MANAGEMENT INFORMATION SYSTEM

LAB EXPERIMENTS

Jashwini A-192124192

9) A college has more than thousand security persons, who are instructed to give duties at  different places within the campus. Additionally, they also maintain a routine, which  contains all information, such as Date, Duty Start Time, Duty End Time, and Place.  Most importantly, all the places are covered by at least one security person. If a security  person takes leave, manual entry is done against that person. Finally, at the end of a  month, the security persons get paid for their duties, while considering the number of  leaves as well. You can see that the manual calculation/operation is a heavy task for the  security manager. Therefore, the objective is to build an Online security management  system using class diagram through which entire security system within the campus can  be controlled in an efficient manner

**Aim:**

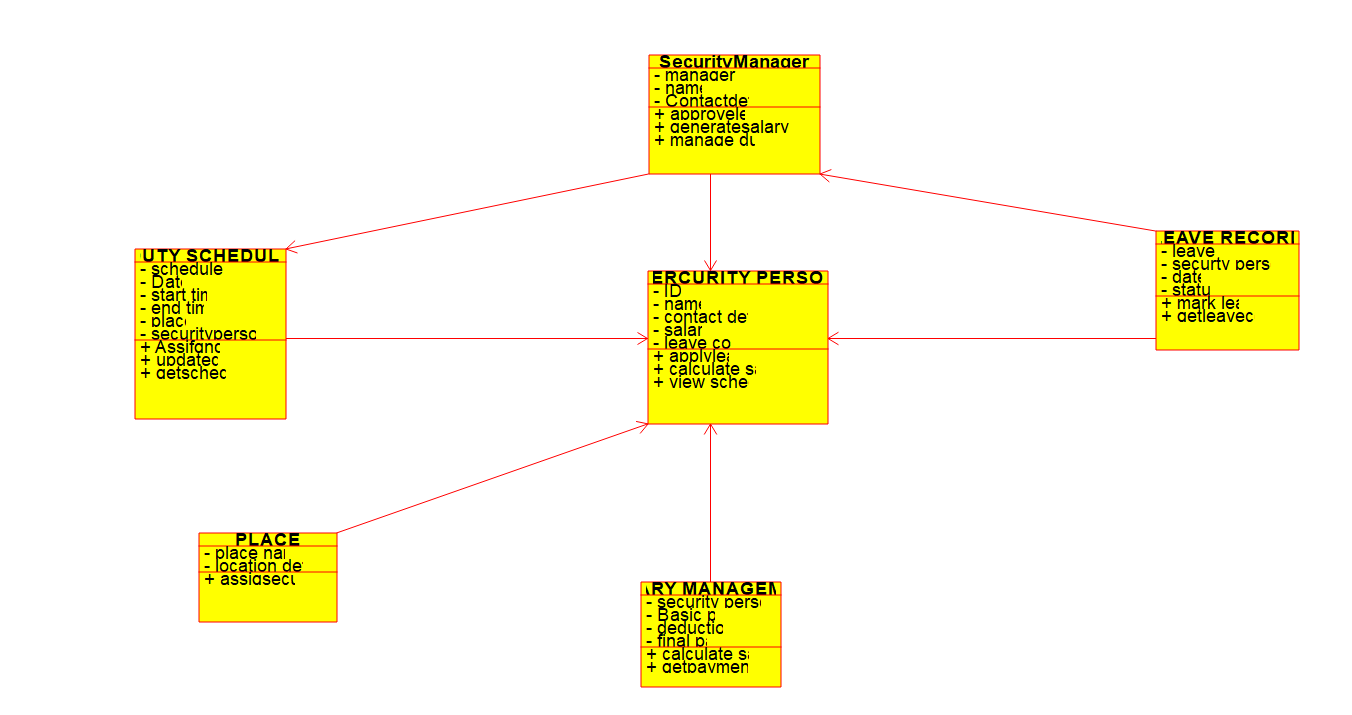
To develop a **UML Class Diagram** for an **Online Security Management System** to efficiently manage the security personnel, their schedules, leave records, and salary calculations.

**Procedure:**

1. **Identify Key Classes:**
   * **SecurityPerson**: Represents a security guard with attributes like ID, name, assigned duty, and leave records.
   * **DutySchedule**: Stores details of duty assignments, including date, start time, end time, and place.
   * **LeaveRecord**: Tracks leave applications and approvals.
   * **Salary**: Calculates the monthly salary based on duties completed and leaves taken.
   * **SecurityManager**: Manages security personnel, assigns duties, approves leaves, and processes salaries.
   * **Place**: Represents locations that must be guarded, ensuring full coverage.
2. **Define Attributes and Methods:**
   * **SecurityPerson**
     + Attributes: ID, Name, Contact, Assigned Duty, Leaves Taken
     + Methods: RequestLeave(), ViewSchedule()
   * **DutySchedule**
     + Attributes: Date, Start Time, End Time, Place, Assigned Security Person
     + Methods: AssignDuty(), UpdateSchedule()
   * **LeaveRecord**
     + Attributes: SecurityPersonID, LeaveDate, ApprovalStatus
     + Methods: ApplyLeave(), ApproveLeave()
   * **Salary**
     + Attributes: SecurityPersonID, Basic Pay, Deductions, Net Salary
     + Methods: CalculateSalary()
   * **SecurityManager**
     + Attributes: ID, Name, Contact
     + Methods: AssignDuties(), ApproveLeaves(), ProcessSalaries()
   * **Place**
     + Attributes: PlaceID, Location Name
     + Methods: AssignSecurity()
3. **Establish Relationships:**
   * A **SecurityPerson** is assigned to a **DutySchedule**.
   * A **DutySchedule** is linked to a **Place** to ensure coverage.
   * A **SecurityPerson** can request **LeaveRecord**, which must be approved by the **SecurityManager**.
   * The **SecurityManager** assigns duties and processes salaries based on duty records and leave deductions.

**Output:**

**Class diagram**

****

**Result:**

A **UML Class Diagram** will be created to visualize the **Online Security Management System**.