DATABASE MANAGEMENT
WEEK 8 ASSIGNMENT

1. Identify the role of Driver Manager.

- The main purpose of the *driver manager* is to stack the Java Database connectivity drivers and move Java Database connectivity function calls from the application to the correct driver.
- The driver manager also handles the initialization of Java Database connectivity.
- It is responsible to log all function calls.
- Further the driver manger is accountable for basic error checking.

2. Create a schema Employee with relevant attributes in SQL. (a)Create a JDBC program which Includes the required package (b) Enter the code to create a connection and statement. (c)Include JDBC methods throw SQLException. (d)Load the MySQL DriverManager (e) Perform a Database Query and View Results of Employee Database.

```
import java.sql.*;
public class FirstnameExample {
  static final String DB_URL = "jdbc:mysql://localhost/Sindhu";
  static final String USERNAME = "sindhu";
  static final String PASSWORD  = "sindhu123";
  static final String QUERY = "SELECT id, firstname, lastname, age FROM EMP";

  public static void main(String[] args) {
    // Open a connection
    try(Connection conn = DriverManager.getConnection(DB_URL, USERNAME, PASSWORD );
      Statement stmt = conn.createStatement();
      ResultSet rs = stmt.executeQuery(QUERY);) {
      // Extract data from result set
      while (rs.next()) {
        // Retrieve by column name
        System.out.print("ID: " + rs.getInt("id"));
        System.out.print(", Age: " + rs.getInt("age"));
        System.out.print(", Firstname: " + rs.getString("firstname"));
        System.out.println(", Lastname: " + rs.getString("lastname"));
      }
    } catch (SQLException e) {
      e.printStackTrace();
    }
  }
}
```

}

3. What type of package is used in JDBC to include Exception?

- Java.sql.BatchUpdateException
- java.sql.SQLException
- java.sql.DataTruncation
- java.sql.SQLWarning

4. "PreparedStatement" is a valid statement in JDBC. Justify

PreparedStatement Interface: Over a generic statement interface PreparedStatement
Interface provides additional functionality. Unlike Statement interface this interface allowed
o pass parameters inside the method.
Eg: PreparedStatement stmt = null;
String SQL = "Update Emp SET salary = ? WHERE age = ?";
stmt = conn.prepareStatement(SQL);
stmt.close();