CSCE-310 Database Systems
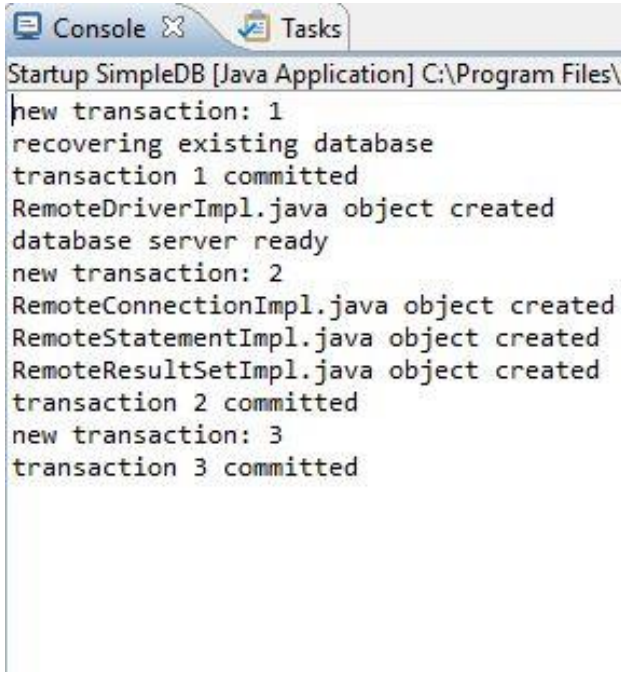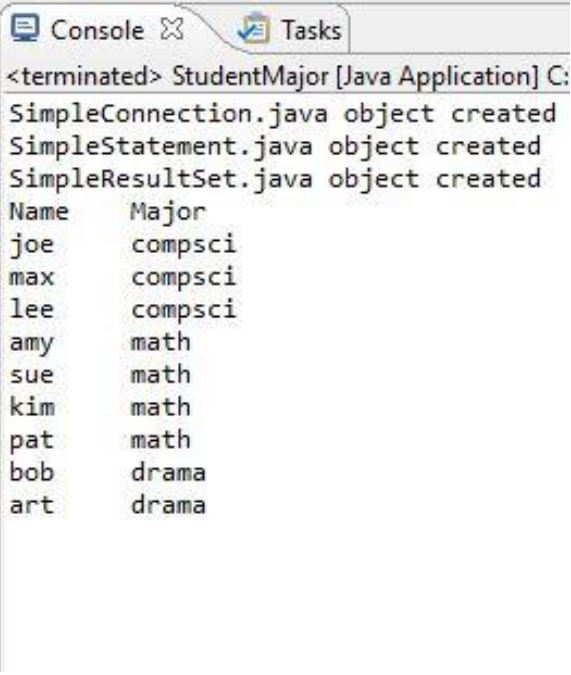Homework 10
Raymond Zhu
923008555

Problem 1:

Seven objects were observed
1. RemoteDriverImpl.java
2. RemoteConnectionImpl.java
3. RemoteStatementImpl.java
4. RemoteResultSetImpl.java
5. SimpleConnection.java
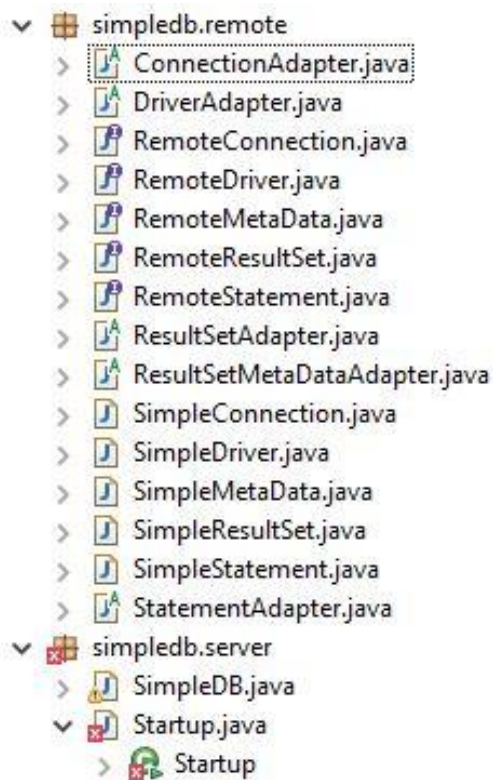6. SimpleStatement.java
7. SimpleResultSet.java

Problem 2:

Server	Client

```
Console 23    Tasks
Startup SimpleDB [Java Application] C:\Program Files\
new transaction: 1
recovering existing database
transaction 1 committed
RemoteDriverImpl.java object created
database server ready
new transaction: 2
RemoteConnectionImpl.java object created
RemoteStatementImpl.java object created
RemoteResultSetImpl.java object created
transaction 2 committed
new transaction: 3
transaction 3 committed
```

```
Console 23    Tasks
<terminated> StudentMajor [Java Application] C:
SimpleConnection.java object created
SimpleStatement.java object created
SimpleResultSet.java object created
Name    Major
joe     compsci
max     compsci
lee     compsci
amy     math
sue     math
kim     math
pat     math
bob     drama
art     drama
```

Problem 3:

This is a bad idea because transaction tx is just an incremental value. So, whenever you send in a transaction as a parameter instead of having it done in the method, you could potentially pass wrong values that could potentially break the server. Additionally, certain operations might end up using the same transaction id creating problems for the server but also debugging log files.

For example, if multiple threads were requesting a transaction from the server, passing by parameter could possible put threads and transaction ids out of sync and violate the referential integrity of the database.

Problem 4:


```
∨ ⊞ simpledb.remote
    > J^A ConnectionAdapter.java
    > J^A DriverAdapter.java
    > J^P RemoteConnection.java
    > J^P RemoteDriver.java
    > J^P RemoteMetaData.java
    > J^P RemoteResultSet.java
    > J^P RemoteStatement.java
    > J^A ResultSetAdapter.java
    > J^A ResultSetMetaDataAdapter.java
    > J SimpleConnection.java
    > J SimpleDriver.java
    > J SimpleMetaData.java
    > J SimpleResultSet.java
    > J SimpleStatement.java
    > J^A StatementAdapter.java
∨ ⊞ simpledb.server
    > J SimpleDB.java
    ∨ J Startup.java
        > Startup
```

The implementation classes are not necessary for the client ("StudentMajor.java") to work, however, it is necessary to have the remote interfaces. This is because the interfaces declare the methods and fields that are required at run time when running the client. The implementation itself varies and so that's why it isn't necessary.

An Aggie does not lie, cheat, steal, or tolerate those who do.