**BANKING SOFTWARE USING JAVA**

Submitted By

**Abhishek .Amble**

**2GI20EE001**

**Electrical and Electronics**

**KLS Gogte Institute of Technology, Belgaum**



**2023**

**ABSTARCT**

I Have created Banking Software using Java Programming Language , in which I have used JDBC which stands for Java Database Connectivity, which is a standard Java API for database-independent connectivity between the Java programming language and a wide range of databases. Along with JDBC I have used MYSQL database for storing and retrieving data .I have used CRUD(create , read , update, delete ) method in this software.

My Banking Software allows to perform various features such as :

1. Creating Account and Logging in
2. Transfer Amount
3. Deposit Amount
4. Withdraw Amount
5. View Balance
6. Change Password
7. Log Out

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| Sl.no | Content |
| 1. | Introduction |
| 2. | Methodology |
| 3. | Implementation Details |
| 4. | Conclusion |
| 5. | Refrences |

1. **INTRODUCTION**

**1)JDBC- Java Database Connectivity:**

JDBC stands for **J**ava **D**ata**b**ase **C**onnectivity, which is a standard Java API for database-independent connectivity between the Java programming language and a wide range of databases.

The JDBC library includes APIs for each of the tasks mentioned below that are commonly associated with database usage.

* Making a connection to a database.
* Creating SQL or MySQL statements.
* Executing SQL or MySQL queries in the database.
* Viewing & Modifying the resulting records.

Fundamentally, JDBC is a specification that provides a complete set of interfaces that allows for portable access to an underlying database. Java can be used to write different types of executables , such as −

* Java Applications
* Java Applets
* Java Servlets
* Java Server Pages (JSPs)
* Enterprise JavaBeans (EJBs).

**2)MYSQL –RDBMS(Relational DataBase Management System)**

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons −

* MySQL is released under an open-source license. So you have nothing to pay to use it.
* MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
* MySQL uses a standard form of the well-known SQL data language.
* MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
* MySQL works very quickly and works well even with large data sets.

1. **METHODOLOGY**

## Software Used:

* MySQL
* IntelliJ IDEA Community Edition 2022.3.3

In any Bank Transaction, there are several parties involved to process transaction like a merchant, bank, receiver, etc. so there are several numbers reasons that transaction may get failed, declined, so to handle a transaction in Java, there is a JDBC (Java Database Connectivity) which provides us an API to connect, execute, fetch data from any databases. It provides the language Java database connectivity standards. It is used to write programs required to access databases.

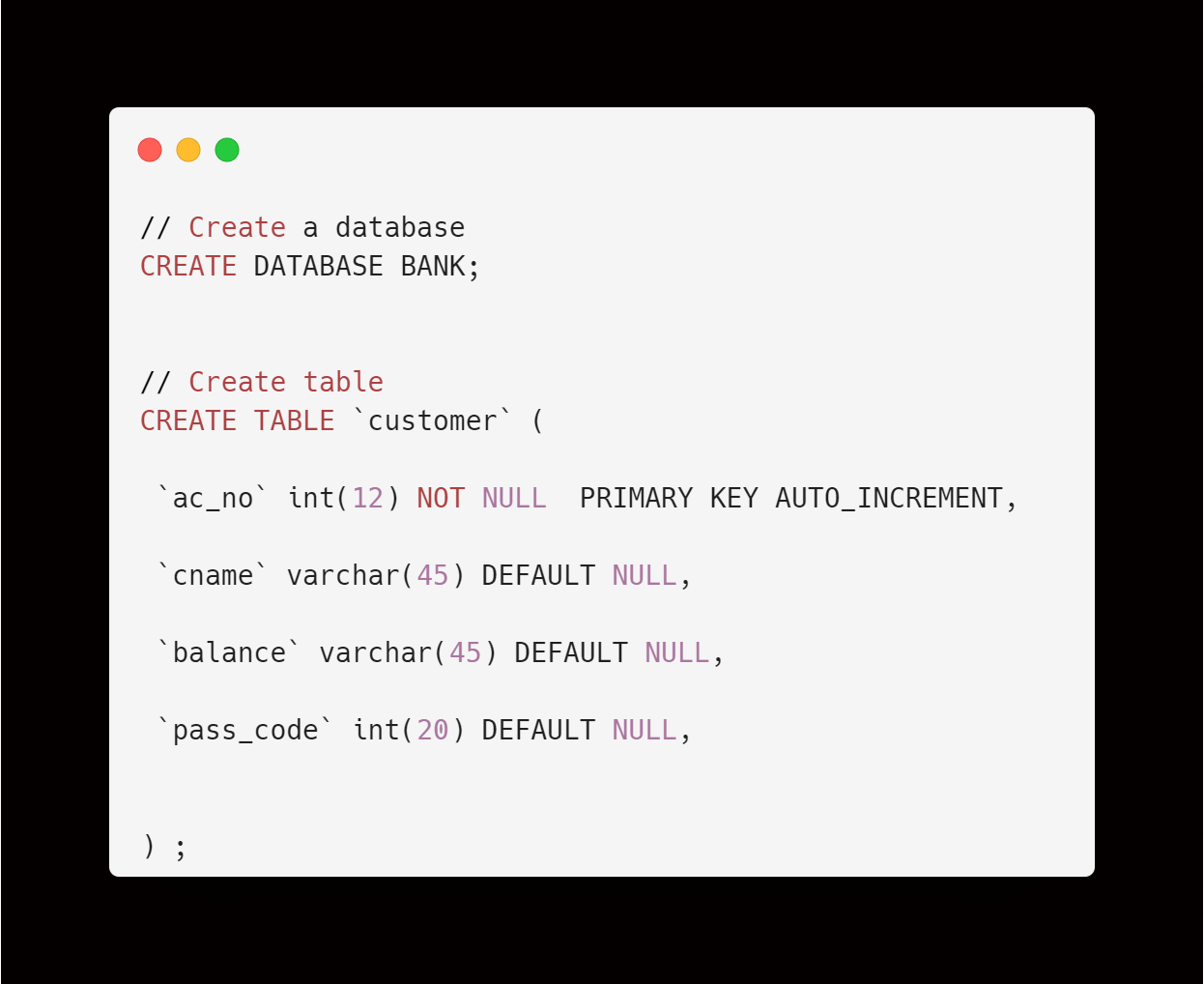
Transactions in JDBC provide us a  feature that considers a complete SQL statement as one unit,  then executes once, and if any statement fails, the entire transaction fails. To use transaction, we have to set **setAutoCommit(false);** manually, and once all the statements are executed successfully, making changes in the database’s **commit()**method will be required.

In this Mini Banking Application, to handle a transaction, we are using JDBC Transaction to make transactions consistent.  This Application Provides Menu-Driven Console Interface to a User Using that User can perform functions like create Account, Login, View Balance And Transfer Money To The Other Customer.

1. **IMPLEMENTATION DETAILS**

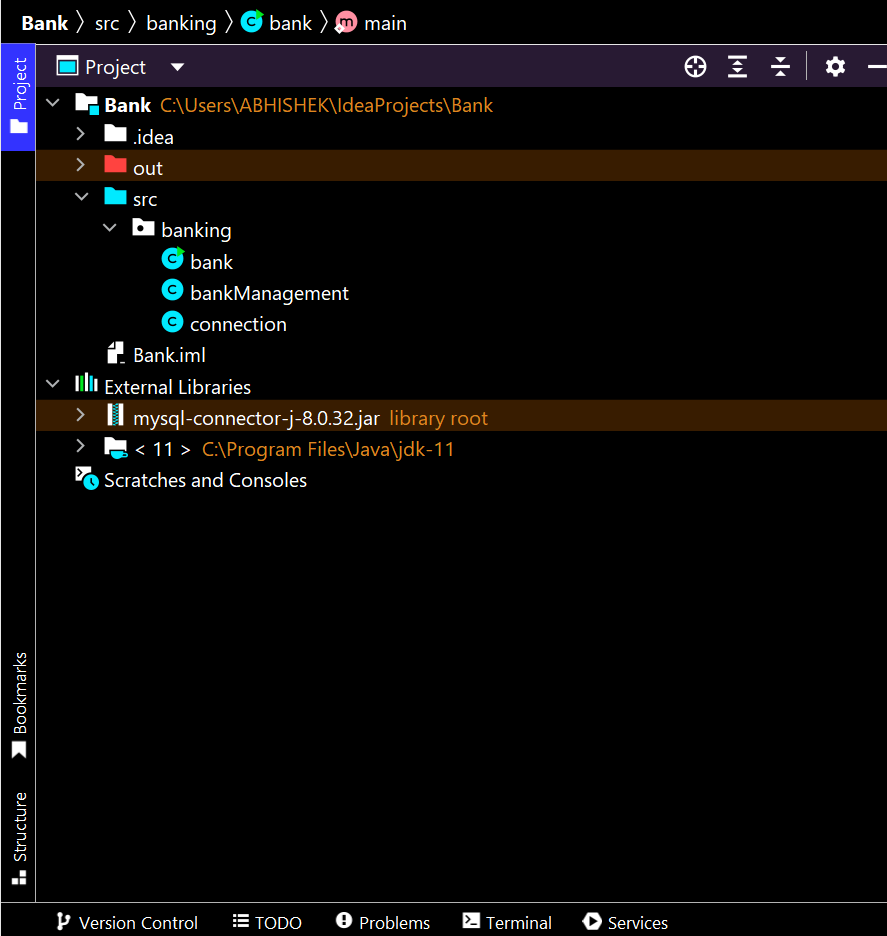
**Step 1) Database Setup:**

* Create a new Database bank
* Create a new table customer
* Insert Table Details as shown below



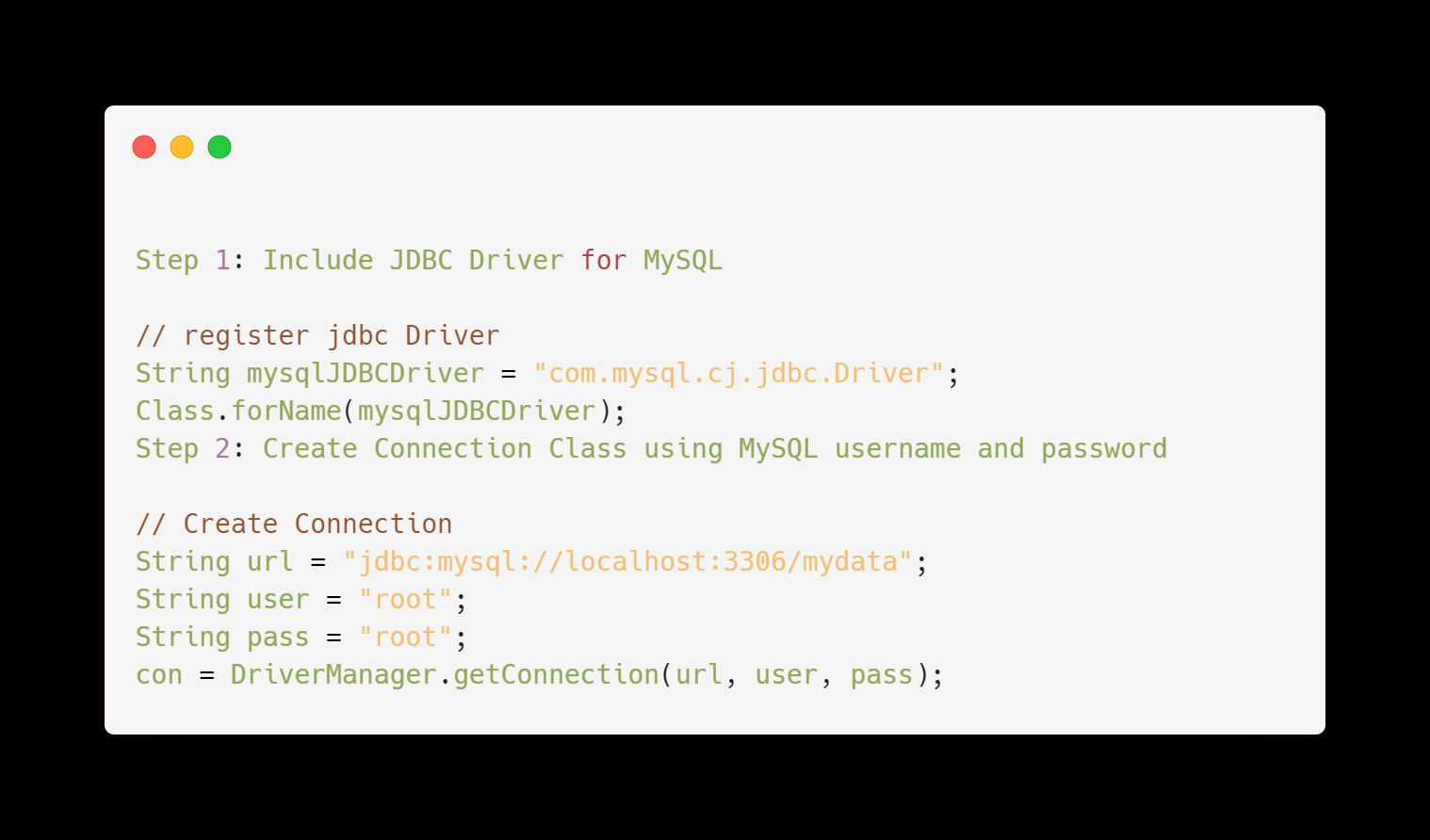
**Step 2)Intellij Project Setup:**

* Create New Project bank
* Create A package name banking
* File configuration is as shown below :



**Step 3) Create a Connection class in the banking package:**

* Create file as shown below:



### >>connection.java

* Which has a class connection which has a method getConnection() , in which all above connection details are implemented ;
* In which I have called inbuilt class of java ‘Connection ’ by initializing its variable;
* In this I have used try and catch block to detect any exceptions and errors.

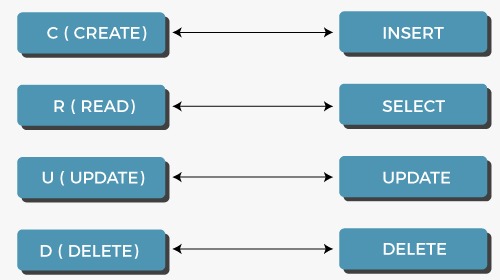


### >> bankmanagment.java

* Which contains all methods of bank software such as Create account , Login account , Transfer amount , Deposit amount , Withdraw amount , View balance and Change password.
* CRUD Operations in SQL:

As we know, CRUD operations act as the foundation of any computer programming language or technology. So before taking a deeper dive into any programming language or technology, one must be proficient in working on its CRUD operations. This same rule applies to databases as well.

Let us start with the understanding of CRUD operations in SQL with the help of examples. We will be writing all the queries in the supporting examples using the MySQL database.



* 1. Create:

In CRUD operations, 'C' is an acronym for create, which means to add or insert data into the SQL table. So, firstly we will create a table using CREATE command and then we will use the INSERT INTO command to insert rows in the created table.

Syntax: CREATE TABLE table\_name (column\_name column\_type constraints);

* 2. Read:

In CRUD operations, 'R' is an acronym for read, which means retrieving or fetching the data from the SQL table. So, we will use the SELECT command to fetch the inserted records from the SQL table. We can retrieve all the records from a table using an asterisk (\*) in a SELECT query. There is also an option of retrieving only those records which satisfy a particular condition by using the WHERE clause in a SELECT query.

Syntax: select \* from table\_name;

* 3. Update:

In CRUD operations, 'U' is an acronym for the update, which means making updates to the records present in the SQL tables. So, we will use the UPDATE command to make changes in the data present in tables.

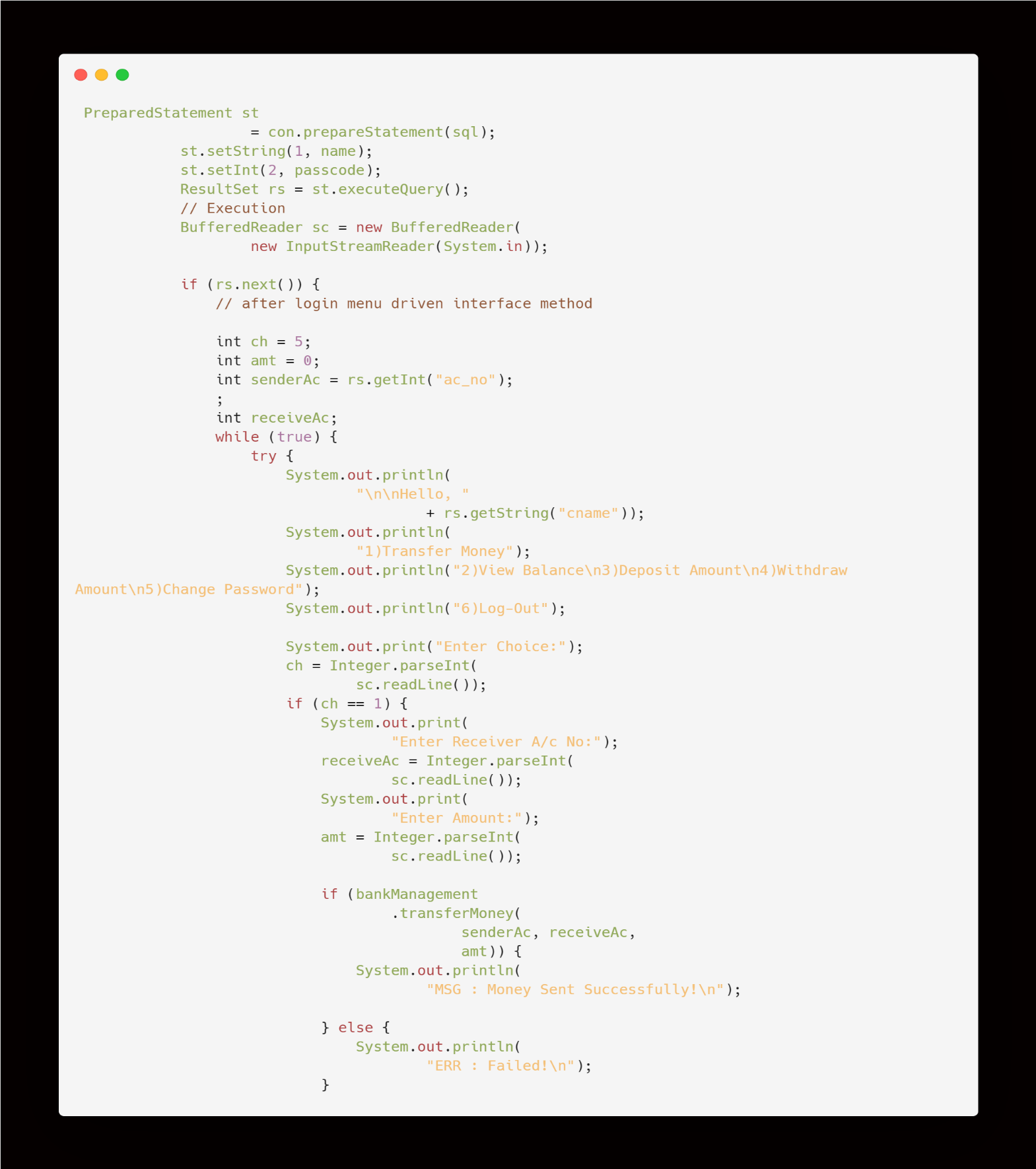
Syntax: Update table\_name set marks = 100 where name = "abhi";

* 4. Delete:

In CRUD operations, 'D' is an acronym for delete, which means removing or deleting the records from the SQL tables. We can delete all the rows from the SQL tables using the DELETE query. There is also an option to remove only the specific records that satisfy a particular condition by using the WHERE clause in a DELETE query.

Syntax: delete from table\_name where marks = 100;

bankmanagment.java:



****

****

****

****

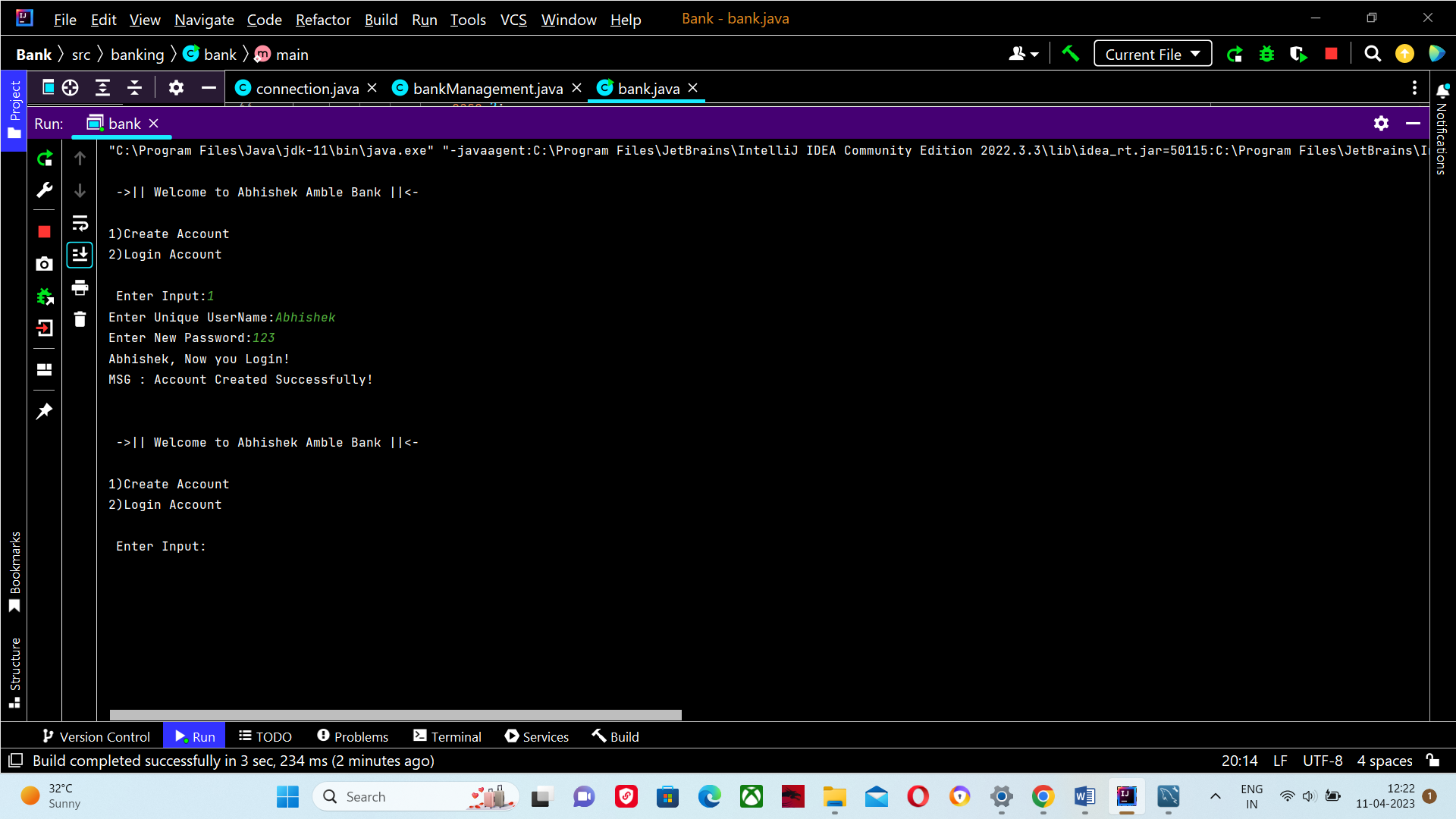
**>>bank.java**

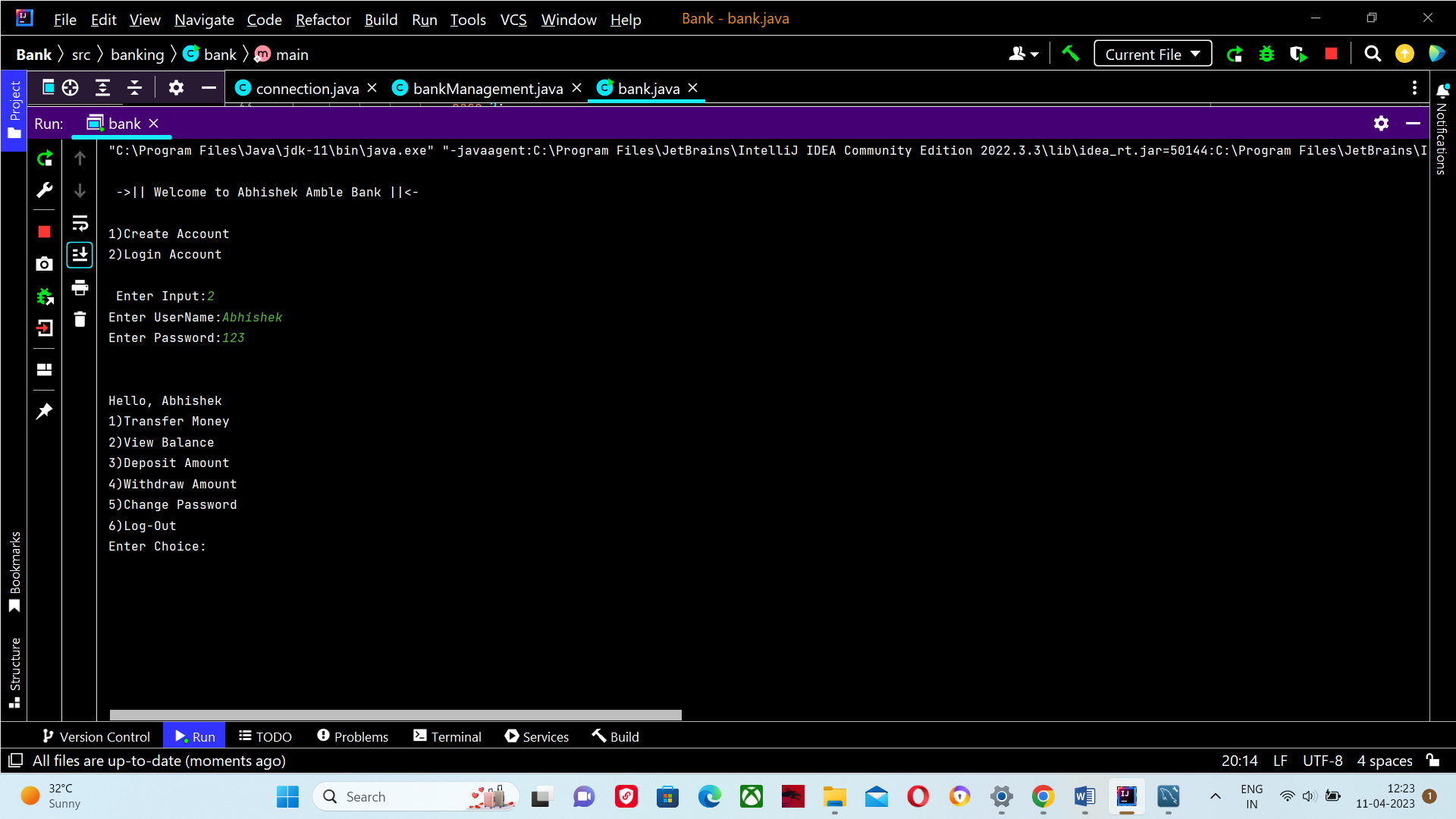
* This java file contains main class bank{ } which is used to call methods in connection.java and bankManagement.java
* Below is the code format of bank.java

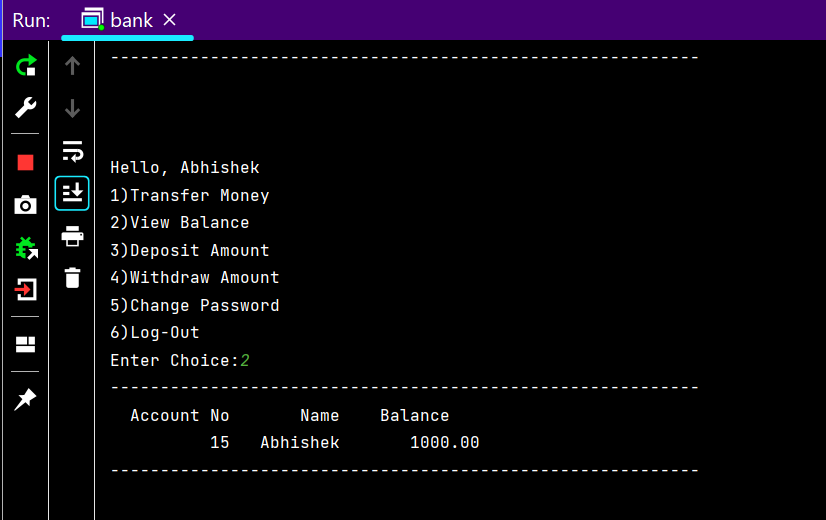


****

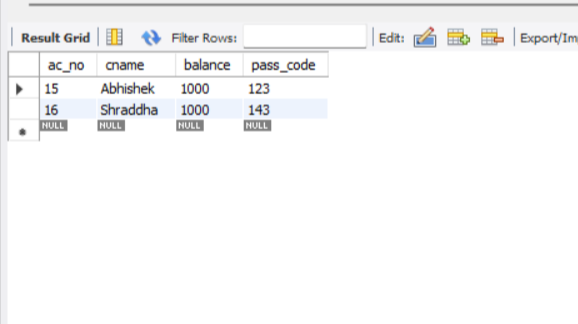
**Output:**

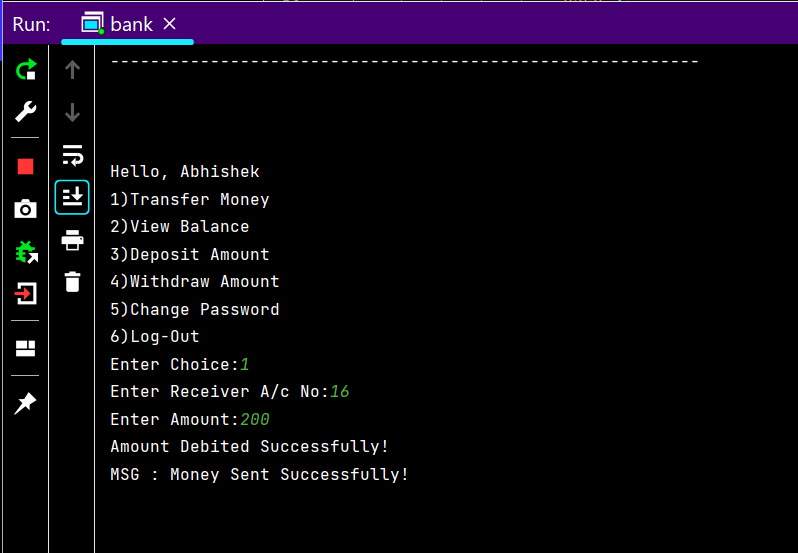
**1)Create Account:**

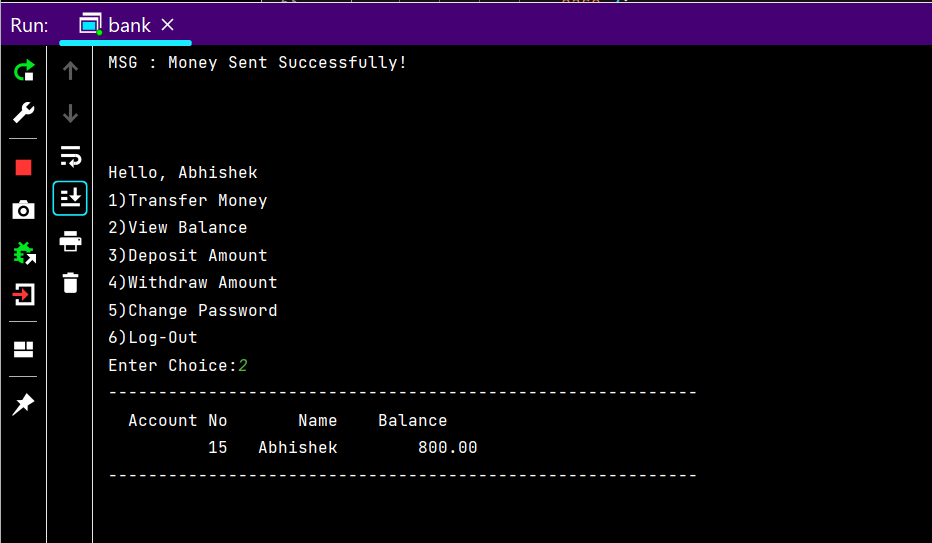
**2)Login Account:**

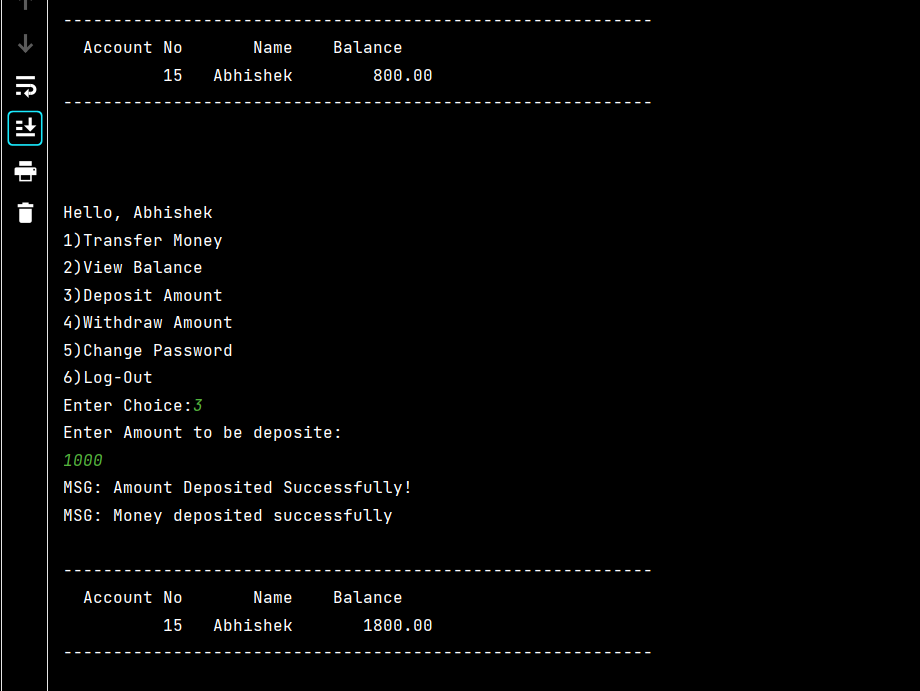
**3) View balance:**

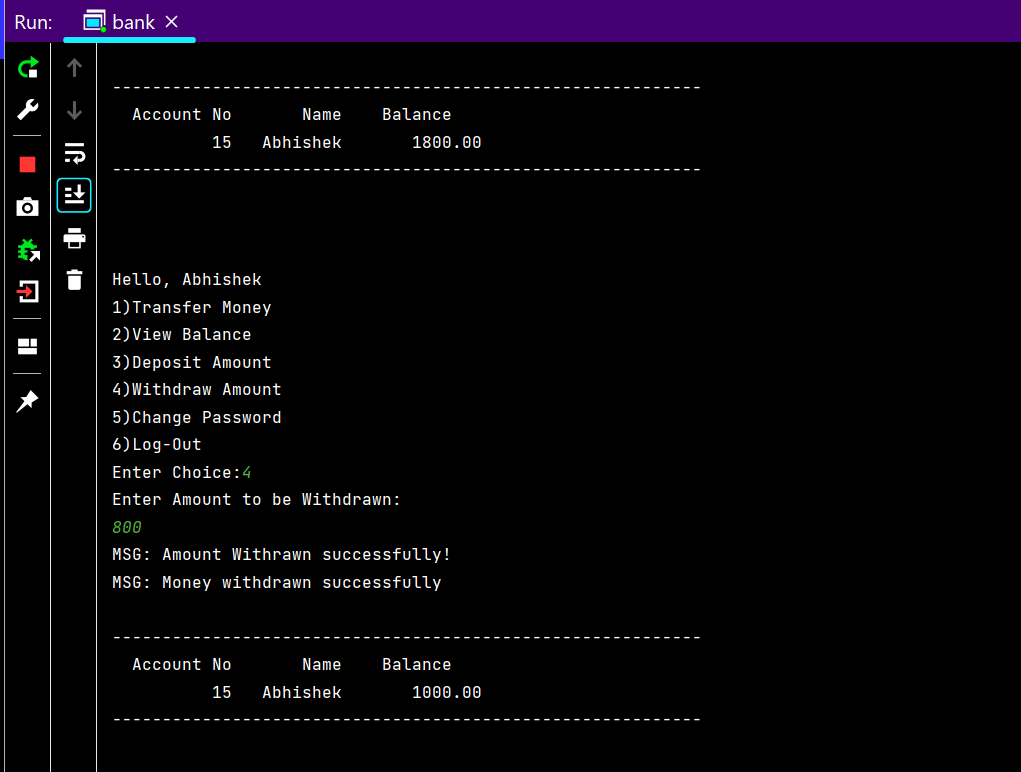
**4)Transfer Money:**

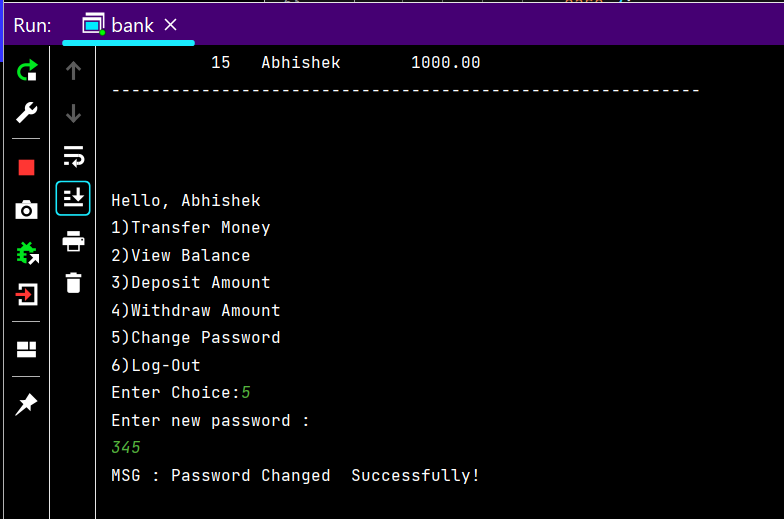
* **database copy:**

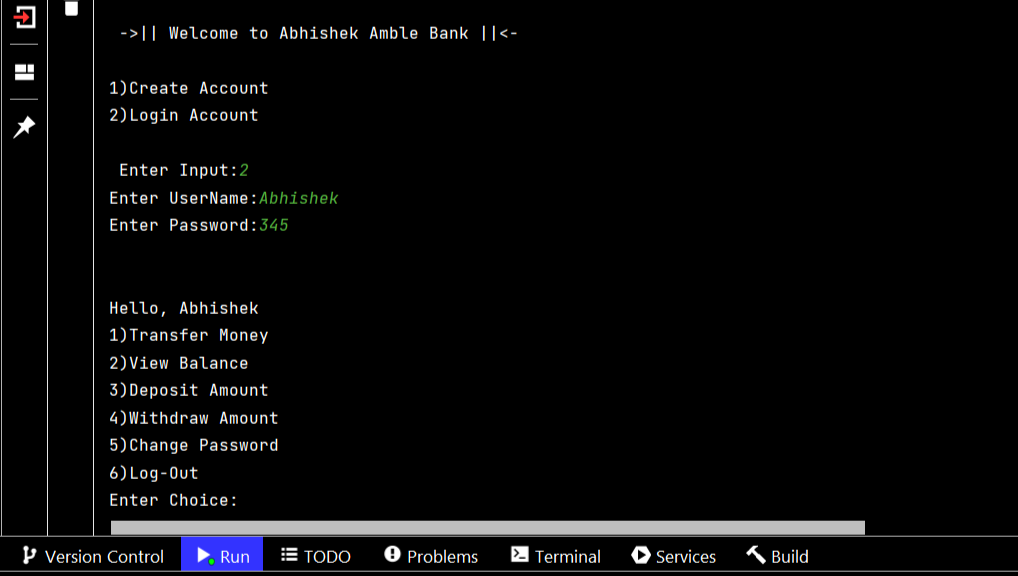
****

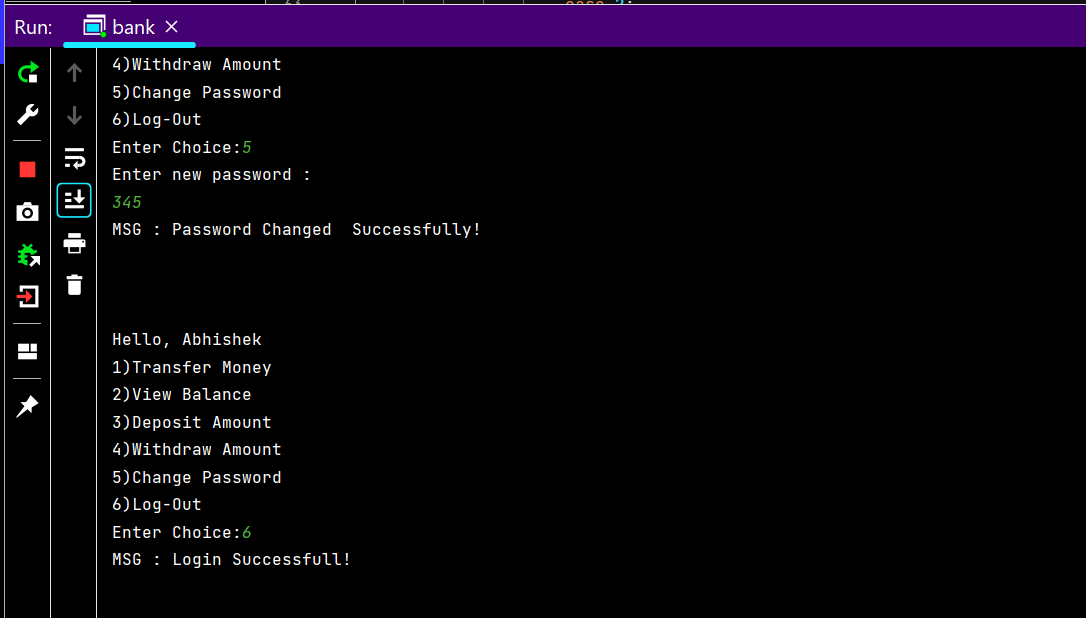
* Balance after money transfer:

**5) Deposit Amount :**

**6)Withdraw Amount:**

**7) Change Password:**

* Login after password change:

**8)Log out:**

**IV. CONCLUSION**

By use of Java programming Language ,JDBC and MYSQL database using CRUD method Bank Software was successfully Implemented and tested Successfully.In which we able to run various processes like any normal banking application.

**V. REFERENCES**

1. <https://www.tutorialspoint.com/jdbc/index.htm>
2. <https://www.tutorialspoint.com/mysql/index.htm>
3. <https://www.javatpoint.com/banking-application-in-java>
4. <https://coderspacket.com/online-banking-application-in-java>
5. <https://www.geeksforgeeks.org/mini-banking-application-in-java>