You will create a java application to simulate simple bank transactions with the following requirements

As a user I can log into my bank application

As a user I can view my bank account balance

As a user I can withdraw money from my account

As a user I can deposit money into my account

As a user I can transfer money between my accounts

As a user I can view my past transactions

As a user I can log out of my bank application

As a user I can request a line of credit. \*

As general requirement users can have multiple accounts

As general requirement users’ transactions must be persisted between sessions. This includes application termination and restarts.

As general requirement you must consider at least 2 exceptional conditions where you application will throw 2 custom exceptions, but will continue execution after handling said exceptions.

\*The system will judge credit requests based on the user’s credit score based on the following criteria

1. Excellent – Good is automatic approval
2. Good – Poor banker review
3. Poor – Bad automatic decline

QC "hints" and quiz study guide

What is a relational DB

What is sql

How do we use sql on a DB

What are the sublanguages of SQL

What are some command from sublanguage "x"

What are the acid properties of a transactions

What does it mean for a transaction to be Atomic, Consistent, Isolated, Durable

Name the joins of sql

Describe "x" join

What are the differences between union and union all

What is a trigger

What is a sequence

What is a function

How do you query for a record

How do you insert, update, or delete

What piece of information is returned from an insert, update, or delete operation

What is a constraint violation

What is a cursor

How to create any one of the above

What is an index

What is referential integrity

What are the constraints you can put on a table column

What the interfaces of JDBC

What is a JDBC driver

Difference between Statement, PreparedStatement, CallableStatement

What is a ResultSet. How do you use it

Explain the DAO design pattern.