**Задание 2**

Изучите данные и выполните следующие задачи. Выполните задание с помощью SQL.

*Решение оформите в одном Word-файл с текстом запросов на каждое задание.*

Есть 3 таблицы с данными. В таблице client\_info хранится информация о клиентах, в частности, пол, возраст и тд. В таблице payments хранится информация о всех клиентских платежах.

В таблице client\_geo хранится расшифровка идентификаторов городов и наименования федеральных округов, в которых расположены клиентские города.

**Таблица payments**

* id\_client (уникальный идентификатор клиента)
* time\_payment (дата и время платежа в формате "гггг-мм-дд чч:мм:сс")
* amt\_payment (размер платежа)

**Таблица client\_info**

* id\_client (уникальный идентификатор клиента)
* gender (пол клиента)
* age (возраст клиента)
* id\_city (идентификатор города клиента)

**Таблица city\_info**

* id\_city (идентификатор города клиента)
* name\_city (название города клиента)
* name\_region (наименование федерального округа, в котором расположен данный город)

### **Задание 2.1**

*Для федерального округа "Поволжье" выведите динамику суммарных платежей по дням.*

Для начала создали с помощью Excel три таблицы с необходимыми столбцами (см выше), сохранили их в csv файлы, загрузили их в <https://sqliteonline.com/>

Для работы с данными в формате дата или дата время использовали данные функции

--to\_date(time\_payment\_, 'dd.mm.yyyy') для приведения excel даты в sql дату

--to\_timestamp(time\_payment\_, 'dd.mm.yy hh24:mi') для даты со временем

**select to\_timestamp(time\_payment\_, 'dd.mm.yy'),**

**sum(amt\_payment\_) as cnt\_payment**

**from payments as pp**

**left join client\_info as ci**

**on pp.id\_client\_=ci.id\_client\_**

**left join city\_info as cd**

**on ci.id\_city\_=cd.id\_city\_**

**where name\_region\_='Povolzhie'**

**group by to\_timestamp(time\_payment\_, 'dd.mm.yy')**

**order by to\_timestamp(time\_payment\_, 'dd.mm.yy')**

### **Задание 2.2**

*Для каждого города найдите долю мужчин (% мужчин среди всех клиентов в данном городе). Ограничьтесь только клиентами, которым от 20 до 40 лет. В выводе используйте названия городов, а не идентификаторы.*

При помощи оконных функций нашли общее число клиентов от 20 до 40 лет всего, потом по всем городам, затем город и пол.

Далее нашли процент клиентов-мужчин в рамках каждого города (поставили условие на мужчин и поделили мужчин в каждом городе на количество мужчин+женщин в этом городе).

**select name\_city\_,**

**count(\*) as cnt\_all,**

**sum(case when gender='M' tHEN 1 else 0 end) as cnt\_men,**

**(sum(case when gender='M' tHEN 1 else 0 end)::float/count(\*)::float)\*100 as share\_men**

**from client\_info as ci**

**left join city\_info as cd**

**on ci.id\_city\_=cd.id\_city\_**

**group by name\_city\_**

**order by name\_city\_**

### **Задание 2.3**

*Определите средний возраст по тем клиентам, которые ни разу ничего не заплатили.*

К таблице с данными клиентов добавляем таблицу с данными о платежах, оставляем только клиентов без единого платежа. Находим средний возраст.

**select**

**avg(age) as avg\_age\_no\_purchase**

**from**

**(select \***

**from client\_info as ci**

**left join payments as pp**

**on ci.id\_client\_=pp.id\_client\_**

**where amt\_payment\_ is NULL) tab**

### **Задание 2.4**

*Для каждого федерального округа выделите первые три платежа.*

Объединяем три таблицы в одну, применяем оконную функцию row\_number и сортируем по возрастанию даты платежей. В результате напротив каждой даты получаем ранг по возрастанию. Группировка по регионам. Ставим фильтр: нам нужны только ранги 1, 2, 3 ( то есть первые три платежа, самые ранние). Итог по каждому региону первые три даты покупки.

**select name\_region\_,**

**to\_timestamp,**

**rn\_pay\_region**

**from**

**(select name\_region\_,**

**to\_timestamp(time\_payment\_, 'dd.mm.yy hh24:mi'),**

**ROW\_NUMBER() over(PARTITION by name\_region\_ order by (to\_timestamp(time\_payment\_, 'dd.mm.yy hh24:mi'))) as rn\_pay\_region**

**from payments as pp**

**left join client\_info as ci**

**on pp.id\_client\_=ci.id\_client\_**

**left join city\_info as cd**

**on ci.id\_city\_=cd.id\_city\_) tab**

**WHERE rn\_pay\_region in(1, 2, 3)**

### **Задание 2.5**

*Ограничьтесь клиентами из федеральных округов "Южный" и "Северный". Для каждого города рассчитайте, сколько в среднем времени проходит между платежами одного клиента.*

Объединили три таблицы. Оставили только два региона: "Южный" и "Северный”. Используя функцию lead создали еще один столбец со следующими датой и временем покупки в рамках одного клиента и каждого города. Далее преобразовали данный столбец в разницу времени следующей покупки и предыдущей также в рамках одного клиента и города (нашли дельту) . Затем с помощью оконной функции avg нашли среднее время между покупками в пределах одного города и клиента.

**select name\_city\_,**

**id\_client\_,**

**avg(delta\_pay\_time) over(PARTITION by id\_client\_) as avg\_time\_between\_purchese**

**from**

**(select cd.name\_city\_,**

**to\_timestamp(time\_payment\_, 'dd.mm.yy hh24:mi'),**

**ci.id\_client\_,**

**lead(to\_timestamp(time\_payment\_, 'dd.mm.yy hh24:mi')) over(PARTITION by cd.name\_city\_, ci.id\_client\_ order by (to\_timestamp(time\_payment\_, 'dd.mm.yy hh24:mi'))) - (to\_timestamp(time\_payment\_, 'dd.mm.yy hh24:mi')) as delta\_pay\_time**

**from payments as pp**

**left join client\_info as ci**

**on pp.id\_client\_=ci.id\_client\_**

**left join city\_info as cd**

**on ci.id\_city\_=cd.id\_city\_**

**WHERE name\_region\_ in('Ug', 'Sever') ) tab**

**order by name\_city\_, id\_client\_**