

1.演示脏读

一个事物里面读到了另外一个事物没有提交的数据: read uncommitted

- 1.开启A, B窗口
- 2.分别查询A, B的隔离级别

```
select @@tx_isolation;
```

- 3.设置A窗口的隔离级别为read uncommitted(读未提交)

```
set session transaction isolation level read uncommitted;
```

- 4.A, B都开启事物
- 5.在B中zs向ls转账100, 事物不提交
- 6.在A中查询账户

2.演示不可重复读(解决脏读)

不可重复读: 在一个事物里面,同一条语句,两次查询的结果不一致.

- 1.开启A, B窗口
- 2.分别查询A,B的隔离级别

```
select @@tx_isolation;
```

- 3.设置A窗口的隔离级别为Read committed(读已提交)

```
set session transaction isolation level Read committed;
```

- 4.A, B都开启事物
- 5.在B中张三向李四转账100, 事物不提交
- 6.在A中查询账户 (避免脏读发生)
- 7.B中提交事物
- 8.在A中查询账户 (两次查询的结果不一致, 不可重复读发生)

3.演示避免不可重复读

- 1.开启A, B窗口
- 2.分别查询A,B的隔离级别

```
select @@tx_isolation
```

- 3.设置A窗口的隔离级别为Repeatable read

```
set session transaction isolation level Repeatable read;
```

- 4.A, B都开启事物

- 5.在B中张三向李四转账100，事物不提交
- 6.A中查询账户
- 7.B中提交
- 8.A中查询账户
- 9.A中结束事物，再重新查询

4.演示隔离级别Serializable

- 1.开启A, B窗口
- 2.分别查询A,B的隔离级别

```
select @@tx_isolation
```

- 3.设置A窗口的隔离级别为Serializable

```
set session transaction isolation level Serializable;
```

- 4.A, B都开始事物
- 5.B中向account账户插入一条数据，不提交
- 6.A中查询
- 7.在B中结束事物
- 8.A中查询