Пример к Coordination для транзакции. Например, если из таблицы «Отделы» удаляется отдел, то СУБД должна запретить это действие, если в таблице «Сотрудники» имеются записи сотрудников удаляемого отдела

Пояснение к Isolation для Транзакций. Изоляция означает, что несмотря на параллельный доступ к базе данных множества транзакций, каждая из них должна проходить изолированно от остальных. Свойство изоляции транзакций гарантирует, что транзакция не увидит изменений, внесенных другими транзакциями в БД, пока она выполняется.

**1 эффект: "Потерянная запись"**



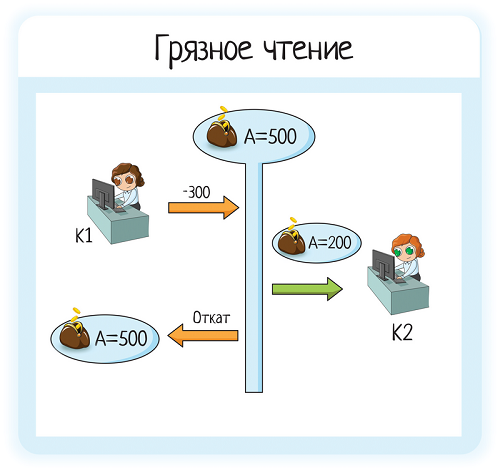
Есть некий счет А, на котором лежит 500 у.е.

Кассир 1 (К1 на рисунке) списал с него 300 у.е. Обозначим его действия рыжими стрелками. Списал 300, на выходе получает 200 = 500 - 300.

Кассир 2 (К2) тоже решил обратиться к этому же счету, и записал туда 300 у.е., пока К1 еще не успел закрыть свою транзакцию. Так как первая транзакция не закрыта, сумма на счете до сих пор 500, получаем 500 + 300 = 800.

Итог — мы "потеряли запись" первого кассира, ведь на выходе у нас А = 800, хотя должно быть 500. "Кто последний вписал результат - того и тапки". Получается так.

**2 эффект: "Грязное чтение"**



Есть некий счет А, на котором лежит 500 у.е.

Кассир 1 списал с него 300 у.е. Обозначим его действия рыжими стрелками. Списал 300. Потом передумал и сделал откат - на выходе остались те же 500 у.е.

Кассиру 2 (К2) понадобилась информация по этому счету и он ее считал до того, как К1 закрыл свою транзакцию.

Итог — второй кассир считал неверную сумму, построил неверный отчет/отказал в визе платежеспособному гражданину и т.д.

**3 эффект: "Повторимое чтение"**



Есть некие данные.

Кассир 1 строит отчет. Операции идут последовательно для каждой колонки. Система считала данные, записала в первую колонку (например, взяв минимум от них).

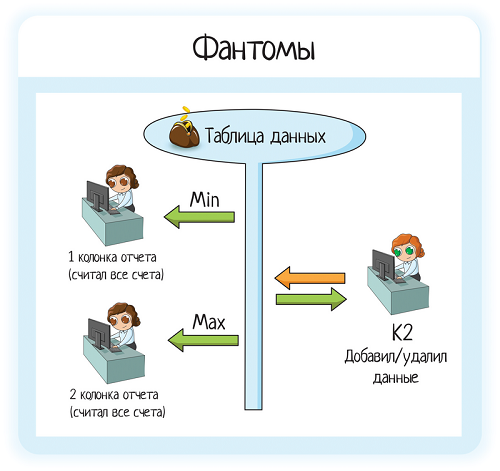
Обозначим получение данных зеленым цветом, а изменение - рыжим.

Кассир 2 влез в эту таблицу данных и изменил некоторые счета в ней.

У кассира 1 продолжается построение отчета. И во вторую колонку система считывает уже новые данные.

Итог - отчет построен на основании разных данных.

**4 эффект: "Фантомы"**



Есть некие данные.

Кассир 1 строит отчет. Операции идут последовательно для каждой колонки. Система считала данные, записала в первую колонку (например, взяв минимум от них).

Обозначим получение данных зеленым цветом, а изменение - рыжим.

Кассир 2 влез в эту таблицу данных и добавил новые счета/удалил некоторые старые.

У кассира 1 продолжается построение отчета. И во вторую колонку система считывает уже новые данные.

Итог — отчет построен на основании разных данных.

Разница между 3-им и 4-ым эффектами в том, что в одном случае данные *изменяются*, а во втором — *добавляются/удаляются*. То есть меняется ещё и их количество.

Пример клинча:

У нас есть транзакция 1, которая работает с ресурсом 1

И транзакция 2, которая работает с ресурсом 2

В какой-то момент транзакции 1 перед завершением понадобилось обратиться к ресурсу 2

А транзакции 2 к ресурсу 1. И в итоге обе транзакции не могут завершиться