DUTY SCHEDULE ALLOTMENT SYSTEM

JAVA - Micro project

Coding:

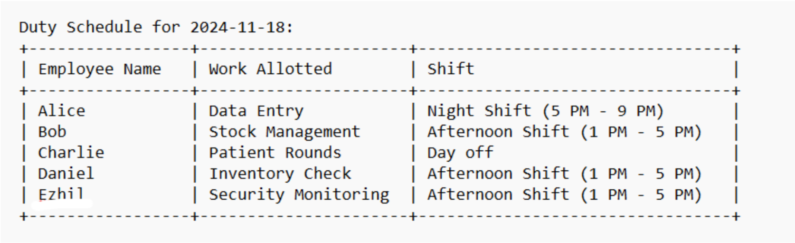
```
import java.sql.*;
import java.util.*;

public class DutyScheduleAllotment {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);

      // Database connection
      String jdbcURL = "jdbc:mysql://localhost:3306/DutyScheduleDB";
      String dbUser = "root";
         String dbPassword = "Pavi@2046";

      // shifts
      String[] shifts = {
         "Morning Shift (9 AM - 1 PM)",
         "Afternoon Shift (1 PM - 5 PM)",
         "Night Shift (5 PM - 9 PM)",
         "Day Off"
      };

      // works
      String[] works = {
         "Inventory Check",
         "Patient Rounds",
         "Data Entry",
            "Security Monitoring",
            "Stock Management"
        };

        try (Connection connection = DriverManager.getConnection(jdbcURL, dbUser,
dbPassword)) {
            System.out.println("Connected to the database!");

            // usser- date
            System.out.print("Enter the date for the schedule (YYYY-MM-DD): ");
            String scheduleDate = scanner.nextLine();

            // Ask - no. of employees
            System.out.print("Enter the number of employees for " + scheduleDate + ": ");
            int numEmployees = scanner.nextInt();
            scanner.nextLine(); // Consume the newline character

            // employee data
            String[] employeeNames = new String[numEmployees];
            String[] employeeWorks = new String[numEmployees];
            String[] employeeShifts = new String[numEmployees];

            // emp.names & work
            for (int i = 0; i < numEmployees; i++) {
                System.out.print("Enter the name of employee " + (i + 1) + ": ");
                employeeNames[i] = scanner.nextLine();

                // Display works
                System.out.println("Available works:");
                for (int j = 0; j < works.length; j++) {
                    System.out.println((j + 1) + ". " + works[j]);
                }


                // select work
                System.out.print("Select the work for " + employeeNames[i] + " (enter number 1-"
+ works.length + "): ");
                int workChoice = scanner.nextInt();
                scanner.nextLine(); // Consume the newline character
                employeeWorks[i] = works[workChoice - 1];
            }


            //assign shifts
            Random random = new Random();
            for (int i = 0; i < numEmployees; i++) {
                employeeShifts[i] = shifts[random.nextInt(shifts.length)];
            }


            // database
            String insertQuery = "INSERT INTO DutySchedule (schedule_date, employee_name,
work_allotted, shift) VALUES (?, ?, ?, ?)";
            PreparedStatement statement = connection.prepareStatement(insertQuery);


            for (int i = 0; i < numEmployees; i++) {
                statement.setDate(1, java.sql.Date.valueOf(scheduleDate)); // Explicitly use
java.sql.Date
                statement.setString(2, employeeNames[i]);
                statement.setString(3, employeeWorks[i]);
                statement.setString(4, employeeShifts[i]);
                statement.executeUpdate();
            }


            System.out.println("\nDuty schedule has been successfully saved to the database!");
```

```
            //final schedule
            displaySchedule(connection, scheduleDate);

        } catch (SQLException e) {
            System.out.println("Error connecting to the database: " + e.getMessage());
        }

        scanner.close();
    }

    // schedule from the database
    private static void displaySchedule(Connection connection, String scheduleDate) throws SQLException {
        String selectQuery = "SELECT employee_name, work_allotted, shift FROM DutySchedule WHERE schedule_date = ?";
        PreparedStatement statement = connection.prepareStatement(selectQuery);
        statement.setDate(1, java.sql.Date.valueOf(scheduleDate)); // Explicitly use java.sql.Date
        ResultSet resultSet = statement.executeQuery();

        System.out.println("\nDuty Schedule for " + scheduleDate + ":");
        System.out.println("+-----------------+----------------------+----------------------+");
        System.out.printf("| %-15s | %-20s | %-20s |\n", "Employee Name", "Work Allotted", "Shift");
        System.out.println("+-----------------+----------------------+----------------------+");

        while (resultSet.next()) {
            String name = resultSet.getString("employee_name");
            String work = resultSet.getString("work_allotted");
            String shift = resultSet.getString("shift");
            System.out.printf("| %-15s | %-20s | %-20s |\n", name, work, shift);
        }
        System.out.println("+----------------+---------------------+---------------------+");
    }
}
```

OUTPUT:

```
Enter the date for the schedule (YYYY-MM-DD): 2024-11-18
Enter the number of employees for 2024-11-18: 5
Enter the name of employee 1: Alice
Available works:
1. Inventory Check
2. Patient Rounds
3. Data Entry
4. Security Monitoring
5. Stock Management
Select the work for Alice (enter number 1-5): 3
Enter the name of employee 2: Bob
Available works:
1. Inventory Check
2. Patient Rounds
3. Data Entry
4. Security Monitoring
5. Stock Management
Select the work for Bob (enter number 1-5): 5
Enter the name of employee 3: Charlie
Available works:
1. Inventory Check
2. Patient Rounds
3. Data Entry
4. Security Monitoring
5. Stock Management
Select the work for Charlie (enter number 1-5): 2
Enter the name of employee 4: Daniel
Available works:
1. Inventory Check
2. Patient Rounds
3. Data Entry
4. Security Monitoring
```

```
5. Stock Management

Select the work for Daniel (enter number 1-5): 1

Enter the name of employee 5: Ezhil

Available works:

1. Inventory Check

2. Patient Rounds

3. Data Entry

4. Security Monitoring

5. Stock Management

Select the work for Ezhil (enter number 1-5): 4


Duty schedule has been successfully saved to the database!
```



```
Duty Schedule for 2024-11-18:
+-----------------+----------------------+---------------------------------+
| Employee Name   | Work Allotted        | Shift                           |
+-----------------+----------------------+---------------------------------+
| Alice           | Data Entry           | Night Shift (5 PM - 9 PM)       |
| Bob             | Stock Management     | Afternoon Shift (1 PM - 5 PM)   |
| Charlie         | Patient Rounds       | Day off                         |
| Daniel          | Inventory Check      | Afternoon Shift (1 PM - 5 PM)   |
| Ezhil           | Security Monitoring  | Afternoon Shift (1 PM - 5 PM)   |
+-----------------+----------------------+---------------------------------+
```

23ADR124

PAVITHRA K