, assign_date  
  
FROM actions, (SELECT actions2.rating_record_id, actions2.ent_name, actions2.grade, actions2."date"  
as assign_date  
  
from actions actions2  
  
WHERE actions2."date" =  
  
(SELECT max("date")  
  
from actions actions2  
  
WHERE actions2.rat_id = actions.rat_id  
  
AND actions.date > actions2."date"))  
  
as f  
  
WHERE actions.rating_record_id = f.rating_record_id  
  
AND change != 'снят'  
  
AND change != 'приостановлен'
```