Duty Schedule Allotment System
(Micro Project)

Name: Pavithra K

Roll No: 23ADR124

Date: 18/11/2024

Problem Statement:

Manual duty allotment scheduling is time-consuming and prone to errors due to:

- Difficulty in managing employee schedules
- Inefficient use of resources
- Lack of automation

Project Overview:

The system automates employee duty schedules by:

- Managing employee data
- Assigning shifts and tasks
- Storing data in a MySQL database

Features:

- Employee data management
- Shift and task assignment
- Schedule generation
- Database integration
- User-friendly interface

Advantages:

- Efficient duty allotment
- Reduced errors
- Improved resource utilization
- Easy data management

Tools Used:

- Development Tools: Java IDE, MySQL Workbench, Command Prompt
- Frameworks/Technologies: JDBC for database integration

Java Code Overview:

- Uses Scanner for user input.
- Connects to MySQL database using JDBC.
- Implements random shift assignment.
- Data is stored in the DutySchedule table.

Flow Chart:

1. Database connection.
2. Gather inputs (Date, No. of employees, Employee name, Work selection).
3. Add data to the database.
4. Display the schedule from the database.
5. Terminate process.

Code Snippets:

Shift Assignment:

Random shift allocation using Java's Random class.

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

Generation Algorithm:

Displays employee name, shift, and assigned work.

```
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();
}
```

Database Query:

Inserts schedule data into DutySchedule using prepared statements.

```
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();
        }
```

Results:

Outputs:

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 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)    |
+-----------------+----------------------+----------------------------------+
```

Conclusion:

This system efficiently automates duty scheduling, reduces errors, and optimizes resource management.