Job Data Management .

```
CREATE DATABASE Job_Portal
```

```
- Create Company Table
CREATE TABLE Company (
Company_ID INT PRIMARY KEY NOT NULL,
Company_Name VARCHAR(255) NOT NULL,
Location VARCHAR(255),
Industry VARCHAR(255)
);
```

```
- Insert Sample Data into Company Table
INSERT INTO Company (Company_ID, Company_Name, Location, Indu
VALUES
(1, 'ABC Inc.', 'New York', 'Technology'),
(2, 'XYZ Corp.', 'San Francisco', 'Finance'),
(3, 'AMD Inc.', 'Silicon Valley', 'Technology'),
(4, 'INTEL Inc.', 'Maharashtra', 'Technology'),
(5, 'SHARP Inc.', 'Dhaka', 'Technology');
```

```
- Create Job Title Table
CREATE TABLE Job_Title (
Job_Title_ID INT PRIMARY KEY NOT NULL,
Job_Title VARCHAR(255) NOT NULL
);
```

```
- Insert Sample Data into Job Title Table
INSERT INTO Job_Title (Job_Title_ID, Job_Title)
VALUES
(1, 'Software Engineer'),
(2, 'Data Analyst'),
(3, 'Game Designer'),
```

```
(4, 'Software Developer'),
(5, 'Ethical Hacker');
```

```
- Create Employee Table
CREATE TABLE Employee (
Employee_ID INT PRIMARY KEY NOT NULL,
First_Name VARCHAR(255) NOT NULL,
Last_Name VARCHAR(255),
Email VARCHAR(255),
Phone_Number VARCHAR(20) NOT NULL,
Hire_Date DATE NOT NULL
);
```

```
- Insert Sample Data into Employee Table
INSERT INTO Employee (Employee_ID, First_Name, Last_Name, Ema
VALUES
(1, 'John', 'Doe', 'john@yahoo.com', '123-456-7890', '2022-01
(2, 'Jane', 'Smith', 'jane@hotmail.com', '987-654-3210', '202
(3, 'kate', 'Roger', 'kitkat@outlook.com', '163-486-7880', '2
(4, 'Auggy', 'leosa', 'auggyles@gmail.com', '127-436-7990', '
(5, 'Samnatha', 'Cook', 'Samantha@hotmail.com', '783-456-7890
```

```
- Create Job Posting Table
CREATE TABLE Job_Posting (
Job_Posting_ID INT PRIMARY KEY NOT NULL,
Company_ID INT NOT NULL,
Job_Title_ID INT NOT NULL,
Posting_Date DATE NOT NULL,
Deadline DATE NOT NULL,
FOREIGN KEY (Company_ID) REFERENCES Company(Company_ID),
FOREIGN KEY (Job_Title_ID) REFERENCES Job_Title(Job_Title_ID)
);
```

```
- Insert Sample Data into Job Posting Table
INSERT INTO Job_Posting (Job_Posting_ID, Company_ID, Job_Titl
VALUES
```

```
(1, 1, 1, '2024-03-10', '2024-04-10'),
(2, 2, 2, '2024-03-15', '2024-04-15'),
(3, 3, 3, '2024-04-11', '2024-05-11'),
(4, 4, 4, '2024-03-21', '2024-04-21'),
(5, 5, 5, '2024-02-27', '2024-03-27');
```

```
- Create Application Table
CREATE TABLE Application (
Application_ID INT PRIMARY KEY NOT NULL,
Job_Posting_ID INT NOT NULL,
Employee_ID INT NOT NULL,
Application_Date DATE NOT NULL,
FOREIGN KEY (Job_Posting_ID) REFERENCES Job_Posting(Job_Posti
FOREIGN KEY (Employee_ID) REFERENCES Employee(Employee_ID)
);
```

```
- Insert Sample Data into Application Table
INSERT INTO Application (Application_ID, Job_Posting_ID, Empl
VALUES
(1, 1, 1, '2024-03-12'),
(2, 2, 2, '2024-03-20'),
(3, 3, 3, '2024-04-22'),
(4, 4, 4, '2024-04-01'),
(5, 5, 5, '2024-03-12');
```

```
- Create Interview Table
CREATE TABLE Interview (
Interview_ID INT PRIMARY KEY NOT NULL,
Application_ID INT NOT NULL,
Interview_Date DATE NOT NULL,
Interview_Type VARCHAR(255) NOT NULL,
FOREIGN KEY (Application_ID) REFERENCES Application(Applicati
);
```

```
- Insert Sample Data into Interview Table
INSERT INTO Interview (Interview_ID, Application_ID, Intervie
```

```
VALUES
(1, 1, '2024-03-18', 'Phone Interview'),
(2, 2, '2024-03-25', 'On-site Interview'),
(3, 3, '2024-04-28', 'Phone Interview'),
(4, 4, '2024-04-07', 'On-site Interview'),
(5, 5, '2024-03-18', 'Phone Interview');
```

```
- Create Offer Table
CREATE TABLE Offer (
Offer_ID INT PRIMARY KEY NOT NULL,
Application_ID INT NOT NULL,
Offer_Date DATE NOT NULL,
Offer_Details VARCHAR(255) NOT NULL,
FOREIGN KEY (Application_ID) REFERENCES Application(Applicati
);
```

```
- Insert Sample Data into Offer Table
INSERT INTO Offer (Offer_ID, Application_ID, Offer_Date, Offe
VALUES
(1, 1, '2024-03-20', 'Offered $80,000 per year'),
(2, 2, '2024-03-28', 'Offered $75,000 per year'),
(3, 3, '2024-05-01', 'Offered $89,000 per year'),
(4, 4, '2024-04-10', 'Offered $95,000 per year'),
(5, 5, '2024-03-21', 'Offered $115,000 per year');
```

```
- Create Hired Employees Table
CREATE TABLE Hired_Employees (
Hire_ID INT PRIMARY KEY NOT NULL,
Application_ID INT NOT NULL,
Hire_Date DATE NOT NULL,
Salary DECIMAL(10, 2) NOT NULL,
FOREIGN KEY (Application_ID) REFERENCES Application(Applicati
);
```

```
- Insert Sample Data into Hired Employees Table
INSERT INTO Hired_Employees (Hire_ID, Application_ID, Hire_Da
```

```
VALUES
(1, 1, '2024-04-01', 85000.00),
(2, 2, '2024-04-05', 80000.00),
(3, 3, '2024-05-05', 95000.00),
(4, 4, '2024-05-01', 100000.00),
(5, 5, '2024-04-01', 120000.00);
```

```
-- Retrieve all companies from the database
SELECT * FROM Company;
```

```
-- Retrieve all job titles available
SELECT * FROM Job_Title;
```

```
-- Retrieve all interviews scheduled
SELECT * FROM Interview;
```

```
-- Retrieve all offers made
SELECT * FROM Offer;
```

```
-- Retrieve all hired employees
SELECT * FROM Hired_Employees;
```

```
-- Retrieve the number of applications per job posting
SELECT Job_Posting_ID, COUNT(*) AS Num_Applications FROM Appl
```

```
-- Retrieve the number of interviews per job posting
SELECT Application.Job_Posting_ID, COUNT(*) AS Num_Interviews
FROM Application INNER JOIN Interview ON Application.Applicat
GROUP BY Application.Job_Posting_ID;
```

```
-- Retrieve the number of offers made per job posting
SELECT Application.Job_Posting_ID, COUNT(*) AS Num_Offers
```

```
FROM Application INNER JOIN Offer ON Application.Application_
GROUP BY Application.Job_Posting_ID;
```

```
-- Retrieve the number of hires per job posting
SELECT Application.Job_Posting_ID, COUNT(*) AS Num_Hires
FROM Application INNER JOIN Hired_Employees ON Application.Ap
GROUP BY Application.Job_Posting_ID;
```

```
-- Calculate the average salary offered
SELECT AVG(Salary) AS Avg_Salary_Offered FROM Hired_Employees
```

```
-- Calculate the total number of job postings
SELECT COUNT(*) AS Total_Job_Postings FROM Job_Posting;
```

```
-- Calculate the total number of applications received
SELECT COUNT(*) AS Total_Applications FROM Application;
```

```
-- Calculate the total number of interviews scheduled
SELECT COUNT(*) AS Total_Interviews FROM Interview;
```

```
-- Calculate the total number of offers made
SELECT COUNT(*) AS Total_Offers FROM Offer;
```

```
-- Calculate the total number of employees hired
SELECT COUNT(*) AS Total_Hired_Employees FROM Hired_Employees
```

```
-- Retrieve all job postings in a specific industry
SELECT * FROM Job_Posting WHERE Company_ID IN (SELECT Company_
```

```
-- Retrieve all applications submitted after a certain date
SELECT * FROM Application WHERE Application_Date > '2024-03-1
```

```
-- Retrieve all interviews scheduled before a certain date
SELECT * FROM Interview WHERE Interview_Date < '2024-04-18';
```