

Name: Mursid Ahammed

Assessments

Solve No.1

You have to develop a Human Resource Management database for a company. It requires that you have to add some information of departments, employees and projects including company. As mentioned, that every department has many employees and each employee work for a department and each department is leading by only one manager who is also an employee. Initially a new department need not have any employee. Here, though an employee belongs a department but they can work for different projects at the same time. Each employee can work for different job position like Developer, Programmer and front-end designer.

Steps of Drawing ERD

1. Identify the Entities Required;
2. Identify the Attributes and Primary key for each Entity;
3. Identify the Relationship needed;
4. Identify the Cardinality Ratio and Participation;
5. Draw the Diagram.

Step 1: Identify the Entities Required

You have to develop a Human Resource Management database for a **company**. It requires that you have to add some information of **departments**, **employees** and **projects** including company. As

mentioned, that every department has many employees and each employee work for a department and each department is leading by only one manager who is also an employee. Initially a new department need not have any employee. Here, though an employee belongs a department but they can work for different projects at the same time. Each employee can work for different **job position** like Developer, Programmer and front-end designer.

Step 2: Identify the Attributes and Primary key for each Entity

- **Company** (Id, company_name, location, founded_date, Managing_director)
- **Departments** (Department_id, name, number_of_employee, manager_id, employee_id, company_id)
- **Employees** (id, first_name, middle_name, last_name, joining_date, email, phone, date_of_birth, gender, designation, salary, department_id, job_title)
- **Address** (Id, employee_id, Village_name, post_office, post_code, Upazila, District)
- **Phone** (Id, number, code, employee_id, sim_operator)
- **Job Position** (position_id, position_name, employee_id, name_of_employee, name_of_department)
- **Projects** (Project_id, name, number_of_employee, start_date, end_date, budget, company_id, description).

Step 3: Identify the Relationship needed

1. Employees work_for Department
2. Employee Manages Department
3. Employees work_on Project
4. Department Controls Project
5. Manager leading department

Step 4: Identify the Cardinality Ratio and Participation

1. **Company to Department:** One-to-Many

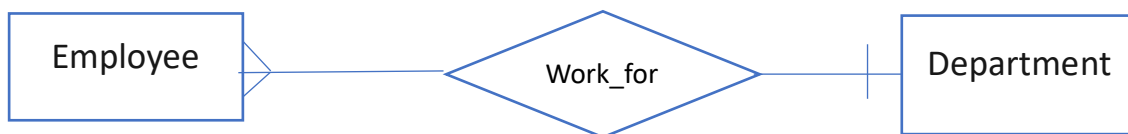
Participation: Total for Department



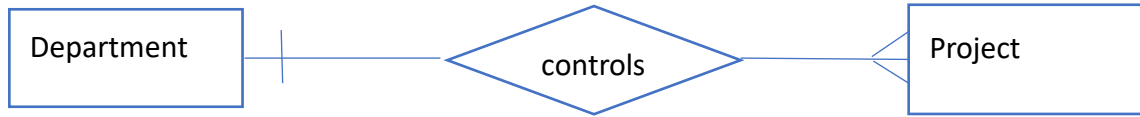
2. **Employee to department:** many to one

total for Employee

Partial for Department

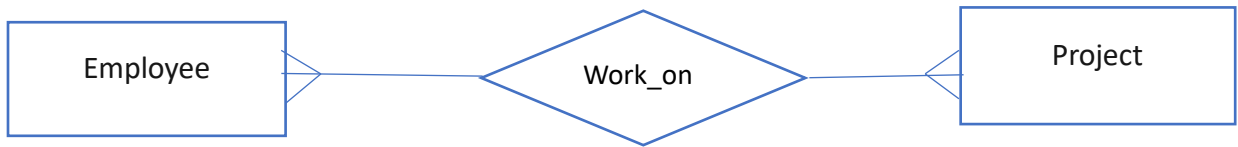


3.Department to project: One-to-Many



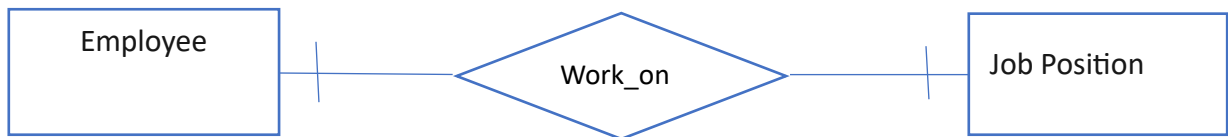
4.Employee to Project: Many-to-Many

Total for both Employee and Project

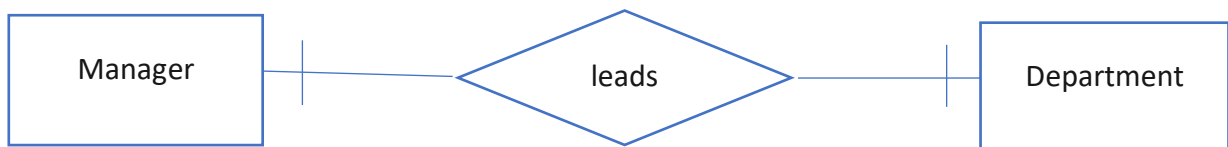


5.Employee to Job Position: One-to-One

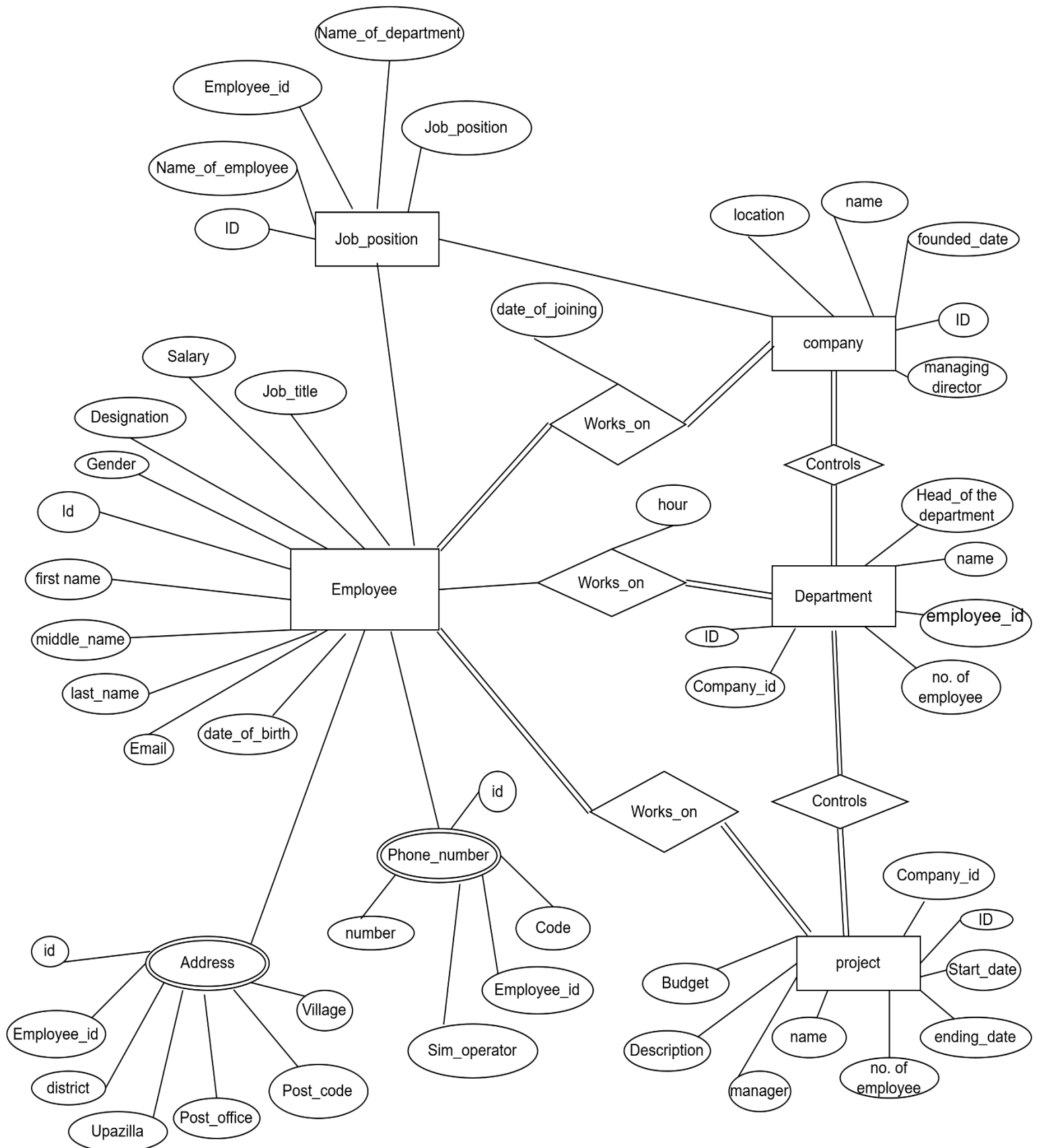
Total for Job Position



6.Manager leads Department: one to one



Step 5: ERD Diagram



phpMyAdmin

Recent

Favorites

department

department management sys

ecommerce

edge

edge.

edge training management sys

ems

exam management system

genetic

human_resource_manage

New

address

company

departments

employees

job_position

phone

projects

ims

information_schema

insert data

library management system

lms

mysql

performance schema

Server: 127.0.0.1 - 2 Database: human_resource_management

Structure

SQL

Search

Query

Export

Import

Operations

Privileges

Routines

Events

Triggers

Tracking

Filters

Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> address	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
<input type="checkbox"/> company	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> departments	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
<input type="checkbox"/> employees	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
<input type="checkbox"/> job_position	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
<input type="checkbox"/> phone	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
<input type="checkbox"/> projects	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
7 tables	Sum	0	InnoDB	utf8mb4_general_ci	208.0 KiB	0 B

☐ Check all

With selected:

Print

Data dictionary

Create new table

Table name

Number of columns

4

Create

Solve NO.2

ER diagram for university management system

Step 1: Identify the Entities Required

Creating a university management system (UMS) database involves setting up tables for various entities like **students**, **courses**, **faculty**, **departments**, **classes**, and **Enrollment**.

Step 2: Identify the Attributes and Primary key for each Entity

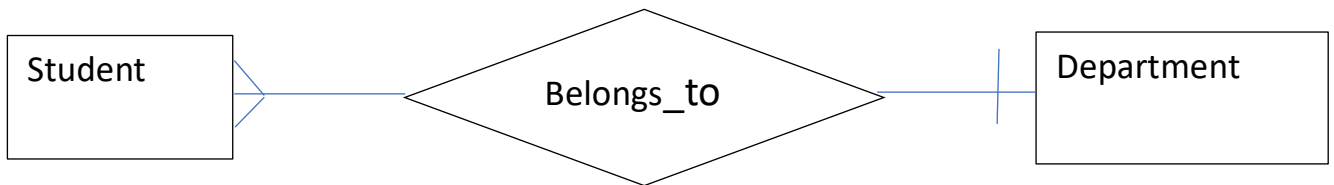
- **Student** (Student_id (Primary Key), First_name, Last_name, Date_of_birth, Gender, Phone_number, Email, Department_id (Foreign Key references Department), Enrollment_date).
- **Department** (Department_id (Primary Key), Department_name, Head_of_department (Foreign Key references Faculty)).
- **Course** (Course_id (Primary Key), Course_name, Credits, Department_id (Foreign Key references Department), Course_description).
- **Faculty** (Faculty_id (Primary Key), First_name, Last_name, Email, Phone_number, Department_id (Foreign Key references Department), Joining_date).
- **Class** (Class_id (Primary Key), Faculty_id (Foreign Key references Faculty), Course-id (Foreign Key references Course), Semester, Year).
- **Enrollment** (Enrollment_id (Primary Key), Student_id (Foreign Key references Student), Class_id (Foreign Key references Class), Grade, Enrollment_date).

Step 3: Identify the Relationship needed

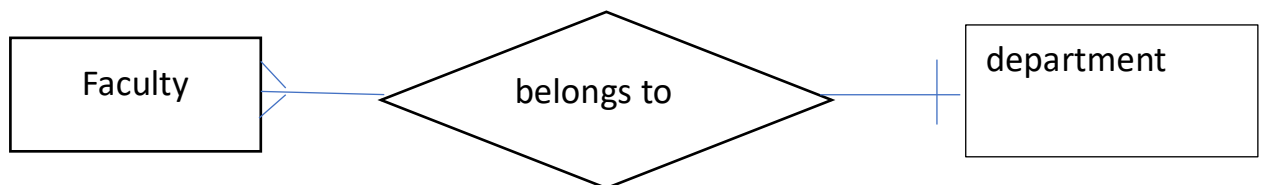
1. Student belongs to department
2. Faculty member belongs to department
3. Each course is offered by department
4. Faculty member teach Courses
5. Student can enroll in Courses
6. Class has multiple timetable slots.

Step 4: Identify the Cardinality Ratio and Participation

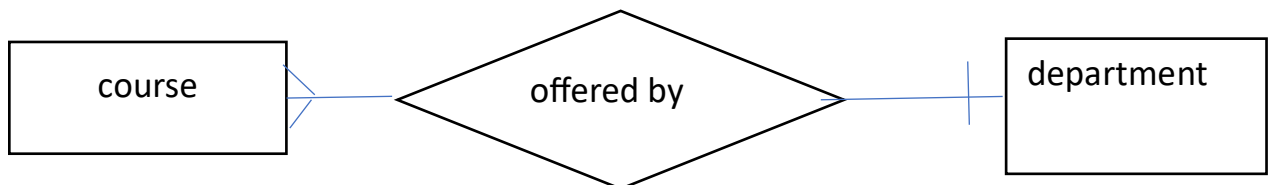
1. Student - Department: Many-to-One



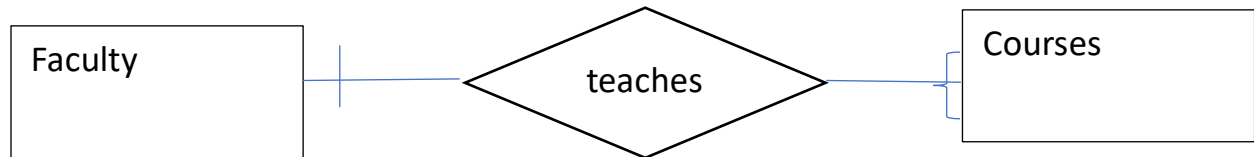
2. Faculty - Department: Many-to-One



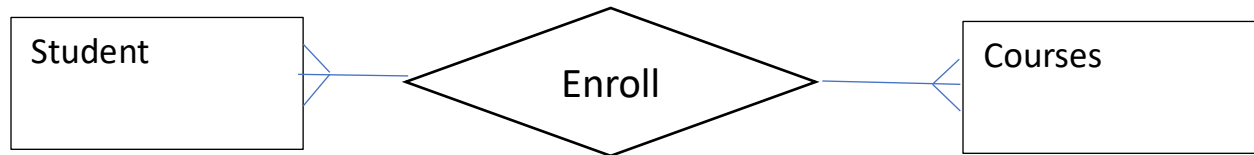
3. Course - Department: Many-to-One



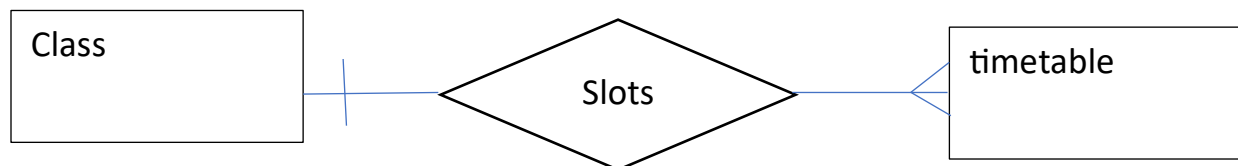
4. Faculty - Class: One-to-Many



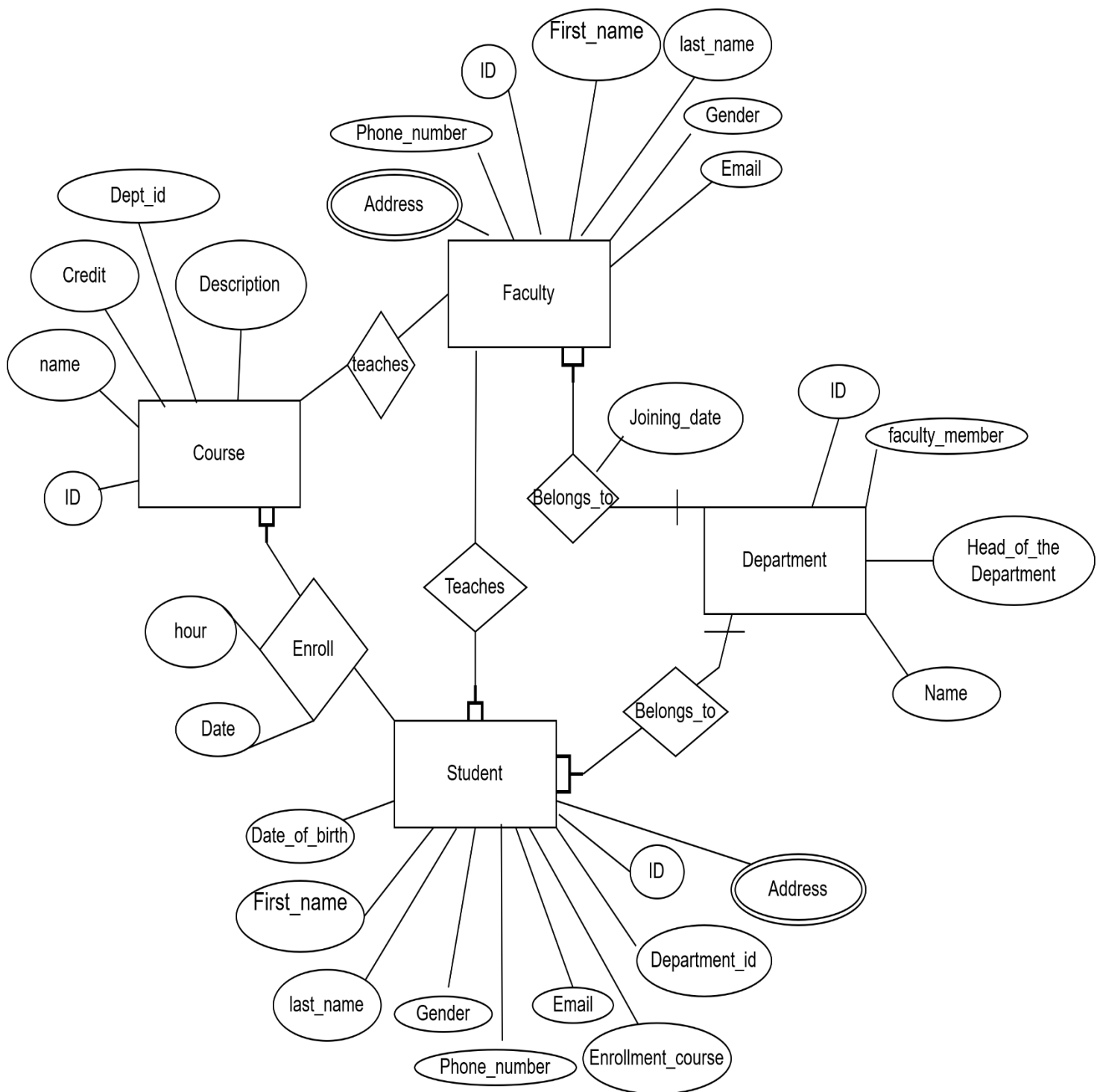
5. Student - Classes: Many-to-Many



6. Class - Timetable: One-to-Many



Step 5: ERD Diagram



Server: 127.0.0.1 » Database: university_management_system

Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers T

Filters

Containing the word:

	Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/>	class	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
<input type="checkbox"/>	course	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
<input type="checkbox"/>	department	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
<input type="checkbox"/>	enrollment	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
<input type="checkbox"/>	faculty	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
<input type="checkbox"/>	student	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
	6 tables	Sum	0	InnoDB	utf8mb4_general_ci	224.0 KiB	0 B

☐ Check all With selected:

Print Data dictionary

Create new table

Table name

Number of columns

Create

Solve No.3

Construct an E-R diagram for a car insurance company whose customers own one or more cars each. Each car has associated with it zero to any number of recorded accidents. Each insurance policy covers one or more cars, and has one or more premium payments associated with it. Each payment is for a particular period of time, and has an associated due date, and the date when the payment was received.

Step 1: Identify the Entities Required

- **Customer:** Customer_id (Primary Key), Name, Address, Phone_number, Email.
- **Car:** Car_ID (Primary Key), License_Plate, Make, Model, Year, CustomerID (Foreign Key to Customer)
- **Accident:** Accident_ID (Primary Key), Date, Location, Description, Car_ID (Foreign Key to Car)
- **Insurance_Policy:** Policy_ID (Primary Key), Policy_Number, Coverage_Details, CustomerID (Foreign Key to Customer)
- **Payment:** Payment_ID (Primary Key), Amount, Period, Due_Date, Payment_Date, Policy_ID (Foreign Key to Insurance_Policy)

Step 3: Identify the Relationship needed

1. Customer can own one or more cars
2. Each accident record is linked to car.
3. Each car can be covered by multiple policies
4. Insurance policy can have multiple premium payments
5. Each payment record is linked to a single policy.

Step 4: Identify the Cardinality Ratio and Participation

- Customer and Car (One-to-Many).
- Car and Accident (One-to-Many).
- Insurance_Policy and Car (Many-to-Many via Car_Policy).
- Insurance_Policy and Payment (One-to-Many).

Step 5: ERD Diagram

