Database Design

Part A

The following describes a real estate firm that lists property for sale.

- **The firm has a number of sales offices in several states. Each sales office has an office number, phone number, and address (including street address, city, state and zipcode).**

- **Each sales office has at least one employee. It may have multiple employees. Information about an employee includes employee's ID, last name, first name. An employee is affiliated to one sales office.**

- **Each sales office has one and only one manager. An employee may manage only the sales office which he/she is assigned to.**

- **The firm lists properties for sale. Each property has an ID number. Additional information about a property includes address (including street address, city, state and zipcode) and square footage.**

- **Each unit of property must be listed with one (and only one) of the sales offices. A sales office may have any number of properties listed, or may have no properties listed.**

- **An owner may own one or more units of property. You want to keep track of the owners' information, such as owner's ID and name (may be an organization). Each unit of property may have more than one owner. So it is necessary to record the percentage of ownership for each property by an owner.**

Develop an E-R diagram for this case using MySQL Workbench.

- **In the space below, provide the screenshot of your ERD.**
- **For each entity, include identifier and all attributes.**
- **Name the relationship if there is ambiguity and diagram both the minimum and maximum cardinalities.**
- **Note any assumptions you believe you have to make in order to develop a complete diagram.**

Answers (part A)
Identifier:

- **salesoffice:** To store the sale office details
- **employee:** To store the employee details
- **property:** To store the property details
- **owner:** To store the owner details

Attributes to the entities:

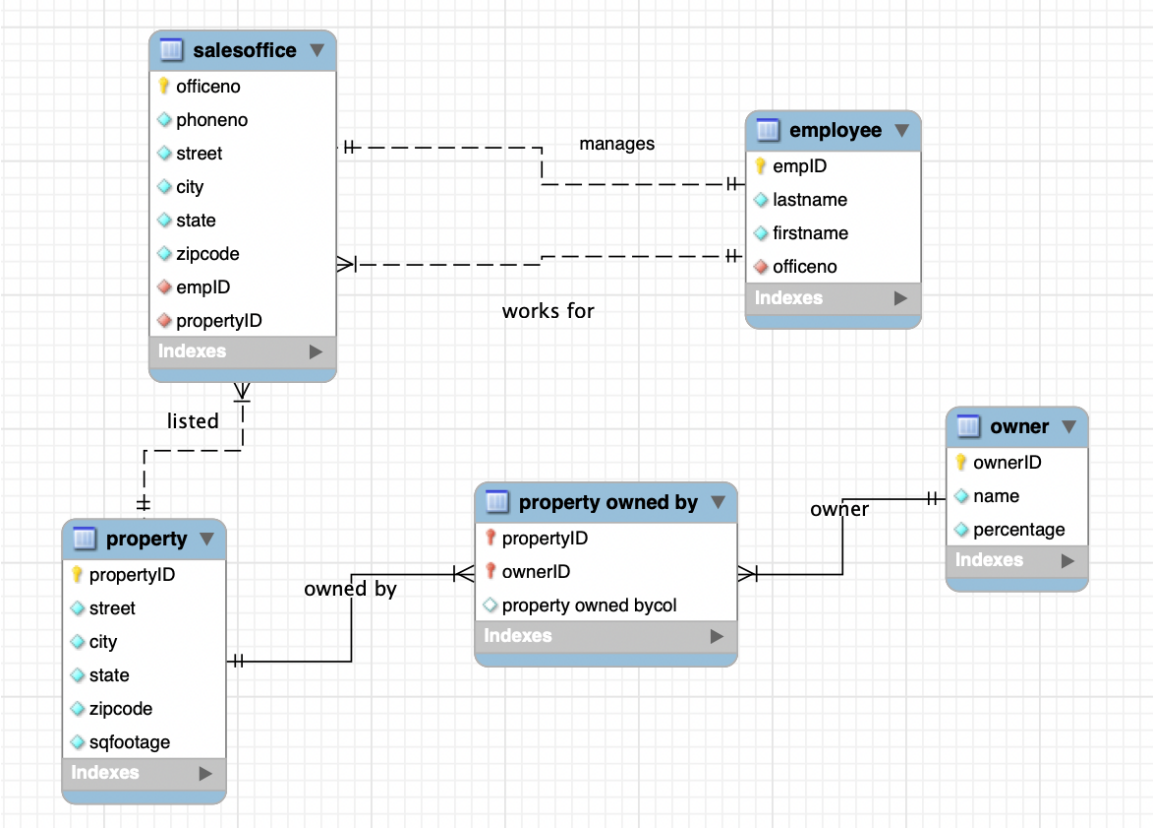
- saleOffice(**officeno**, phoneno, Address (street, city, state, zipcode); Added officeno as primary key
- employee (**empID**, lastname, firstname)
- property(**propertyID**, Address (street, city, state, zipcode), sqfootage)
- owner (**ownerID**, name, percentage); Added ownerID as primary key

Assumptions:

- empID is is unique for each employee throughout all Offices
- Propertyno is unique for all properties
- Officeno is unique for all offices
- Every employee has a salesoffice as an employee belongs to one salesoffice only
- Every office has one and only one manager who is an employee.

Relationships:

- A salesoffice must have at least one employee and can have many employees (cardinality: mandatory one), but an employee can only work for one single salesoffice (one to many relationship).
- An employee manages only one salesoffice, and one salesoffice can have only one manager (one to one relationship)
- Each unit of property must be listed with only one salesoffice, however, a salesoffice can have many properties listed, or none at all (many to one, cardinality optional many)
- A property can be owned by owner, and an owner can have multiple property (many to many)



Part B

Projects, Inc, is an engineering firm with approximately 500 employees. It has many projects nationwide. To manage the employee and project information, Projects, Inc. decides to build a database.

The database should keep track of all employees, their names, dates of birth, their skills, and departments worked in. Each employee is given a job title (for example, engineer, secretary, and so on). An employee can have many skills (preparing material requisitions, checking drawings, and so on).

There are eleven different departments, each with a unique name. Each employee is affiliated to only one department.

Each department maintains a list of equipment. In addition to model, size, weight. Each equipment has records for scheduled and unscheduled maintenance. The departments want to track the maintenance expense.

The company has many projects nationwide (e.g. Southwest Refinery, California Petrochemicals, and so on). The company has an estimated cost for each project. A project requires at least one employee to complete but some large projects have to require more than one employee. An employee can work on many projects.

Develop an E-R diagram for this case using MySQL Workbench.

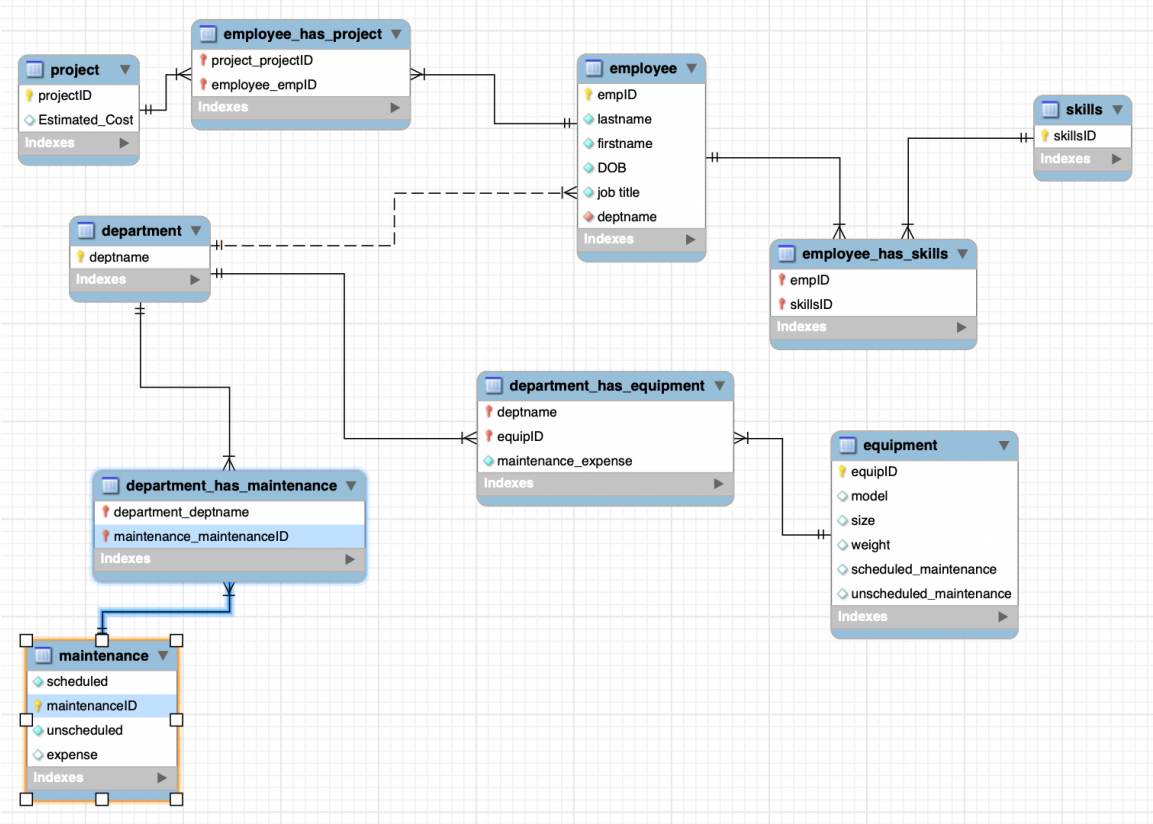
- **In the space below, provide the screenshot of your ERD.**
- **For each entity, include identifier and all attributes.**
- **Name the relationship if there is ambiguity and diagram both the minimum and maximum cardinalities.**
- **Note any assumptions you believe you have to make in order to develop a complete diagram.**

Identifier:

- **employee:** To store the employee details
- **skills:** To store the skills details
- **department:** To store the department details
- **equipment** To store the equipment details
- **project:** To store the projectdetails

Attributes:

- Employee: empID, lastname, firstname, DOB, job title
- Skills: skillsID
- Department: deptname
- Equipment: equipID, model, size, weight
- Maintenance: scheduledmaintenance, unscheduledmaintenance, expense
- Projects: projectID, Estimated_ Cost


project
projectID
Estimated_Cost
Indexes
employee_has_project
project_projectID
employee_empID
Indexes
employee
empID
lastname
firstname
DOB
job title
deptname
Indexes
skills
skillsID
Indexes
department
deptname
Indexes
employee_has_skills
empID
skillsID
Indexes
department_has_equipment
deptname
equipID
maintenance_expense
Indexes
equipment
equipID
model
size
weight
scheduled_maintenance
unscheduled_maintenance
Indexes
department_has_maintenance
department_deptname
maintenance_maintenanceID
Indexes
maintenance
scheduled
maintenanceID
unscheduled
expense
Indexes