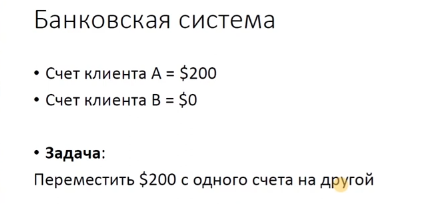
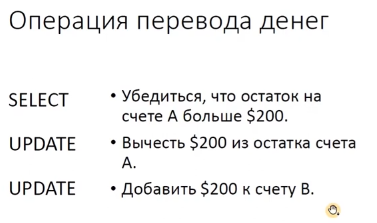
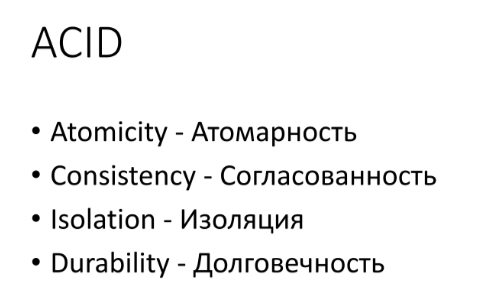
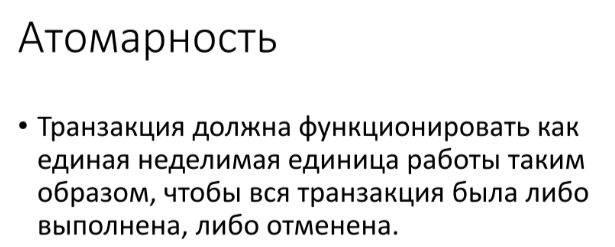
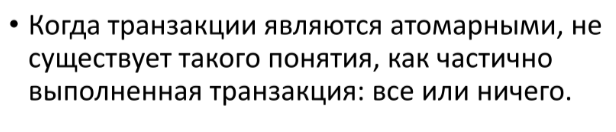
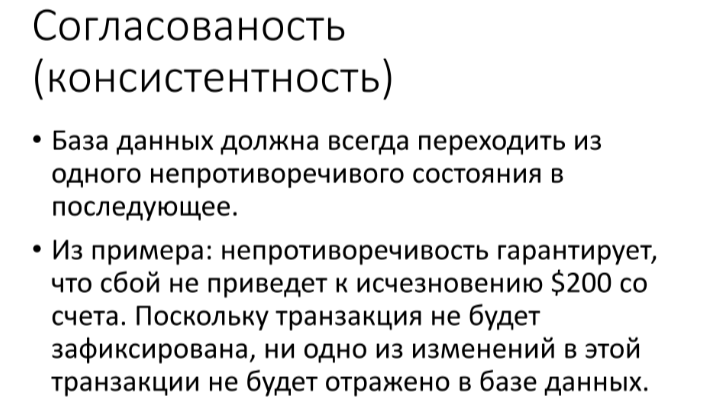
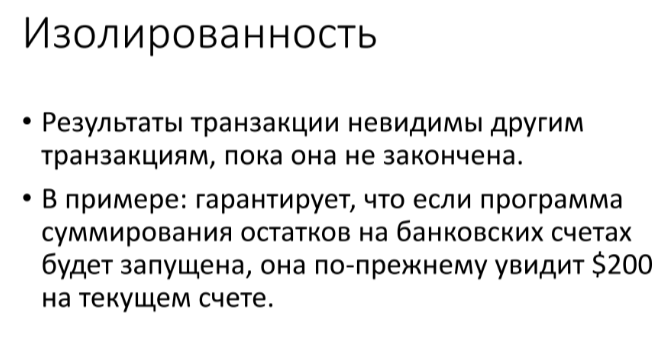
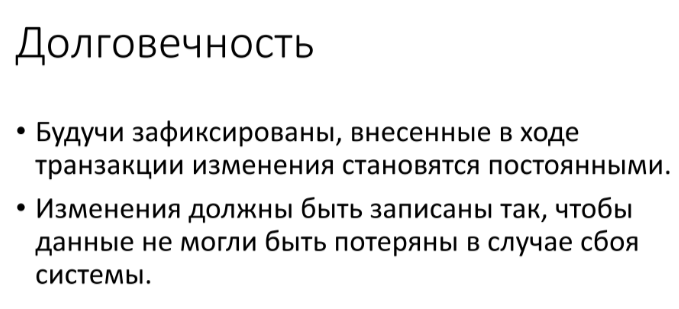
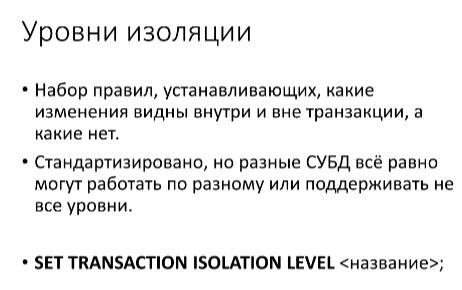
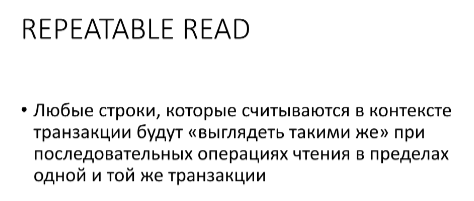
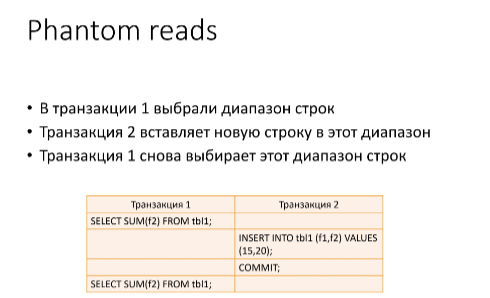
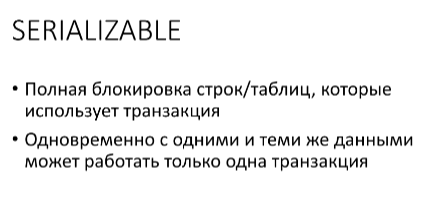
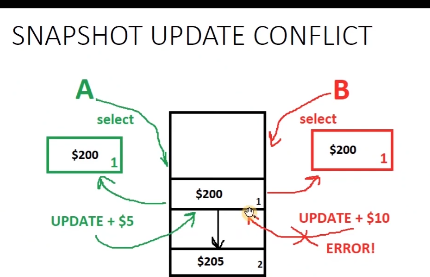
  
  
Для чего может это потребоваться?  
Пример с банковской схемой.  
  
  
Выполнить эту операцию одним запросом не получится.  
  
  
  
Почему нам важно, чтобы эти 3 операции были как единое целое?  
Потому что при ошибке операция будет незакончена и это недопустимо.  
  
Start Transaction, Commit.  
  
Rollback – откат.  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
Пессимистическая работает на блокировке данных.  
Оптимистическая предполагает что мы одни работаем с данными.  
  
  
  
Проблема uncommitted – мы можем читать неподтвержденные данные.  
  
  
  
Проблема – возможно невоспроизводимое чтение.  
Один запрос – разные данные.  
  
  
  
Проблема – фантомное чтение.  
  
  


Оптимистическая стратегия.  
Располагает предположением, что мы одни.  
  
Snapshot.  
Если клиент обратился к БД, то создается копия.   
  
Read commited snapshot.  
Каждый запрос новая копия.  
  
SNAPSHOT UPDATE.  


  
ы  
  
  
ы  
  
  
ы  
  
  
ы