

## Задача 1.

### 1. Какие проблемы решает использование транзаций?

Одно атомарное с точки зрения бизнес логики действие, например перевод денег от одного пользователя к другому, может требовать проведения операций над несколькими таблицами. В моем понимании основное предназначение транзакций – это решение данной проблемы. Благодаря транзакциям мы можем либо выполнить операцию, состоящую из нескольких шагов, полностью, либо отменить ее целиком.

### 2. К каким последствиям может привести отсутствие транзакций в используемом хранилище данных? Приведите примеры.

Это может привести к следующим последствиям:

1. Риск перехода базы в "несогласованное состояние". Пример: У нас есть операция по переводу денег между двумя пользователями. Допустим, мы это делаем в такой последовательности :

```
update account_balances set balance += 100.34 where account_id = 2;  
update account_balances set balance -= 100.34 where account_id = 1;  
insert into transaction_log(from_id, to_id, amount) values (1, 2, 100.34);
```

Если на account\_balances у нас повешен constraint наподобии: "alter table account\_balances add constraint balance\_should\_be\_above\_zero check(balance > 0)", а у пользователя 1 денег меньше чем 100.34, то на втором запросе произойдет ошибка. В итоге баланс пользователя 2 увеличится, а баланс пользователя 1 не изменится.

2. Риск чтения противоречивых данных. Пример: Допустим у нас есть какой-нибудь достаточно долгий отчет для бухгалтерии, в котором мы считаем все проведенные транзакции плюсуем с текущими балансами пользователей и хотим в графе "итого" видеть 0. Если в момент выборки данных баланс пользователя 2 из предыдущего примера уже изменился, а пользователя один еще нет, то в результате запроса мы получим некорректные данные.

### 3. С какими трудностями можно столкнуться, эксплуатируя транзакционную СУБД? Приведите примеры.

Лично я сталкивался с такими проблемами как взаимные блокировки и снижение производительности.

Взаимные блокировки возникают в случае если в разных транзакциях нам нужны блокировки на одни и те же объекты БД и одна транзакция уже получила блокировку на один объект, а другая на другой. Пример: две операции по автоматической смене состояния заявок выполняются по крону, одна каждые сутки в 3 часа ночи, вторая в 3 часа ночи в воскресенье. В 3 часа ночи в воскресенье они запускаются одновременно взаимно блокируют друг друга и никакого обновления не происходит.

Снижение производительности я наблюдал в двух вариантах: значительное увеличение времени выполнения массивных update/insert запросов внутри одной транзакции, а также торможение прочих запросов, иногда даже на select пока не пройдет commit большого update/insert запроса. Пример: Плохо подумав запускаем на продакшен базе в рабочее время запрос к таблице "миллионику":

```
start transaction; update tickets set cost += cost * 0.05, status_id = 5432 where executor_id = 12345 and not is_complete; commit;
```

В момент когда у нас проходит выполнение данного запроса у всех операторов запросы на смену статуса для таких заявок встану в очередь, что будет сильно мешать их работе.