

# Explanation of the Code

## 1. Import SQL Classes

```
import java.sql.*;
```

This line allows your program to use JDBC classes like Connection, PreparedStatement, ResultSet.

## 2. Start the Program

```
public class DB_Connection {  
    public static void main(String[] args) {
```

This is your main class and main method — the starting point of the program.

## 3. Load Oracle JDBC Driver

```
String driver = "oracle.jdbc.OracleDriver";  
Class.forName(driver);
```

This loads the Oracle driver file so Java can talk to Oracle Database.

## 4. Database Connection Details

```
String url ="jdbc:Oracle:thin:@localhost:1521:XE";  
String user = "system";  
String pass = "12345";
```

- **url** → Where your Oracle database is located
- **user** → Database username
- **pass** → Database password

## 5. Query to Run

```
String table = "users";  
String query = "SELECT * FROM " + table;  
This makes a simple SQL query:  
SELECT * FROM users
```

## 6. Connect to Database

```
Connection con = DriverManager.getConnection(url,user,pass);  
This line actually creates a connection from Java to Oracle.
```

## 7. Prepare and Execute SQL Query

```
PreparedStatement ps = con.prepareStatement(query);  
ResultSet rs = ps.executeQuery();  


- PreparedStatement sends your SQL query to Oracle
- executeQuery() runs the SELECT query
- rs (ResultSet) stores the result coming from the database

```

## 8. Check If Data Exists

```
if(rs.next()) {  
    System.out.println("Connection Sucessful");  
} else {
```

```

        System.out.println("Connection Failed");
    }
    • rs.next() means → is there any row in the table?
    • If yes → print Connection Successful
    • If no → table is empty → print Connection Failed

```

## 9. Close Connection

```
con.close();
```

Always close the database connection after use.

## 10. Catch Errors

```

catch (ClassNotFoundException e) { ... }
catch (SQLException e){ ... }

```

These catch driver loading errors or database errors.

## Screenshot of Output:

The screenshot shows the IntelliJ IDEA interface. In the code editor, the file `DB_Connection.java` is open, displaying Java code for connecting to an Oracle database. The run output window at the bottom shows the command used to run the application and the resulting output: "Connection Sucessful".

```

import java.sql.*;
public class DB_Connection {
    public static void main(String[] args) {
        try{
            String driver = "oracle.jdbc.OracleDriver";
            Class.forName(driver);
            String url ="jdbc:Oracle:thin:@localhost:1521:XE";
            String user = "system";
            String pass = "12345";
            String table = "users";
            String query = "SELECT * FROM " + table;
            Connection con = DriverManager.getConnection(url,user,pass);
        }
    }
}

```

```

"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:E:\IntelliJ\IntelliJ IDEA Community Edition 2025.1.3\lib\idea_rt.jar=56199" -Dfile.encoding=UTF-8 DB_Connection
Connection Sucessful
Process finished with exit code 0

```