Project Title: Petroleum Company Management System

Team Members:

Mohamed Ashraf Rabie:

ID: 21100854

Email: mohamed.rabie.2022@aiu.edu.eg

Ali Ehab Ali Hassan:

ID: 21100800

Email: ali.hassan.2022@aiu.edu.eg

Ramez Mohamed Fathi:

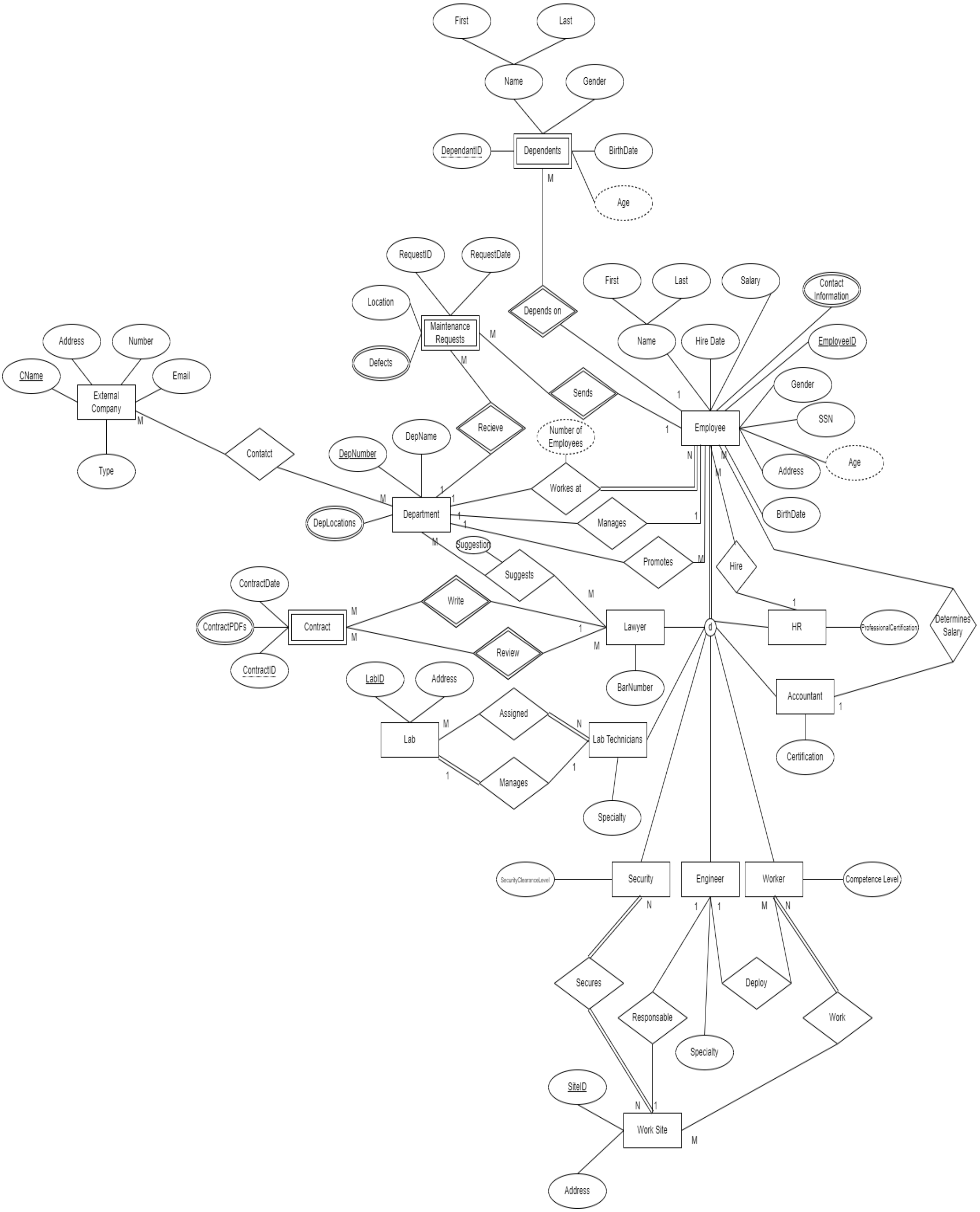
ID: 21100793

Email: ramez.ahmed.2022@aiu.edu.eg

Mohamed Ashraf Mahmoud:

ID: 21100873

Email: mohamed.ashraf.2022@aiu.edu.eg


First
Last
Name
Gender
DependantID
Dependents
BirthDate
M
Age
RequestID
RequestDate
Location
Maintenance Requests
Defects
M
M
Depends on
First
Last
Salary
Contact Information
Name
Hire Date
EmployeeID
Gender
Sends
1
1
Employee
SSN
Age
Address
BirthDate
Address
Number
CName
Email
External Company
M
Type
Contatct
Recieve
Number of Employees
DepNumber
DepName
M
1
1
Workes at
N
M
M
DepLocations
Department
1
1
Manages
1
Suggestion
M
Suggests
Promotes
M
Hire
ContractDate
M
Write
M
Lawyer
d
1
HR
ProfessionalCertification
Determines Salary
ContractPDFs
Contract
M
1
Review
M
ContractID
BarNumber
LabID
Address
Accountant
1
Assigned
M
N
Lab
Lab Technicians
Certification
1
Manages
1
Specialty
SecurityClearanceLevel
Security
Engineer
Worker
Competence Level
N
1
1
M
N
Secures
Deploy
Responsable
Work
Specialty
SiteID
N
1
Work Site
M
Address

Project Description:

Aim: increasing the efficiency of operation in the company

This system is made with international standards and according to the company requirements in the petroleum industry to increase the efficiency of operations and monitoring it and to help employees perform tasks in a better way.

The company wants the system to save the Schedule for every employee including the task distribution and times and operation dates and employee’s info including name, id, and salary and no of dependents, birthdate, contact info and hire date and the want save maintenance history and reports.

This system will not be fully involved in the production process.

Actors:

Employees:

Eng

Workers

Lab Technicians

HR

Security

Lawyer

Accountant

Departments:

Time Control Management Department

Production Department

Financial Department

Industry Development Department

HR Department

Department of Legal Affairs

Maintenance Department

Laboratory Department

Recycling Department

Insurance Department

Transportation Department

External Actors:

Recruitment Agencies

Specialized Tool Companies

Insurance Companies

Transportation Companies

Locations:

Labs

Work Sites

Weak Entities:

Dependents

Contracts

Requests

Business Rules:

- Dependents have (Name, Related Employee ID)
- Work Sites must have (Location ID, Address, Department name)
- Labs must have (Lab ID, Location)
- Contracts have (Contract ID, Contract Date, Contract PDF)
- Maintenance Requests have (Request ID, Request Date, location, Defect)
- All Employees must have (name, address, birth date, contact information, SSN, hire date, salary, gender, age, Dep number)
- Engs, Workers, Lab Technicians, HR Employees, Security Employees, Lawyers, Accountants are Employees.
- An Eng can Deploy many workers under him.
- Many Workers can be deployed under one Eng.
- An Eng may be responsible for a Work site.
- A Worker Can Work in Many Work Sites.
- A Work Site Can Have Many Workers.
- A Lab Technicians May Works in Many Labs.
- A Lab Has Many Lab Technicians.
- A Lab Technician May Manage only one Lab.
- A Lab can only be managed by one Lab Technician
- A Security Employee must be assigned to a location.
- An HR Employee can Hire many Employees.
- An Employee is Hired by an HR Employee.
- A Department can Request a Lawyers Suggestion.

- A Lawyer can Review Many Contracts.
- A Contract can be Reviewed by Many Lawyers.
- A Lawyer can Write Many Contracts.
- Only one Lawyer can Write a contract.
- An Accountant Assigns many Employees salary.
- An employees’ salaries is set by an Accountant.
- All Departments have (Dep number, Dep name, Dep locations, Dep manager, Dep number of employees)
- An Employee can Send Many Maintenance Requests to the Maintenance Department.
- The Maintenance Department can receive many Maintenance Requests from the Employees.
- All External Companies have (Company Name, Company Number, Company Email, type)
- Recruitment Agencies, Specialized Tool Companies, Insurance Companies, Transportation Companies are All types of external companys.
- A Department Can Send a Request to an External Company Each request has An ID and an Attached PDF of the request.
- Employees can Have Dependents Under their insurance plan.
- All Employees belong to a department.
- One Employee manages only one department, each department can only be managed by a single employee.
- One Eng can Assign many Workers a task.
- Departments can contact each other.
- Departments can promote Employees under them.

Potential Queries:

- Which Employees have a salary above (Set amount)
- Which Employees Work at (Set Location)
- Which employees work in department number (enter department number)
- Which Employees have an age above (enter number)
- Who is the manager of department (enter department number)
- What are the insurance companies we are collaborating with
- What are the transportation companies we are collaborating with
- Which Employees are under (Set Department)
- Were there any Maintenance requests during (Set Date)
- Which Employees have Dependents
- Which Employees have more than (set amount) Dependents
- Which Employees have less than (set amount) Dependents
- Who are the Dependents of (set Employee)

Initial Relational Schima:

Employee (Employee ID, FirstName, LastName, Hire Date, Salary, Gender, SSN, Address, Birthdate, Accountant ID, Employer ID, Position, Department Number)

Employee ID → FirstName

Employee ID → Last Name

Employee ID → Gender

Employee ID → SSN

Employee ID → Address

Employee ID → Birth Date

Employee ID → Employer

Employee ID → Position

Employee ID → Accountant ID

Employee ID → Salary

Accountant ID → Salary

Employer ID → Hire Date

Department Number → Position

Employee (Employee ID, FirstName, LastName, Hire Date, Salary, Gender, SSN, Address, Birthdate, Accountant ID, Employer ID, Position, Department Number)

Transformed into 3NF:

Employee Salary (Employee ID, Accountant ID, Salary)

Hired Employee (Employee ID, Employer ID, Hire Date)

Employee Position (Employee ID, Department Number, Position)

Employee (Employee ID, FirstName, LastName, Salary, Gender, SSN, Address, Birthdate)

Contact Information → Employee ID

Contact Information (Employee ID, Contact information)

Is already in 3NF.

Dependent (Dependent ID, Employee ID, Birthdate, Gender, FirstName, LastName)

Dependent ID → Employee ID

Dependent ID → Birthdate

Dependent ID → Gender

Dependent ID → FirstName

Dependent ID → LastName

Dependent ID → Employee ID

Dependent (Dependent ID, Employee ID, Birthdate, Gender, FirstName, LastName)

Is already 3NF.

Employee ID → Professional Certification

HR (Employee ID, Professional Certification)

Is already 3NF.

Employee ID → Certification

Accountant (Employee ID, Certification)

Is already 3NF.

Employee ID → Engineer ID

Employee ID → Competence Level

Worker (Employee ID, Competence Level, Engineer ID)

Is already 3NF.

Worker ID → Site ID

Worker Working in Work Site (Worker ID, Site ID)

Is already 3NF.

Employee ID → Specialization

Engineer (Employee ID, Specialization)

Is already 3NF.

Employee ID → Security Clearance Level

Security (Employee ID, Security Clearance Level)

Is already 3NF.

Security ID → Site ID

Security Securing Worksite (Security ID, Site ID)

Is already 3NF.

Site ID → Address

Site ID → Responsible Engineer ID

Worksite (Site ID, Address, Responsible Engineer ID)

Is already 3NF.

Employee ID → Specialization

Lab Technicians (Employee ID, Specialization)

Is already 3NF.

Lab Technicians Assigned Lab (Lab Technician ID, Lab ID)

Is already 3NF.

Lab ID → Address

Lab ID → Lab Manager

Lab (Lab ID, Address, Lab Manager)

Is already 3NF.

Employee ID → Bar Number

Lawyer (Employee ID, Bar Number)

Is already 3NF.

Lawyer ID → Suggestion

Department ID → Suggestion

Lawyer Suggestion (Lawyer ID, Department ID, Suggestion)

Is already 3NF.

Contract ID → Lawyer ID

Contract Review (Lawyer ID, Contract ID)

Is already 3NF.

Contract ID → Lawyer ID

Contract ID → Contract Date

Contract (Contract ID, Lawyer ID, Contract Date)

Is already 3NF.

Contract ID → Contract PDF

Contract PDF (Contract ID, Contract PDF)

Is already 3NF.

Department Number → Department Name

Department Number → Manager

Department (Department Number, Department Name, Manager)

Is already 3NF.

Department Number → Location

Department Location (Department Number, Location)

Is already 3NF.

Maintenance Request (Request ID, Location, Request Date, Department ID, Employee ID)

Request ID → Location

Request ID → Request Date

Employee ID → Request Date

Employee ID → Location

Request ID → Department ID

Maintenance Request (Request ID, Location, Request Date, Department ID, Employee ID)

Transformed into 3NF:

Employee Request (Employee ID, Request ID, Location, Request Date)

Maintenance Request (Request ID, Employee ID, Department ID)

Defect → Request ID

Defects in Maintenance Request (Request ID, Defect)

Is already 3NF.

Department Number → Company Name

Department Contacting External Company (Company Name, Department Number)

Is already 3NF.

Company Name → Address

Company Name → Number

Company Name → Email

Company Name → Type

External Company (Company Name, Address, Number, Email, Type)

Is already 3NF.

Final Schema is:

-Employee Salary (Employee ID, Accountant ID, Salary)

Accountant ID is a foreign Key for Accountant.

-Hired Employee (Employee ID, Employer ID, Hire Date)

Employer ID is a foreign Key for HR Employee.

-Employee Position (Employee ID, Department Number, Position)

Department Number is a foreign Key for Department.

-Employee (Employee ID, FirstName, LastName, Salary, Gender, SSN, Address, Birthdate)

Contact Information (Employee ID, Contact information)

-Dependent (Dependent ID, Employee ID, Birthdate, Gender, FirstName, LastName)

Employee ID is a foreign Key for Employee.

-HR (Employee ID, Professional Certification)

-Accountant (Employee ID, Certification)

-Worker (Employee ID, Competence Level, Engineer ID)

Engineer ID is a foreign Key for Engineer.

-Worker Working in Work Site (Worker ID, Site ID)

-Engineer (Employee ID, Specialization)

-Security (Employee ID, Security Clearance Level)

-Security Securing Worksite (Security ID, Site ID)

-Worksite (Site ID, Address, Responsible Engineer ID)

Responsible Engineer ID is a foreign Key for Engineer.

-Lab Technicians (Employee ID, Specialty)

-Lab Technicians Assigned Lab (Lab Technician ID, Lab ID)

-Lab (Lab ID, Address, Lab Manager)

Lab Manager is foreign Key for Lab Technician.

-Lawyer (Employee ID, Bar Num)

-Lawyer Suggestion (Lawyer ID, Department ID, Suggestion)

-Contract Review (Lawyer ID, Contract ID)

-Contract (Contract ID, Lawyer ID, Contract Date)

Lawyer ID is a foreign Key for Lawyer

-Contract PDF (Contract ID, Contract PDF)

-Department (Department Number, Department Name, Manager)

Manager is a foreign key for Employee.

-Department Location (Department Number, Location)

-Employee Request (Employee ID, Request ID, Location, Request Date)

-Maintenance Request (Request ID, Employee ID, Department ID)

-Defects in Maintenance Request (Request ID, Defect)

-Department Contacting External Company (Company Name, Department Number)

-External Company (Company Name, Address, Number, Email, Type)

```
create database if not exists company ;
use company;


Create table if not exists Employee(
EmployeeID int primary key,
fristname varchar(20) not null,
lastname  varchar(20) not null,
Gender varchar(6) not null,
SNN int unique not null,
Address varchar(150) not null,
Birthdate date not null
);


INSERT IGNORE INTO Employee (EmployeeID, fristname, lastname, Gender, SNN, Address, Birthdate)
VALUES
(1,'ahmed','mohamed','male',10001,'alamein university','29-12-04'),
(2,'hussein','ahmed','male',10002,'alamein university','2-12-04'),
(3,'ahmed','abdullah','male',10003,'alamein university','9-1-04'),
(4,'mohamed','ibrahim','male',10004,'alamein university','10-2-04'),
(5,'youssra','mohamed','female',10005,'alamein university','2-10-04'),
(6,'nermin','essam','female',10006,'alamein university','15-9-04'),
(7, 'William', 'Taylor', 'Male', 10007, '456 Oak St', '1993-08-30'),
(8, 'Sophia', 'Anderson', 'Female', 10008, '789 Pine St', '1990-02-14'),
(9, 'Matthew', 'Martinez', 'Male', 10009, '321 Elm St', '1994-07-27'),
(10, 'Ava', 'Hernandez', 'Female', 10010, '654 Main St', '1992-01-10'),
(11, 'Jacob', 'Garcia', 'Male', 10011, '987 Cedar St', '1991-03-23'),
(12, 'Isabella', 'Lopez', 'Female', 10012, '123 Oak St', '1989-06-06'),
(13, 'Ethan', 'Wilson', 'Male', 10013, '456 Pine St', '1993-10-19'),
(14, 'Mia', 'Lee', 'Female', 10014, '789 Elm St', '1990-04-01'),
(15, 'Alexander', 'Clark', 'Male', 10015, '321 Main St', '1995-07-14'),
(16, 'Olivia', 'Hall', 'Female', 10016, '654 Cedar St', '1992-11-27'),

(17, 'James', 'Young', 'Male', 10017, '987 Oak St', '1991-05-10'),

(18, 'Sofia', 'Walker', 'Female', 10018, '123 Pine St', '1994-09-23'),

(19, 'Logan', 'Harris', 'Male', 10019, '456 Elm St', '1993-01-06'),

(20, 'Charlotte', 'Gonzalez', 'Female', 10020, '789 Main St', '1990-03-20'),

(21, 'Benjamin', 'Allen', 'Male', 10021, '321 Oak St', '1995-06-02'),

(22, 'Amelia', 'Perez', 'Female', 10022, '654 Pine St', '1989-09-15'),

(23, 'Mila', 'Parker', 'Female', 10023, '789 Cedar St', '1990-07-29'),

(24, 'Emma', 'Turner', 'Female', 10024, '123 Elm St', '1994-12-28'),

(25, 'Henry', 'Baker', 'Male', 10025, '456 Main St', '1991-04-10'),

(26, 'Grace', 'Hill', 'Female', 10026, '789 Cedar St', '1990-08-23'),

(27, 'Sebastian', 'Ward', 'Male', 10027, '321 Oak St', '1995-01-05'),

(28, 'Scarlett', 'Price', 'Female', 10028, '654 Pine St', '1992-03-18'),

(29, 'Jack', 'Foster', 'Male', 10029, '987 Elm St', '1994-06-30'),

(30, 'Lily', 'Brooks', 'Female', 10030, '123 Main St', '1990-11-12'),

(31, 'Owen', 'Kelly', 'Male', 10031, '456 Elm St', '1993-02-25'),

(32, 'Victoria', 'Coleman', 'Female', 10032, '789 Oak St', '1989-07-08'),

(33, 'Gabriel', 'Gomez', 'Male', 10033, '321 Pine St', '1995-10-21'),

(34, 'Nora', 'Simmons', 'Female', 10034, '654 Cedar St', '1991-02-03'),

(35, 'Carter', 'Hughes', 'Male', 10035, '987 Oak St', '1990-05-16'),

(36, 'Hazel', 'Rivera', 'Female', 10036, '123 Pine St', '1994-08-29'),

(37, 'Daniel', 'Ward', 'Male', 10037, '456 Elm St', '1993-12-11'),

(38, 'Luna', 'Reed', 'Female', 10038, '789 Main St', '1990-03-25'),

(39, 'Max', 'Cooper', 'Male', 10039, '321 Oak St', '1995-06-07'),

(40, 'Stella', 'Ross', 'Female', 10040, '654 Pine St', '1989-09-20'),

(41, 'Elijah', 'Barnes', 'Male', 10041, '123 Elm St', '1994-12-03'),

(42, 'Avery', 'Wood', 'Female', 10042, '456 Main St', '1991-03-16');



CREATE TABLE IF NOT EXISTS Accountant (

    EmployeeID INT PRIMARY KEY,
  Certification VARCHAR(100),
  FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID) ON DELETE CASCADE
);
INSERT IGNORE INTO Accountant (EmployeeID, Certification)
VALUES (1, 'Certification1'),
    (37, 'Certification2'),
    (36, 'Certification3'),
    (35, 'Certification4'),
    (34, 'Certification5'),
    (33, 'Certification6');



create table if not exists Employee_Salary(
EmployeeID int primary key,
AccountantID int,
salary float,
foreign key(EmployeeID) references Employee(EmployeeID) ON DELETE CASCADE,
foreign key(AccountantID) references Accountant(EmployeeID) ON DELETE SET NULL
);


INSERT IGNORE INTO Employee_Salary (EmployeeID, AccountantID, salary)
VALUES
(1, 1, 5000.00),
(2, 37, 6000.00),
(3, 36, 5500.00),
(4, 35, 5200.00),
(5, 34, 4800.00),
(6, 33, 5100.00),
(7, 1, 5200.00),
(8, 37, 5500.00),
(9, 36, 5300.00),

(10, 35, 5800.00),

(11, 34, 5200.00),

(12, 33, 5000.00),

(13, 1, 5400.00),

(14, 37, 5200.00),

(15, 36, 5100.00),

(16, 35, 5500.00),

(17, 34, 5600.00),

(18, 33, 5300.00),

(19, 1, 5800.00),

(21, 36, 5200.00),

(22, 35, 5300.00),

(23, 34, 5400.00),

(24, 33, 5500.00),

(25, 1, 5200.00),

(26, 37, 4800.00),

(27, 36, 5100.00),

(28, 35, 5200.00),

(29, 34, 5000.00),

(30, 33, 5300.00),

(31, 1, 5200.00),

(32, 1, 5500.00),

(33, 1, 5300.00),

(34, 1, 5800.00),

(35, 1, 5200.00),

(36, 1, 5000.00),

(37, 1, 5400.00),

(38, 37, 5200.00),

(39, 36, 5100.00),

(40, 35, 5500.00),

(41, 34, 5600.00),
(42, 33, 5300.00);


create table if not exists HR (
EmployeeID INT primary key,
foreign key(EmployeeID) references Employee(EmployeeID) ON DELETE CASCADE
);
INSERT IGNORE INTO HR (EmployeeID)
VALUES (2),(38),(39),(40),(41),(42);

create table if not exists HiredEmployee(
EmployeeID int primary key,
HireDate date not null,
HR_hiredby_ID int ,
foreign key(EmployeeID) references Employee(EmployeeID) ON DELETE CASCADE,
foreign key(HR_hiredby_ID) references HR(EmployeeID) ON DELETE SET NULL
);
INSERT IGNORE INTO HiredEmployee (EmployeeID, HireDate, HR_hiredby_ID)
VALUES
(1, '2023-01-01', 2),
(2, '2023-01-02', 2),
(3, '2023-01-03', 39),
(4, '2023-01-04', 40),
(5, '2023-01-05', 41),
(6, '2023-01-06', 42),
(7, '2023-01-07', 2),
(8, '2023-01-08', 38),
(9, '2023-01-09', 39),
(10, '2023-01-10', 40),
(11, '2023-01-11', 41),
(12, '2023-01-12', 42),
(13, '2023-01-13', 2),
(14, '2023-01-14', 38),
(15, '2023-01-15', 39),
(16, '2023-01-16', 40),
(17, '2023-01-17', 41),
(18, '2023-01-18', 42),
(19, '2023-01-19', 2),
(20, '2023-01-20', 38),
(21, '2023-01-21', 39),
(22, '2023-01-22', 40),
(23, '2023-01-23', 41),
(24, '2023-01-24', 42),
(25, '2023-01-25', 2),
(26, '2023-01-26', 38),
(27, '2023-01-27', 39),
(28, '2023-01-28', 40),
(29, '2023-01-29', 41),
(30, '2023-01-30', 42),
(31, '2023-01-31', 2),
(32, '2023-02-01', 38),
(33, '2023-02-02', 39),
(34, '2023-02-03', 40),
(35, '2023-02-04', 41),
(36, '2023-02-05', 42),
(37, '2023-02-06', 2),
(38, '2023-02-07', 2),
(39, '2023-02-08', 2),
(40, '2023-02-09', 2),
(41, '2023-02-10', 2),
(42, '2023-02-11', 2);


create table if not exists Department (
```

```sql
Departmentno int primary key,
Dapartmentname varchar(30) not null,
Manger_ID int,
foreign key(Manger_ID) references Employee(EmployeeID) ON DELETE SET NULL
);
INSERT IGNORE INTO Department (Departmentno, Dapartmentname, Manger_ID)
VALUES
(101, 'Financial', 1),
(201, 'Production', 3),
(301, 'HR', 2),
(401, 'Legal Affairs', 7),
(501, 'Maintenance', 5),
(601, 'Laboratory', 6),
(701, 'Security', 4);

create table if not exists Employee_postion (
EmployeeID int primary key,
Position varchar(30) not null,
Department_no int,
foreign key(EmployeeID) references Employee(EmployeeID) ON DELETE CASCADE,
foreign key(Department_no) references Department(Departmentno) ON DELETE SET NULL
);
INSERT IGNORE INTO employee_postion (EmployeeID, Position, Department_no)
VALUES
-- Head of Financial Department
(1, 'Financial Department Head', 101),
-- Head of HR Department
(2, 'HR Department Head', 301),
-- Head of Production Department
(3, 'Production Department Head', 201),
-- Head of Security Department
(4, 'Security Department Head', 701),
-- Head of Maintenance Department
(5, 'Maintenance Department Head', 501),
-- Head of Laboratory Department
(6, 'Laboratory Department Head', 601),
-- Head of Legal Affairs Department
(7, 'Legal Affairs Department Head', 401),
-- HR Employees in HR Department
(38, 'HR', 301),
(39, 'HR', 301),
(40, 'HR', 301),
(41, 'HR', 301),
(42, 'HR', 301),
-- Accountants in Financial Department
(37, 'Accountant', 101),
(36, 'Accountant', 101),
(35, 'Accountant', 101),
(34, 'Accountant', 101),
(33, 'Accountant', 101),
-- Engineers in Production Department
(32, 'Engineer', 201),
(31, 'Engineer', 201),
(30, 'Engineer', 201),
(29, 'Engineer', 201),
(28, 'Engineer', 201),
-- Security Workers in Security Department
(27, 'Security Worker', 701),
(26, 'Security Worker', 701),
(25, 'Security Worker', 701),
(24, 'Security Worker', 701),
(23, 'Security Worker', 701),
-- Workers in Maintenance Department
(22, 'Worker', 501),
(21, 'Worker', 501),
(20, 'Worker', 501),
(19, 'Worker', 501),
(18, 'Worker', 501),
-- Lawyers in Legal Affairs Department
(12, 'Lawyer', 401),
(11, 'Lawyer', 401),
(10, 'Lawyer', 401),
(9, 'Lawyer', 401),
(8, 'Lawyer', 401),
-- Lab Technicians in Laboratory Department
(17, 'Lab Technician', 601),
(16, 'Lab Technician', 601),
(15, 'Lab Technician', 601),
(14, 'Lab Technician', 601),
(13, 'Lab Technician', 601);

create table if not exists Contact_information(
EmployeeID int not null,
Contact_information varchar(200),
primary key(EmployeeID  ,Contact_information ),
foreign key(EmployeeID) references Employee(EmployeeID) ON DELETE CASCADE
);
INSERT IGNORE INTO Contact_information (EmployeeID, Contact_information)
VALUES
-- Employees with 3 contact information entries
(1, 'employee1@example.com'),
(1, '123-456-7890'),
(1, '123 Main St'),
(2, 'employee2@example.com'),
(2, '456-789-0123'),
(2, '456 Elm St'),
(3, 'employee3@example.com'),
(3, '789-012-3456'),
(3, '789 Oak Ave'),
-- Employees with 2 contact information entries
(4, 'employee4@example.com'),
(4, '234-567-8901'),
(5, 'employee5@example.com'),
(5, '567-890-1234'),
(6, 'employee6@example.com'),
(6, '890-123-4567'),
-- Employees with 1 contact information entry
(7, 'employee7@example.com'),
(8, 'employee8@example.com'),
(9, 'employee9@example.com'),
(10, 'employee10@example.com'),
(11, 'employee11@example.com'),
(12, 'employee12@example.com'),
(13, 'employee13@example.com'),
(14, 'employee14@example.com'),
(15, 'employee15@example.com'),
(16, 'employee16@example.com'),
(17, 'employee17@example.com'),
(18, 'employee18@example.com'),
(19, 'employee19@example.com'),
(20, 'employee20@example.com'),
(21, 'employee21@example.com'),
(22, 'employee22@example.com'),
(23, 'employee23@example.com'),
(24, 'employee24@example.com'),
(25, 'employee25@example.com'),
(26, 'employee26@example.com'),
(27, 'employee27@example.com'),
(28, 'employee28@example.com'),
(29, 'employee29@example.com'),
(30, 'employee30@example.com'),
(31, 'employee31@example.com'),
(32, 'employee32@example.com'),
(33, 'employee33@example.com'),
(34, 'employee34@example.com'),
(35, 'employee35@example.com'),
(36, 'employee36@example.com'),
(37, 'employee37@example.com'),
(38, 'employee38@example.com'),
(39, 'employee39@example.com'),
(40, 'employee40@example.com'),
(41, 'employee41@example.com'),
(42, 'employee42@example.com');


create table if not exists Dependent (
Dependent int primary key,
Birthdate date not null,
fristname varchar(20) not null,
lastname  varchar(20) not null,
Gender varchar(6) not null,
EmployeeID int not null,
foreign key(EmployeeID) references Employee(EmployeeID) ON DELETE CASCADE
);
INSERT IGNORE INTO Dependent (Dependent, Birthdate, fristname, lastname, Gender, EmployeeID)
VALUES
-- Employees with 3 dependents
(1, '2000-01-01', 'John', 'Doe', 'Male', 1),
(2, '2005-02-03', 'Emily', 'Smith', 'Female', 1),
(3, '2010-06-10', 'Daniel', 'Johnson', 'Male', 1),
(4, '2001-03-15', 'Sophia', 'Anderson', 'Female', 2),
(5, '2007-07-20', 'Oliver', 'Wilson', 'Male', 2),
(6, '2014-09-25', 'Isabella', 'Taylor', 'Female', 2),
-- Employees with 2 dependents
(7, '2002-05-12', 'Mason', 'Brown', 'Male', 3),
(8, '2008-09-18', 'Emma', 'Jones', 'Female', 3),
(9, '2004-11-07', 'Michael', 'Clark', 'Male', 4),
(10, '2011-12-30', 'Ava', 'Martinez', 'Female', 4),
-- Employees with 1 dependent
(11, '2003-08-14', 'William', 'Harris', 'Male', 5),
(12, '2009-10-22', 'Sophie', 'Lee', 'Female', 6);

CREATE TABLE IF NOT EXISTS Engineer (
    EmployeeID INT PRIMARY KEY,
    Specialty VARCHAR(50) not null,
    FOREIGN KEY (EmployeeID) REFERENCES Employee (EmployeeID) ON DELETE CASCADE
);

INSERT IGNORE INTO Engineer (EmployeeID, Specialty) VALUES
    (3, 'Specialty1'),
    (32, 'Specialty2'),
    (31, 'Specialty3'),
    (30, 'Specialty4'),
    (29, 'Specialty5'),
    (28, 'Specialty6');

CREATE TABLE IF NOT EXISTS Security_worker (
    EmployeeID INT PRIMARY KEY,
    SecurityClearanceLevel VARCHAR(50),
    FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID) ON DELETE CASCADE
);
```

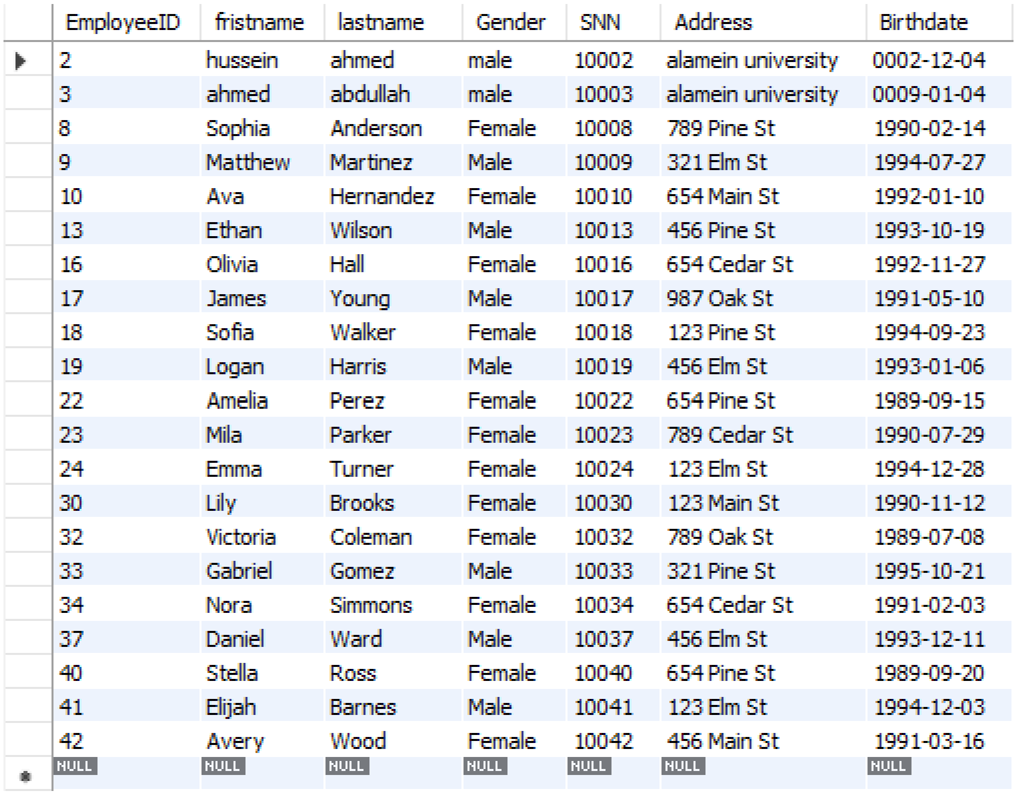
```
INSERT IGNORE INTO Security_worker (EmployeeID, SecurityClearanceLevel)
VALUES
    (4, 'Confidential'),
    (27, 'Secret'),
    (26, 'Top Secret'),
    (25, 'Confidential'),
    (24, 'Secret'),
    (23, 'Top Secret');


CREATE TABLE IF NOT EXISTS Worker (
    EmployeeID INT PRIMARY KEY,
    EngineerID INT,
    ConfidenceLevel INT,
    FOREIGN KEY (EngineerID) REFERENCES Engineer(EmployeeID) ON DELETE SET NULL,
    FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID) ON DELETE CASCADE
);
INSERT IGNORE INTO worker (EmployeeID, EngineerID, ConfidenceLevel)
VALUES
    (5, 3, 100),
    (22, 32, 90),
    (21, 31, 75),
    (20, 30, 85),
    (19, 29, 70),
    (18, 28, 95);



create table if not exists Worksite (
SiteID int primary key ,
Address varchar(150) not null,
Responsible_Engineer_ID int,
foreign key(Responsible_Engineer_ID) references Engineer(EmployeeID) ON DELETE SET NULL
);
INSERT IGNORE INTO Worksite (SiteID, Address, Responsible_Engineer_ID)
VALUES
(1, '123 Main St, City A', 3),
(2, '456 Elm St, City B', 32),
(3, '789 Oak St, City C', 31),
(4, '321 Pine St, City D', 30),
(5, '654 Maple St, City E', 29),
(6, '987 Cedar St, City F', 28);

create table if not exists Worker_Working_inWork_Site (
WorkerID int not null,
siteID int not null,
primary key(WorkerID, siteID),
foreign key(WorkerID) references worker(EmployeeID) ON DELETE CASCADE,
foreign key(siteID) references Worksite(SiteID) ON DELETE CASCADE
);
INSERT IGNORE INTO Worker_Working_inWork_Site (WorkerID, siteID)
VALUES
(5, 1),
(22, 1),
(21, 1),
(20, 1),
(19, 1),
(18, 1);


create table if not exists Security_Securing_Worksite (
SecurityID int not null,
siteID int not null,
primary key(SecurityID, siteID),
foreign key(SecurityID) references Security_worker(EmployeeID) ON DELETE CASCADE,
foreign key(siteID) references Worksite(SiteID) ON DELETE CASCADE
);
INSERT IGNORE INTO Security_Securing_Worksite (SecurityID, siteID)
VALUES
(4, 1),
(27, 1),
(26, 1),
(25, 1),
(24, 1),
(23, 1);


CREATE TABLE IF NOT EXISTS Lawyer (
   EmployeeID INT PRIMARY KEY,
   BarNum VARCHAR(50),
   FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID) ON DELETE CASCADE
);
INSERT IGNORE INTO Lawyer (EmployeeID, BarNum)
VALUES
   (7, 'BAR123'),
   (12, 'BAR456'),
   (11, 'BAR789'),
   (10, 'BAR012'),
   (9, 'BAR345'),
   (8, 'BAR678');



CREATE TABLE if not exists lab (
  lab_id INT PRIMARY KEY,
  Address VARCHAR(150) NOT NULL,
  LabManger_ID INT
);



CREATE TABLE IF NOT EXISTS Lab_Technicians (
  EmployeeID INT PRIMARY KEY,
  Lab_id INT,
  Specialty VARCHAR(100),
  FOREIGN KEY (EmployeeID) REFERENCES Employee (EmployeeID) ON DELETE CASCADE,
  FOREIGN KEY (Lab_id) REFERENCES lab (lab_id) ON DELETE SET NULL
);



ALTER TABLE lab
ADD FOREIGN KEY (LabManger_ID) REFERENCES Lab_Technicians (EmployeeID) ON DELETE SET NULL;

INSERT IGNORE INTO lab values (1,'alamein university 10',null);
INSERT IGNORE INTO Lab_Technicians (EmployeeID, Lab_id, Specialty)
VALUES
    (6, 1, 'Microbiology'),
    (17, 1, 'Chemistry'),
    (16, 1, 'Pathology'),
    (15, 1, 'Genetics'),
    (14, 1, 'Immunology'),
    (13, 1, 'Hematology');

UPDATE lab
SET LabManger_ID = 6
WHERE lab_id = 1;



create table if not exists Lawyer_Suggestion(
Lawyer_ID int not null,
DepartmentID int not null,
Suggestion varchar(5000),
primary key(Lawyer_ID, DepartmentID),
foreign key(Lawyer_ID) references Lawyer(EmployeeID) ON DELETE CASCADE,
foreign key(DepartmentID) references Department(Departmentno) ON DELETE CASCADE
);
INSERT IGNORE INTO Lawyer_Suggestion (Lawyer_ID, DepartmentID, Suggestion)
VALUES
(8, 101, 'Implement stricter financial regulations.'),
(9, 201, 'Improve production efficiency through automation.'),
(10, 301, 'Enhance employee training and development programs.'),
(11, 401, 'Review and update legal contracts and agreements.'),
(12, 501, 'Increase preventive maintenance for equipment.'),
(7, 601, 'Upgrade laboratory equipment for better research capabilities.');


create table if not exists Contract (
Contract_ID int primary key ,
Contract_Date date not null,
Contract_Details varchar(2000),
Lawyer_ID int ,
foreign key(Lawyer_ID) references Lawyer(EmployeeID) ON DELETE SET NULL
);
INSERT IGNORE INTO Contract (Contract_ID, Contract_Date, Contract_Details, Lawyer_ID)
VALUES
(101, '2023-01-15', 'Supply agreement with ABC Company', 8),
(102, '2023-02-10', 'Service contract with XYZ Corporation', 9),
(103, '2023-03-22', 'Lease agreement for property rental', 10),
(104, '2023-04-05', 'Consulting services contract with DEF Inc.', 11),
(105, '2023-05-18', 'Construction agreement for project A', 12),
(106, '2023-06-30', 'Software licensing contract with GHI Corporation', 7);
```

```
create table if not exists contract_reveiw(
Lawyer_ID int ,
Contract_ID int ,
primary key (Lawyer_ID ,Contract_ID),
foreign key(Lawyer_ID) references Lawyer(EmployeeID) ON DELETE CASCADE,
foreign key(Contract_ID) references Contract(Contract_ID) ON DELETE CASCADE
);
INSERT IGNORE INTO contract_reveiw (Lawyer_ID, Contract_ID)
VALUES
(8, 101),
(9, 102),
(10, 103),
(11, 104),
(12, 105),
(7, 106);


create table if not exists Department_location(
Departmentno int,
DepartmentLocation varchar(500),
primary key(Departmentno ,DepartmentLocation),
foreign key(Departmentno) references Department(Departmentno) ON DELETE CASCADE
);
INSERT IGNORE INTO Department_location (Departmentno, DepartmentLocation)
VALUES
(101, 'Location 1A'),
(101, 'Location 1B'),
(101, 'Location 1C'),
(201, 'Location 2A'),
(301, 'Location A1'),
(301, 'Location B1'),
(401, 'Location C1'),
(501, 'Location X'),
(601, 'Location Y'),
(701, 'Location Z');


create table if not exists Employee_Request(
EmployeeID int not null,
RequestID int not null,
requestDetails varchar(2000) not null,
RequestDate date not null,
primary key(EmployeeID,RequestID),
foreign key(EmployeeID) references Employee(EmployeeID) ON DELETE CASCADE
);
INSERT IGNORE INTO Employee_Request (EmployeeID, RequestID, requestDetails, RequestDate) VALUES
-- Employee 12 - 3 requests
(12, 1, 'Request details 1', '2023-05-01'),
(12, 2, 'Request details 2', '2023-05-02'),
(12, 3, 'Request details 3', '2023-05-03'),
-- Employee 17 - 2 requests
(17, 4, 'Request details 4', '2023-05-04'),
(17, 5, 'Request details 5', '2023-05-05'),
-- Employees 22-42 - 1 request each
(22, 6, 'Request details 6', '2023-05-06'),
(27, 7, 'Request details 7', '2023-05-07'),
(32, 8, 'Request details 8', '2023-05-08'),
(37, 9, 'Request details 9', '2023-05-09'),
(42, 10, 'Request details 10', '2023-05-10');


create table if not exists Maintenance_Request(
RequestID int not null,
EmployeeID int not null,
requestDetails varchar(2000),
Departmentno int,
primary key(RequestID, EmployeeID, Departmentno),
foreign key(EmployeeID) references Employee(EmployeeID) ON DELETE CASCADE,
foreign key(Departmentno) references Department(Departmentno) ON DELETE CASCADE
);
INSERT IGNORE INTO Maintenance_Request (RequestID, EmployeeID, requestDetails, Departmentno) VALUES
(1, 12, 'Request details 1', 501),
(2, 12, 'Request details 2', 501),
(3, 12, 'Request details 3', 501),
(4, 17, 'Request details 4', 501),
(5, 17, 'Request details 5', 501),
(6, 22, 'Request details 6', 501),
(7, 27, 'Request details 7', 501),
(8, 32, 'Request details 8', 501),
(9, 37, 'Request details 9', 501),
(10, 42, 'Request details 10', 501);

create table if not exists Defects_in_Maintenance_Request(
RequestID int not null,
Defect varchar(500) not null,
primary key(RequestID, Defect),
foreign key(RequestID) references Maintenance_Request(RequestID) ON DELETE CASCADE
);
INSERT IGNORE INTO Defects_in_Maintenance_Request (RequestID, Defect) VALUES
(1, 'Defect 1'),
(2, 'Defect 2'),
(3, 'Defect 3'),
(4, 'Defect 4'),
(5, 'Defect 5'),
(6, 'Defect 6'),
(7, 'Defect 7'),
(8, 'Defect 8'),
(9, 'Defect 9'),
(10, 'Defect 10');


CREATE TABLE IF NOT EXISTS External_Company (
    Company_ID INT PRIMARY KEY,
    Company_name VARCHAR(100),
    Address VARCHAR(100),
    Num VARCHAR(20),
    Email VARCHAR(100),
    Type VARCHAR(50)
);

INSERT IGNORE INTO External_Company (Company_ID, Company_name, Address, Num, Email, Type) VALUES
    (1, 'Company 1', 'Address 1', '123456789', 'company1@example.com', 'Recruitment Agency'),
    (2, 'Company 2', 'Address 2', '987654321', 'company2@example.com', 'Recruitment Agency'),
    (3, 'Company 3', 'Address 3', '567891234', 'company3@example.com', 'Recruitment Agency'),
    (4, 'Company 4', 'Address 4', '321456789', 'company4@example.com', 'Recruitment Agency'),
    (5, 'Company 5', 'Address 5', '678912345', 'company5@example.com', 'Recruitment Agency'),
    (6, 'Company 6', 'Address 6', '234567891', 'company6@example.com', 'Recruitment Agency'),
    (7, 'Company 7', 'Address 7', '789123456', 'company7@example.com', 'Specialized Tool Company'),
    (8, 'Company 8', 'Address 8', '456789123', 'company8@example.com', 'Specialized Tool Company'),
    (9, 'Company 9', 'Address 9', '912345678', 'company9@example.com', 'Specialized Tool Company'),
    (10, 'Company 10', 'Address 10', '345678912', 'company10@example.com', 'Specialized Tool Company'),
    (11, 'Company 11', 'Address 11', '891234567', 'company11@example.com', 'Specialized Tool Company'),
    (12, 'Company 12', 'Address 12', '678912345', 'company12@example.com', 'Specialized Tool Company'),
    (13, 'Company 13', 'Address 13', '123456789', 'company13@example.com', 'Insurance Company'),
    (14, 'Company 14', 'Address 14', '987654321', 'company14@example.com', 'Insurance Company'),
    (15, 'Company 15', 'Address 15', '567891234', 'company15@example.com', 'Insurance Company'),
    (16, 'Company 16', 'Address 16', '321456789', 'company16@example.com', 'Insurance Company'),
    (17, 'Company 17', 'Address 17', '678912345', 'company17@example.com', 'Insurance Company'),
    (18, 'Company 18', 'Address 18', '234567891', 'company18@example.com', 'Insurance Company'),
    (19, 'Company 19', 'Address 19', '789123456', 'company19@example.com', 'Transportation Company'),
    (20, 'Company 20', 'Address 20', '456789123', 'company20@example.com', 'Transportation Company'),
    (21, 'Company 21', 'Address 21', '912345678', 'company21@example.com', 'Transportation Company'),
    (22, 'Company 22', 'Address 22', '345678912', 'company22@example.com', 'Transportation Company'),
    (23, 'Company 23', 'Address 23', '891234567', 'company23@example.com', 'Transportation Company'),
    (24, 'Company 24', 'Address 24', '678912345', 'company24@example.com', 'Transportation Company');
```

```
create table if not exists Department_Contacting_External_Company(
Company_ID int not null,
Departmentno int not null,
primary key(Company_ID, Departmentno),
foreign key(Company_ID) references External_Company(Company_ID) ON DELETE CASCADE,
foreign key(Departmentno) references Department(Departmentno) ON DELETE CASCADE
);
INSERT IGNORE INTO Department_Contacting_External_Company (Company_ID, Departmentno) VALUES
(1, 301),
(2, 301),
(7, 201),
(8, 201),
(13, 101),
(19, 501);
```

-- Which Employees have a salary above (Set amount)

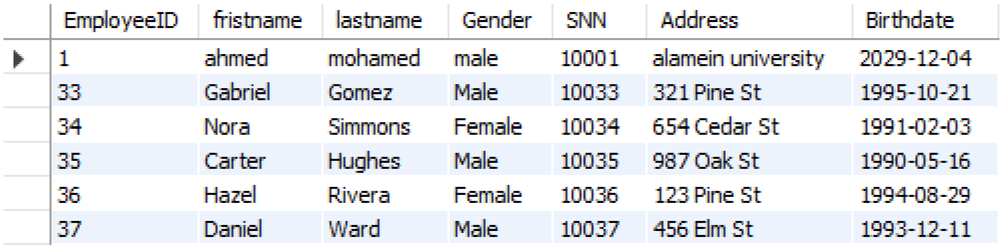
```
SELECT *
FROM Employee
WHERE EmployeeID IN (
   SELECT EmployeeID
   FROM Employee_Salary
   WHERE salary > 5200
);
```



EmployeeID	fristname	lastname	Gender	SNN	Address	Birthdate
2	hussein	ahmed	male	10002	alamein university	0002-12-04
3	ahmed	abdullah	male	10003	alamein university	0009-01-04
8	Sophia	Anderson	Female	10008	789 Pine St	1990-02-14
9	Matthew	Martinez	Male	10009	321 Elm St	1994-07-27
10	Ava	Hernandez	Female	10010	654 Main St	1992-01-10
13	Ethan	Wilson	Male	10013	456 Pine St	1993-10-19
16	Olivia	Hall	Female	10016	654 Cedar St	1992-11-27
17	James	Young	Male	10017	987 Oak St	1991-05-10
18	Sofia	Walker	Female	10018	123 Pine St	1994-09-23
19	Logan	Harris	Male	10019	456 Elm St	1993-01-06
22	Amelia	Perez	Female	10022	654 Pine St	1989-09-15
23	Mila	Parker	Female	10023	789 Cedar St	1990-07-29
24	Emma	Turner	Female	10024	123 Elm St	1994-12-28
30	Lily	Brooks	Female	10030	123 Main St	1990-11-12
32	Victoria	Coleman	Female	10032	789 Oak St	1989-07-08
33	Gabriel	Gomez	Male	10033	321 Pine St	1995-10-21
34	Nora	Simmons	Female	10034	654 Cedar St	1991-02-03
37	Daniel	Ward	Male	10037	456 Elm St	1993-12-11
40	Stella	Ross	Female	10040	654 Pine St	1989-09-20
41	Elijah	Barnes	Male	10041	123 Elm St	1994-12-03
42	Avery	Wood	Female	10042	456 Main St	1991-03-16
NULL	NULL	NULL	NULL	NULL	NULL	NULL

-- Which Employees Work at (Set Location)

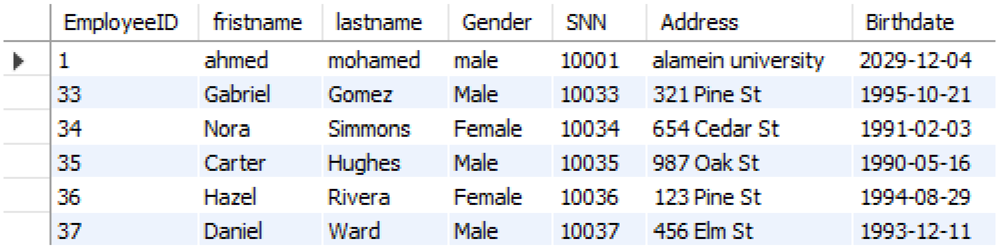
```
SELECT E.*

FROM Employee AS E

JOIN Employee_postion AS EP ON E.EmployeeID = EP.EmployeeID

JOIN Department AS D ON EP.Department_no = D.Departmentno

JOIN Department_location AS DL ON D.Departmentno = DL.Departmentno

WHERE DL.DepartmentLocation = 'Location 1A';
```



EmployeeID	fristname	lastname	Gender	SNN	Address	Birthdate
1	ahmed	mohamed	male	10001	alamein university	2029-12-04
33	Gabriel	Gomez	Male	10033	321 Pine St	1995-10-21
34	Nora	Simmons	Female	10034	654 Cedar St	1991-02-03
35	Carter	Hughes	Male	10035	987 Oak St	1990-05-16
36	Hazel	Rivera	Female	10036	123 Pine St	1994-08-29
37	Daniel	Ward	Male	10037	456 Elm St	1993-12-11

-- Which employees work in department number (enter department number)

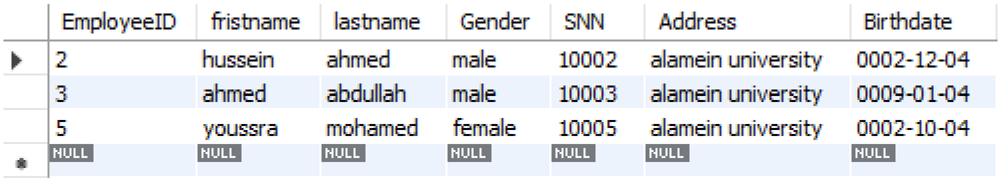
```
SELECT E.*

FROM Employee AS E

JOIN Employee_postion AS EP ON E.EmployeeID = EP.EmployeeID

WHERE EP.Department_no = 101;
```



EmployeeID	fristname	lastname	Gender	SNN	Address	Birthdate
1	ahmed	mohamed	male	10001	alamein university	2029-12-04
33	Gabriel	Gomez	Male	10033	321 Pine St	1995-10-21
34	Nora	Simmons	Female	10034	654 Cedar St	1991-02-03
35	Carter	Hughes	Male	10035	987 Oak St	1990-05-16
36	Hazel	Rivera	Female	10036	123 Pine St	1994-08-29
37	Daniel	Ward	Male	10037	456 Elm St	1993-12-11

-- Which Employees have an age above (enter number)

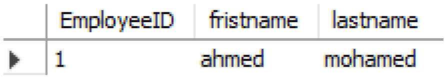
```
SELECT *

FROM Employee

WHERE TIMESTAMPDIFF(YEAR, Birthdate, CURDATE()) > 40;
```



EmployeeID	fristname	lastname	Gender	SNN	Address	Birthdate
2	hussein	ahmed	male	10002	alamein university	0002-12-04
3	ahmed	abdullah	male	10003	alamein university	0009-01-04
5	youssra	mohamed	female	10005	alamein university	0002-10-04
NULL	NULL	NULL	NULL	NULL	NULL	NULL

-- Who is the manager of department (enter department number)

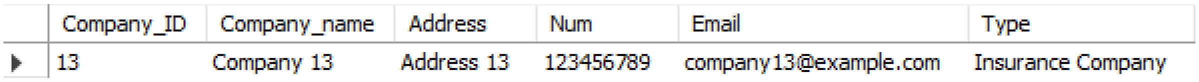
```
SELECT E.EmployeeID, E.fristname, E.lastname

FROM Employee AS E

JOIN Department AS D ON E.EmployeeID = D.Manger_ID

WHERE D.Departmentno = 101;
```



EmployeeID	fristname	lastname
1	ahmed	mohamed

-- Who are the Insurance Companies that we are in contact with

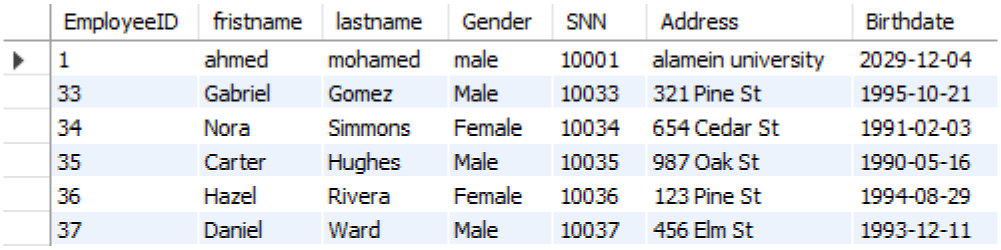
```
SELECT EC.*

FROM External_Company AS EC

JOIN Department_Contacting_External_Company AS DCEC ON EC.Company_ID = DCEC.Company_ID

WHERE EC.Type = 'Insurance Company';
```



Company_ID	Company_name	Address	Num	Email	Type
13	Company 13	Address 13	123456789	company13@example.com	Insurance Company

-- Which Employees are under (Set Department)

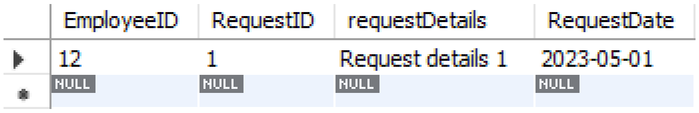
```
SELECT E.*

FROM Employee AS E

JOIN Employee_postion AS EP ON E.EmployeeID = EP.EmployeeID

JOIN Department AS D ON EP.Department_no = D.Departmentno

WHERE D.Dapartmentname = 'financial';
```



EmployeeID	fristname	lastname	Gender	SNN	Address	Birthdate
1	ahmed	mohamed	male	10001	alamein university	2029-12-04
33	Gabriel	Gomez	Male	10033	321 Pine St	1995-10-21
34	Nora	Simmons	Female	10034	654 Cedar St	1991-02-03
35	Carter	Hughes	Male	10035	987 Oak St	1990-05-16
36	Hazel	Rivera	Female	10036	123 Pine St	1994-08-29
37	Daniel	Ward	Male	10037	456 Elm St	1993-12-11

-- Were there any Maintenance requests during (Set Date) that date only

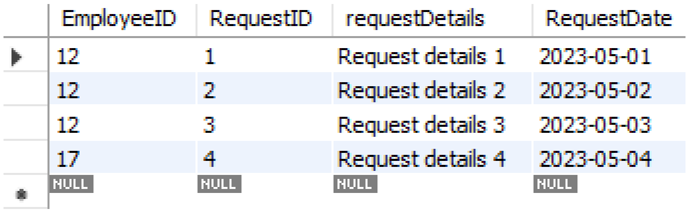
```
SELECT *

FROM Employee_Request

WHERE RequestDate = '2023-05-01';
```



EmployeeID	RequestID	requestDetails	RequestDate
12	1	Request details 1	2023-05-01
NULL	NULL	NULL	NULL

-- Were there any Maintenance requests during (Start Date, End Date)

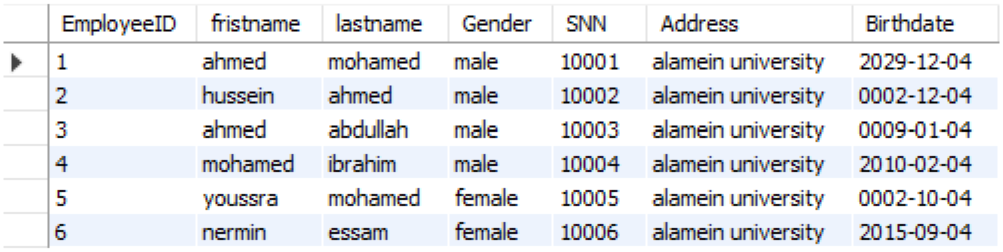
```
SELECT *

FROM Employee_Request

WHERE RequestDate BETWEEN '2023-05-01' AND '2023-05-04';
```



EmployeeID	RequestID	requestDetails	RequestDate
12	1	Request details 1	2023-05-01
12	2	Request details 2	2023-05-02
12	3	Request details 3	2023-05-03
17	4	Request details 4	2023-05-04
NULL	NULL	NULL	NULL

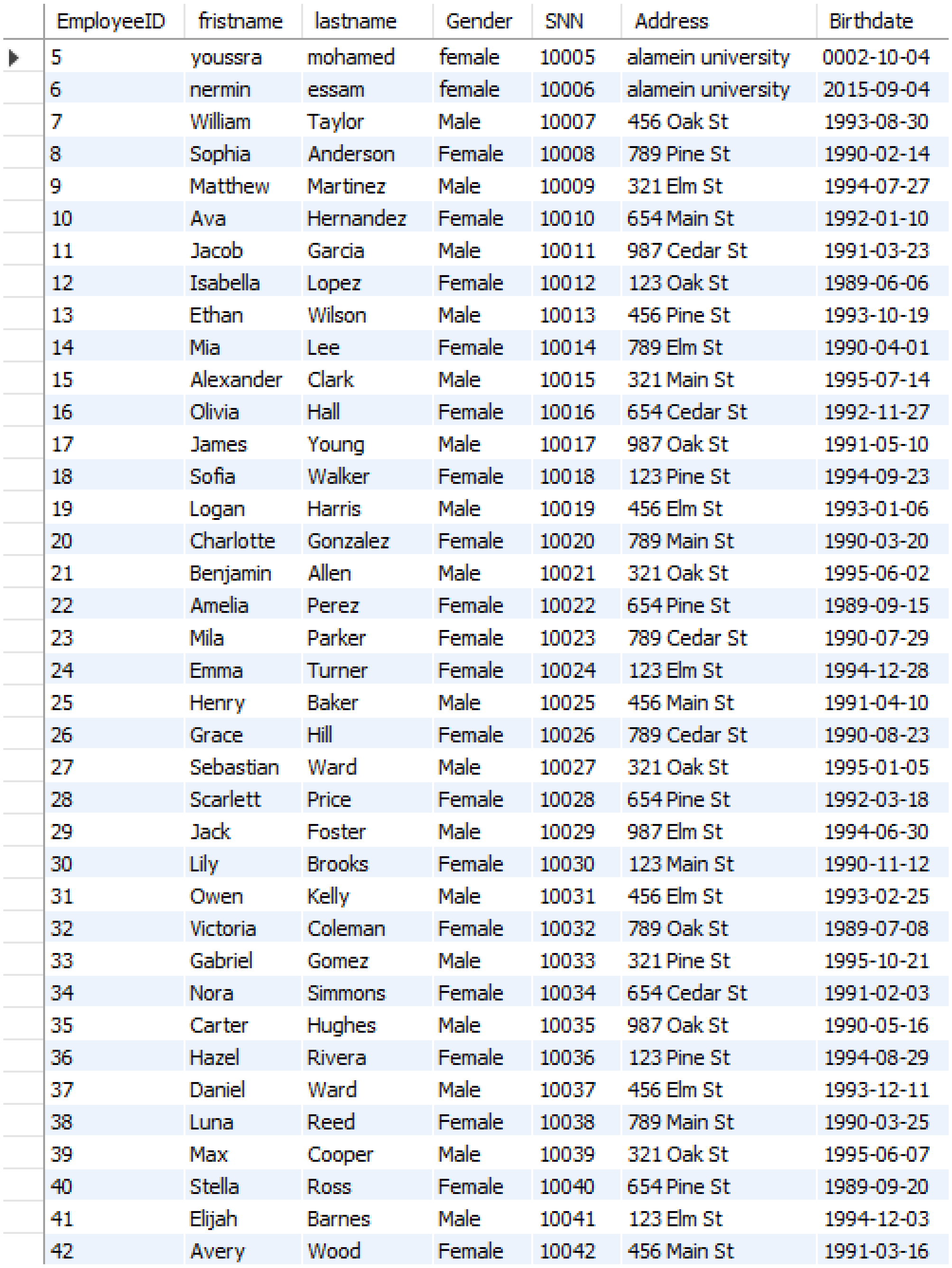
-- Which Employees have Dependents

```
SELECT DISTINCT E.*

FROM Employee AS E

JOIN Dependent AS D ON E.EmployeeID = D.EmployeeID;
```



EmployeeID	fristname	lastname	Gender	SNN	Address	Birthdate
1	ahmed	mohamed	male	10001	alamein university	2029-12-04
2	hussein	ahmed	male	10002	alamein university	0002-12-04
3	ahmed	abdullah	male	10003	alamein university	0009-01-04
4	mohamed	ibrahim	male	10004	alamein university	2010-02-04
5	youssra	mohamed	female	10005	alamein university	0002-10-04
6	nermin	essam	female	10006	alamein university	2015-09-04

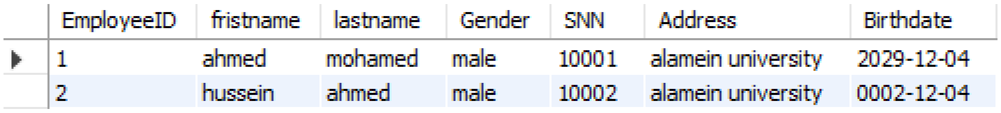
```
-- Which Employees have less than (set amount) Dependents

SELECT E.*

FROM Employee AS E

LEFT JOIN (

   SELECT EmployeeID, COUNT(*) AS DependentCount

   FROM Dependent

   GROUP BY EmployeeID

) AS D ON E.EmployeeID = D.EmployeeID

WHERE D.DependentCount < 2 OR D.DependentCount IS NULL;
```



EmployeeID	fristname	lastname	Gender	SNN	Address	Birthdate
5	youssra	mohamed	female	10005	alamein university	0002-10-04
6	nermin	essam	female	10006	alamein university	2015-09-04
7	William	Taylor	Male	10007	456 Oak St	1993-08-30
8	Sophia	Anderson	Female	10008	789 Pine St	1990-02-14
9	Matthew	Martinez	Male	10009	321 Elm St	1994-07-27
10	Ava	Hernandez	Female	10010	654 Main St	1992-01-10
11	Jacob	Garcia	Male	10011	987 Cedar St	1991-03-23
12	Isabella	Lopez	Female	10012	123 Oak St	1989-06-06
13	Ethan	Wilson	Male	10013	456 Pine St	1993-10-19
14	Mia	Lee	Female	10014	789 Elm St	1990-04-01
15	Alexander	Clark	Male	10015	321 Main St	1995-07-14
16	Olivia	Hall	Female	10016	654 Cedar St	1992-11-27
17	James	Young	Male	10017	987 Oak St	1991-05-10
18	Sofia	Walker	Female	10018	123 Pine St	1994-09-23
19	Logan	Harris	Male	10019	456 Elm St	1993-01-06
20	Charlotte	Gonzalez	Female	10020	789 Main St	1990-03-20
21	Benjamin	Allen	Male	10021	321 Oak St	1995-06-02
22	Amelia	Perez	Female	10022	654 Pine St	1989-09-15
23	Mila	Parker	Female	10023	789 Cedar St	1990-07-29
24	Emma	Turner	Female	10024	123 Elm St	1994-12-28
25	Henry	Baker	Male	10025	456 Main St	1991-04-10
26	Grace	Hill	Female	10026	789 Cedar St	1990-08-23
27	Sebastian	Ward	Male	10027	321 Oak St	1995-01-05
28	Scarlett	Price	Female	10028	654 Pine St	1992-03-18
29	Jack	Foster	Male	10029	987 Elm St	1994-06-30
30	Lily	Brooks	Female	10030	123 Main St	1990-11-12
31	Owen	Kelly	Male	10031	456 Elm St	1993-02-25
32	Victoria	Coleman	Female	10032	789 Oak St	1989-07-08
33	Gabriel	Gomez	Male	10033	321 Pine St	1995-10-21
34	Nora	Simmons	Female	10034	654 Cedar St	1991-02-03
35	Carter	Hughes	Male	10035	987 Oak St	1990-05-16
36	Hazel	Rivera	Female	10036	123 Pine St	1994-08-29
37	Daniel	Ward	Male	10037	456 Elm St	1993-12-11
38	Luna	Reed	Female	10038	789 Main St	1990-03-25
39	Max	Cooper	Male	10039	321 Oak St	1995-06-07
40	Stella	Ross	Female	10040	654 Pine St	1989-09-20
41	Elijah	Barnes	Male	10041	123 Elm St	1994-12-03
42	Avery	Wood	Female	10042	456 Main St	1991-03-16

-- Which Employees have more than (set amount) Dependents

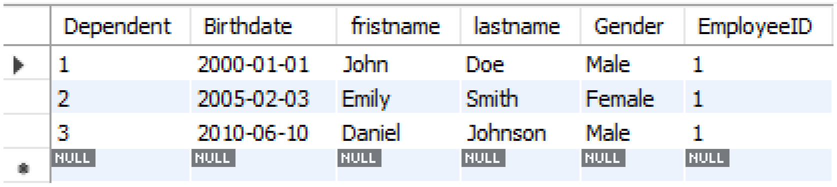
```
SELECT E.*
FROM Employee AS E
JOIN (
    SELECT EmployeeID, COUNT(*) AS DependentCount
    FROM Dependent
    GROUP BY EmployeeID
    HAVING COUNT(*) > 2
) AS D ON E.EmployeeID = D.EmployeeID;
```



	EmployeeID	fristname	lastname	Gender	SNN	Address	Birthdate
▶	1	ahmed	mohamed	male	10001	alamein university	2029-12-04
	2	hussein	ahmed	male	10002	alamein university	0002-12-04

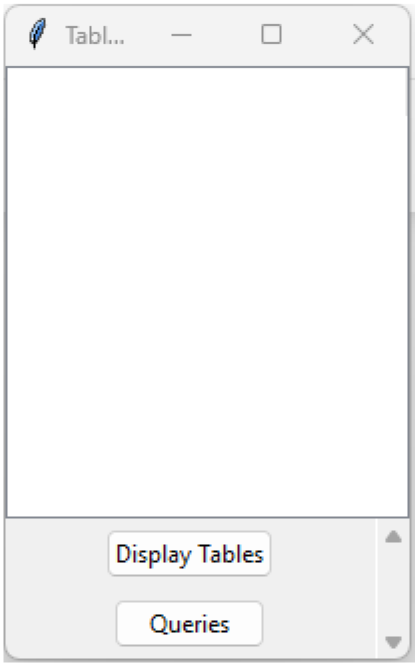
-- Who are the Dependents of (set Employee)

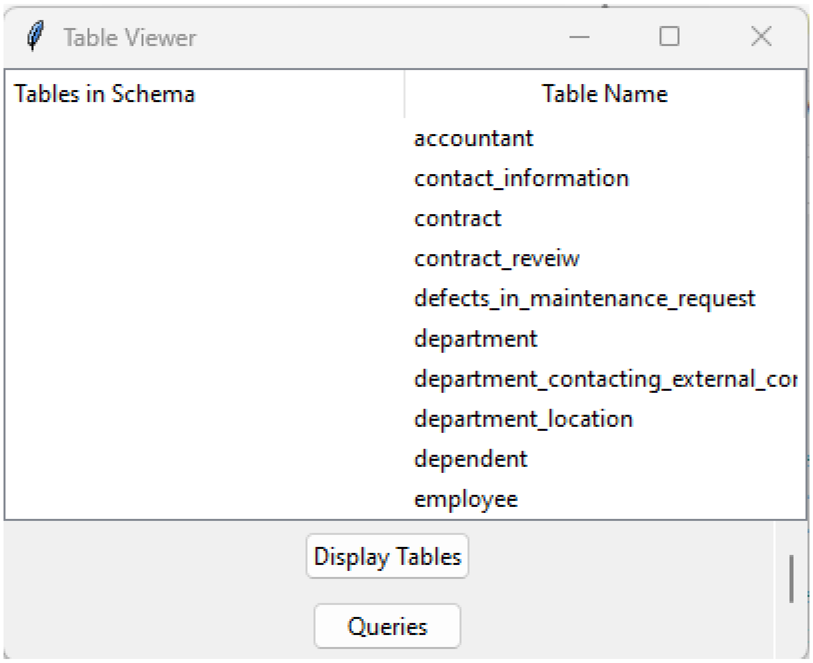
```
SELECT *
FROM Dependent
WHERE EmployeeID = 1;
```



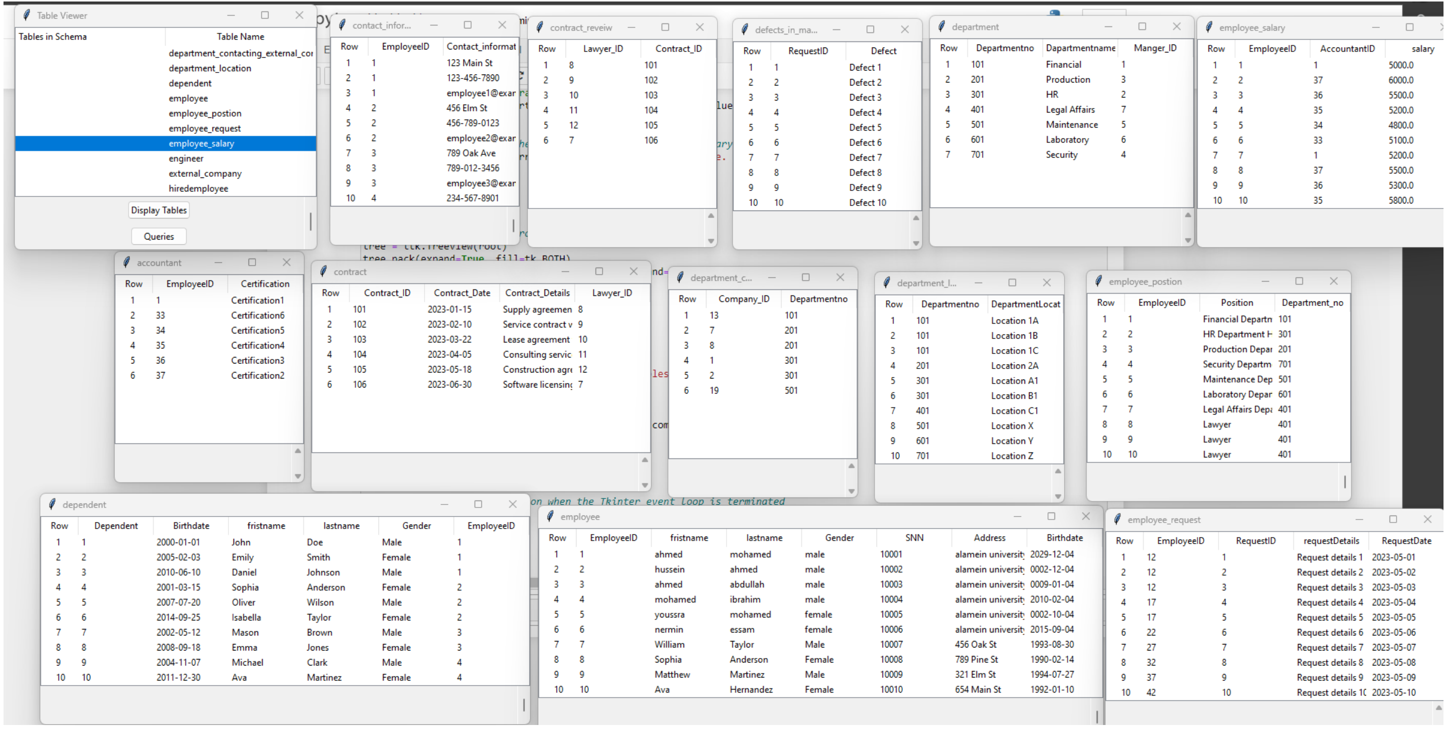
	Dependent	Birthdate	fristname	lastname	Gender	EmployeeID
▶	1	2000-01-01	John	Doe	Male	1
	2	2005-02-03	Emily	Smith	Female	1
	3	2010-06-10	Daniel	Johnson	Male	1
*	NULL	NULL	NULL	NULL	NULL	NULL

Example Gui:





Tables in Schema	Table Name
	accountant
	contact_information
	contract
	contract_reveiw
	defects_in_maintenance_request
	department
	department_contacting_external_cor
	department_location
	dependent
	employee



Tables in Schema	Table Name
	department_contacting_external_cor
	department_location
	dependent
	employee
	employee_postion
	employee_request
	employee_salary
	engineer
	external_company
	hiredemployee

contact_infor...

Row	EmployeeID	Contact_informat
1	1	123 Main St
2	1	123-456-7890
3	1	employee1@exar
4	2	456 Elm St
5	2	456-789-0123
6	2	employee2@exar
7	3	789 Oak Ave
8	3	789-012-3456
9	3	employee3@exar
10	4	234-567-8901

contract_reveiw

Row	Lawyer_ID	Contract_ID
1	8	101
2	9	102
3	10	103
4	11	104
5	12	105
6	7	106

defects_in_ma...

Row	RequestID	Defect
1	1	Defect 1
2	2	Defect 2
3	3	Defect 3
4	4	Defect 4
5	5	Defect 5
6	6	Defect 6
7	7	Defect 7
8	8	Defect 8
9	9	Defect 9
10	10	Defect 10

department

Row	Departmentno	Dapartmentname	Manger_ID
1	101	Financial	1
2	201	Production	3
3	301	HR	2
4	401	Legal Affairs	7
5	501	Maintenance	5
6	601	Laboratory	6
7	701	Security	4

employee_salary

Row	EmployeeID	AccountantID	salary
1	1	1	5000.0
2	2	37	6000.0
3	3	36	5500.0
4	4	35	5200.0
5	5	34	4800.0
6	6	33	5100.0
7	7	1	5200.0
8	8	37	5500.0
9	9	36	5300.0
10	10	35	5800.0

accountant

Row	EmployeeID	Certification
1	1	Certification1
2	33	Certification6
3	34	Certification5
4	35	Certification4
5	36	Certification3
6	37	Certification2

contract

Row	Contract_ID	Contract_Date	Contract_Details	Lawyer_ID
1	101	2023-01-15	Supply agreemen	8
2	102	2023-02-10	Service contract v	9
3	103	2023-03-22	Lease agreement	10
4	104	2023-04-05	Consulting servic	11
5	105	2023-05-18	Construction agre	12
6	106	2023-06-30	Software licensin	7

department_c...

Row	Company_ID	Departmentno
1	13	101
2	7	201
3	8	201
4	1	301
5	2	301
6	19	501

department_l...

Row	Departmentno	DepartmentLocat
1	101	Location 1A
2	101	Location 1B
3	101	Location 1C
4	201	Location 2A
5	301	Location A1
6	301	Location B1
7	401	Location C1
8	501	Location X
9	601	Location Y
10	701	Location Z

employee_postion

Row	EmployeeID	Position	Department_no
1	1	Financial Departn	101
2	2	HR Department H	301
3	3	Production Depar	201
4	4	Security Departm	701
5	5	Maintenance Dep	501
6	6	Laboratory Depar	601
7	7	Legal Affairs Depa	401
8	8	Lawyer	401
9	9	Lawyer	401
10	10	Lawyer	401

dependent

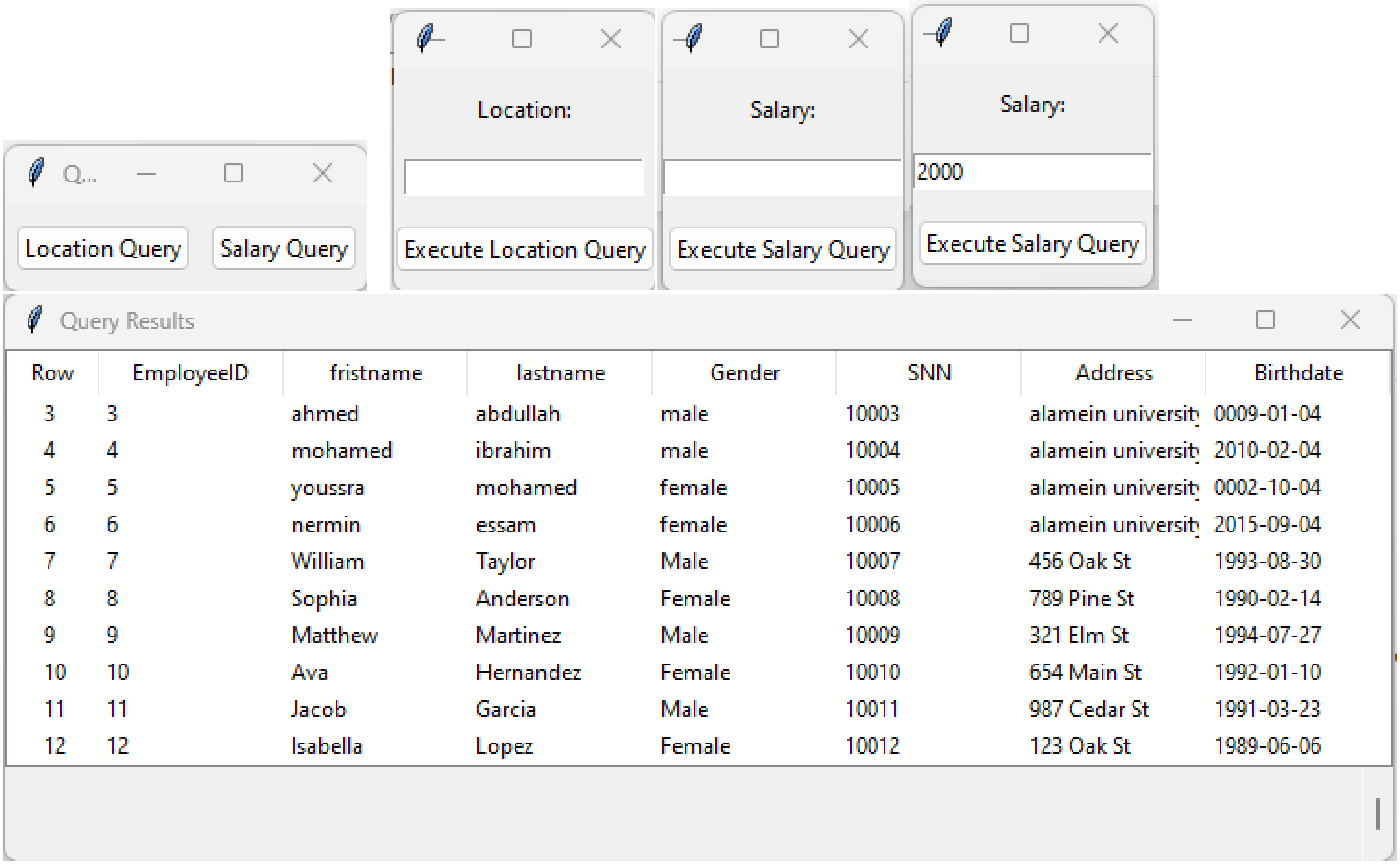
Row	Dependent	Birthdate	fristname	lastname	Gender	EmployeeID
1	1	2000-01-01	John	Doe	Male	1
2	2	2005-02-03	Emily	Smith	Female	1
3	3	2010-06-10	Daniel	Johnson	Male	1
4	4	2001-03-15	Sophia	Anderson	Female	2
5	5	2007-07-20	Oliver	Wilson	Male	2
6	6	2014-09-25	Isabella	Taylor	Female	2
7	7	2002-05-12	Mason	Brown	Male	3
8	8	2008-09-18	Emma	Jones	Female	3
9	9	2004-11-07	Michael	Clark	Male	4
10	10	2011-12-30	Ava	Martinez	Female	4

employee

Row	EmployeeID	fristname	lastname	Gender	SNN	Address	Birthdate
1	1	ahmed	mohamed	male	10001	alamein university	2029-12-04
2	2	hussein	ahmed	male	10002	alamein university	0002-12-04
3	3	ahmed	abdullah	male	10003	alamein university	0009-01-04
4	4	mohamed	ibrahim	male	10004	alamein university	2010-02-04
5	5	youssra	mohamed	female	10005	alamein university	0002-10-04
6	6	nermin	essam	female	10006	alamein university	2015-09-04
7	7	William	Taylor	Male	10007	456 Oak St	1993-08-30
8	8	Sophia	Anderson	Female	10008	789 Pine St	1990-02-14
9	9	Matthew	Martinez	Male	10009	321 Elm St	1994-07-27
10	10	Ava	Hernandez	Female	10010	654 Main St	1992-01-10

employee_request

Row	EmployeeID	RequestID	requestDetails	RequestDate
1	12	1	Request details 1	2023-05-01
2	12	2	Request details 2	2023-05-02
3	12	3	Request details 3	2023-05-03
4	17	4	Request details 4	2023-05-04
5	17	5	Request details 5	2023-05-05
6	22	6	Request details 6	2023-05-06
7	27	7	Request details 7	2023-05-07
8	32	8	Request details 8	2023-05-08
9	37	9	Request details 9	2023-05-09
10	42	10	Request details 1(	2023-05-10



Salary: 2000

Row	EmployeeID	fristname	lastname	Gender	SNN	Address	Birthdate
3	3	ahmed	abdullah	male	10003	alamein universit	0009-01-04
4	4	mohamed	ibrahim	male	10004	alamein universit	2010-02-04
5	5	youssra	mohamed	female	10005	alamein universit	0002-10-04
6	6	nermin	essam	female	10006	alamein universit	2015-09-04
7	7	William	Taylor	Male	10007	456 Oak St	1993-08-30
8	8	Sophia	Anderson	Female	10008	789 Pine St	1990-02-14
9	9	Matthew	Martinez	Male	10009	321 Elm St	1994-07-27
10	10	Ava	Hernandez	Female	10010	654 Main St	1992-01-10
11	11	Jacob	Garcia	Male	10011	987 Cedar St	1991-03-23
12	12	Isabella	Lopez	Female	10012	123 Oak St	1989-06-06