```
/*
1. Provide a SQL script that initializes the database for the Job Board scenario “CareerHub”.
*/
create database if not exists CareerHub;
use CareerHub;
/*
2. Create tables for Companies, Jobs, Applicants and Applications.
*/
-- -----------------------------------------------------
-- Table `CareerHub`.`companies`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `CareerHub`.`companies` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `companyName` VARCHAR(255) NOT NULL,
  `Location` VARCHAR(255) NOT NULL,
  PRIMARY KEY (`id`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `CareerHub`.`jobs`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `CareerHub`.`jobs` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `companies_id` INT NOT NULL,
  `jobTitle` VARCHAR(255) NOT NULL,
  `jobDescription` VARCHAR(500) NOT NULL,
  `jobLocation` VARCHAR(255) NOT NULL,
  `salary` DOUBLE NULL,
  `jobType` VARCHAR(255) NOT NULL,
  `postedDate` DATETIME NULL,
  PRIMARY KEY (`id`),
  INDEX `fk_jobs_companies1_idx` (`companies_id` ASC) VISIBLE,
  CONSTRAINT `fk_jobs_companies1`
    FOREIGN KEY (`companies_id`)
    REFERENCES `CareerHub`.`companies` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `CareerHub`.`applicants`
-- -----------------------------------------------------
```

```
CREATE TABLE IF NOT EXISTS `CareerHub`.`applicants` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `firstName` VARCHAR(255) NOT NULL,
  `lastName` VARCHAR(255) NULL,
  `email` VARCHAR(255) NOT NULL,
  `phone` VARCHAR(255) NOT NULL,
  `resume` VARCHAR(255) NOT NULL,
  PRIMARY KEY (`id`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `CareerHub`.`applications`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `CareerHub`.`applications` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `jobs_id` INT NOT NULL,
  `applicants_id` INT NOT NULL,
  `applicationDate` DATETIME NOT NULL,
  `coverLetter` TEXT(500) NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `fk_applications_applicants_idx` (`applicants_id` ASC) VISIBLE,
  INDEX `fk_applications_jobs1_idx` (`jobs_id` ASC) VISIBLE,
  CONSTRAINT `fk_applications_applicants`
    FOREIGN KEY (`applicants_id`)
    REFERENCES `CareerHub`.`applicants` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_applications_jobs1`
    FOREIGN KEY (`jobs_id`)
    REFERENCES `CareerHub`.`jobs` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;

/*
3. Define appropriate primary keys, foreign keys, and constraints.
*/
-- jobs table
CREATE TABLE IF NOT EXISTS `CareerHub`.`jobs` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `companies_id` INT NOT NULL,
  `jobTitle` VARCHAR(255) NOT NULL,
  `jobDescription` VARCHAR(500) NOT NULL,
```

```
  `jobLocation` VARCHAR(255) NOT NULL,
  `salary` DOUBLE NULL,
  `jobType` VARCHAR(255) NOT NULL,
  `postedDate` DATETIME NULL,
  PRIMARY KEY (`id`),
  INDEX `fk_jobs_companies1_idx` (`companies_id` ASC) VISIBLE,
  CONSTRAINT `fk_jobs_companies1`
    FOREIGN KEY (`companies_id`)
    REFERENCES `CareerHub`.`companies` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
/*
4. Ensure the script handles potential errors, such as if the database or tables already exist.
*/
create database if not exists CareerHub;


-- companies table
insert into companies (companyName, Location) VALUES
('Pearson Hardman', 'New York'),
('Zane Specter Litt', 'New York'),
('Kaldor & Zane', 'London');

-- jobs table
insert into jobs (companies_id, jobTitle, jobDescription, jobLocation, salary, jobType,
postedDate) VALUES
(1, 'Senior Partner', 'Lead the firm and manage high-profile cases.', 'New York', 250000.00,
'full-time', '2024-04-01'),
(2, 'Associate Attorney', 'Handle various legal issues', 'New York', 120000.00, 'full-time',
'2024-04-02'),
(1, 'Junior Associate', 'Assist senior attorneys', 'New York', 80000.00, 'full-time', '2024-04-03'),
(2, 'Investment Analyst', 'Analyze market trends and make investment recommendations.', 'New
York', 150000.00, 'full-time', '2024-04-04'),
(3, 'Paralegal', 'Assist attorneys with document preparation.', 'London', 60000.00, 'full-time',
'2024-04-05'),
(1, 'Legal Secretary', 'Provide administrative support', 'New York', 70000.00, 'full-time',
'2024-04-06');

-- applicants table
insert into applicants (firstName, lastName, email, phone, resume) VALUES
('Harvey', 'Specter', 'harvey@example.com', '12345', 'harveyspecter_resume.pdf'),
('Mike', 'Ross', 'mike@example.com', '54321', 'mikeross_resume.pdf'),
('Donna', 'Paulsen', 'donnan@example.com', '55500', 'donnapaulsen_resume.pdf'),
```

('Jessica', 'Pearson', 'jessica@example.com', '34321', 'jessicapearson_resume.pdf'),
('Louis', 'Litt', 'louis@example.com', '58765', 'louislitt_resume.pdf'),
('Rachel', 'Zane', 'rachel@example.com', '52468', 'rachelzane_resume.pdf'),
('Robert', 'Zane', 'robert@example.com', '50357', 'robertzane_resume.pdf');

-- applications table
insert into applications (jobs_id, applicants_id, applicationDate, coverLetter) VALUES
(3, 2, '2024-04-08', 'I am a highly skilled attorney with a passion for justice.'),
(1, 1, '2024-04-08', 'I have experience in handling complex legal matters.'),
(2, 3, '2024-04-09', 'I am eager to learn and contribute to the legal team.'),
(1, 4, '2024-04-10', 'I have a strong background in legal analysis.'),
(5, 5, '2024-04-11', 'I am detail-oriented and adept at legal research.'),
(6, 6, '2024-04-12', 'I have extensive experience in legal administration.');

/*
5. Write an SQL query to count the number of applications received for each job listing in the "Jobs" table.
Display the job title and the corresponding application count. Ensure that it lists all jobs, even if they have no applications.
*/

select j.jobtitle, count(a.id) as application_count
from jobs j
left join applications a on j.id = a.jobs_id
group by j.jobtitle;

jobtitle	application_co...
Senior Partner	2
Associate Attorney	1
Junior Associate	1
Investment Analyst	0
Paralegal	1
Legal Secretary	1

/*
6. Develop an SQL query that retrieves job listings from the "Jobs" table within a specified salary range.
Allow parameters for the minimum and maximum salary values. Display the job title, company name, location, and salary for each matching job.
*/
select j.jobtitle, c.companyname, j.joblocation, j.salary
from jobs j
join companies c on j.companies_id = c.id
where j.salary >= 50000 and j.salary <= 100000;

jobtitle	companyname	joblocation	salary
Junior Associate	Pearson Hardman	New York	80000
Paralegal	Kaldor & Zane	London	60000
Legal Secretary	Pearson Hardman	New York	70000

/*
7. Write an SQL query that retrieves the job application history for a specific applicant.
Allow a parameter for the ApplicantID, and return a result set with the job titles, company names,
and application dates for all the jobs the applicant has applied to.
*/

```
select j.jobtitle, c.companyname, a.applicationdate
from applications a
join jobs j on a.jobs_id = j.id
join companies c on j.companies_id = c.id
where a.applicants_id = 1;
```

jobtitle	companyname	applicationdate
Senior Partner	Pearson Hardman	2024-04-08 00:00:00

/*
8. Create an SQL query that calculates and displays the average salary offered by all companies for job listings in the "Jobs" table.
Ensure that the query filters out jobs with a salary of zero.
*/

```
select avg(j.salary) as average_salary
from jobs j
where j.salary > 0;
```

average_salary
121666.66666666667

-- average salary offered by each company

```
select c.companyName, AVG(j.salary) AS average_salary
from jobs j
join companies c ON j.companies_id = c.id
where j.salary > 0
group by c.companyName;
```

companyName	average_salary
Pearson Hardman	133333.33333333334
Zane Specter Litt	135000
Kaldor & Zane	60000

/*
9. Write an SQL query to identify the company that has posted the most job listings. Display the company name along with the count of job listings they have posted. Handle ties if multiple companies have the same maximum count.
*/

```
select c.companyname, count(j.id) as job_count
from companies c
join jobs j on c.id = j.companies_id
group by c.companyname
order by job_count desc
limit 1;
```

companyname	job_count
Pearson Hardman	3

/*
10. Find the applicants who have applied for positions in companies located in 'CityX' and have at least 3 years of experience.
*/

```
select distinct a.firstname, a.lastname
from applicants a
join applications app on a.id = app.applicants_id
join jobs j on app.jobs_id = j.id
join companies c on j.companies_id = c.id
where c.location = 'New York' and a.experience >= 3;

select distinct a.firstname, a.lastname
from applicants a
join applications app on a.id = app.applicants_id
join jobs j on app.jobs_id = j.id
join companies c on j.companies_id = c.id
where c.location = 'London';
```

firstname	lastname
Louis	Litt

/*
11. Retrieve a list of distinct job titles with salaries between $60,000 and $80,000.
*/

```
select distinct jobtitle
from jobs
where salary between 60000 and 80000;
```

jobtitle
Junior Associate
Paralegal
Legal Secretary

/*
12. Find the jobs that have not received any applications.
*/

```
select j.jobtitle
from jobs j
left join applications a on j.id = a.jobs_id
where a.id is null;
```

jobtitle
Investment Analyst

/*
13. Retrieve a list of job applicants along with the companies they have applied to and the positions they have applied for.
*/

```
select a.firstname, a.lastname, c.companyname, j.jobtitle
from applicants a
join applications app on a.id = app.applicants_id
join jobs j on app.jobs_id = j.id
join companies c on j.companies_id = c.id;
```

firstname	lastname	companyname	jobtitle
Harvey	Specter	Pearson Hardman	Senior Partner
Jessica	Pearson	Pearson Hardman	Senior Partner
Mike	Ross	Pearson Hardman	Junior Associate
Rachel	Zane	Pearson Hardman	Legal Secretary
Donna	Paulsen	Zane Specter Litt	Associate Attorney
Louis	Litt	Kaldor & Zane	Paralegal

/*
14. Retrieve a list of companies along with the count of jobs they have posted, even if they have not received any applications.
*/

```
select c.companyname, count(j.id) as job_count
from companies c
join jobs j on c.id = j.companies_id
group by c.companyname;
```

companyname	job_count
Pearson Hardman	3
Zane Specter Litt	2
Kaldor & Zane	1

/*
15. List all applicants along with the companies and positions they have applied for, including those who have not applied.
*/

```
select a.firstname, a.lastname, c.companyname, j.jobtitle
from applicants a
left join applications app on a.id = app.applicants_id
left join jobs j on app.jobs_id = j.id
left join companies c on j.companies_id = c.id;
```

firstname	lastname	companyname	jobtitle
Harvey	Specter	Pearson Hardman	Senior Partner
Mike	Ross	Pearson Hardman	Junior Associate
Donna	Paulsen	Zane Specter Litt	Associate Attorney
Jessica	Pearson	Pearson Hardman	Senior Partner
Louis	Litt	Kaldor & Zane	Paralegal
Rachel	Zane	Pearson Hardman	Legal Secretary
Robert	Zane	NULL	NULL

/*
16. Find companies that have posted jobs with a salary higher than the average salary of all jobs.
*/

```
select distinct c.companyname
from companies c
join jobs j on c.id = j.companies_id
where j.salary > (select avg(salary) from jobs);
```

companyname
Pearson Hardman
Zane Specter Litt

/*
17. Display a list of applicants with their names and a concatenated string of their city and state.
*/

```
select a.firstname, a.lastname, concat_ws(', ', c.city, c.state) as location
from applicants a
join cities c on a.city_id = c.id;
```

/*
18. Retrieve a list of jobs with titles containing either 'Developer' or 'Engineer'.
*/

```
select *
from jobs
where jobtitle like '%legal%' or jobtitle like '%associate%';
```

id	companies_id	jobTitle	jobDescription	jobLocation	salary	jobType	postedDate
2	2	Associate Attorney	Handle various legal issues	New York	120000	full-time	2024-04-02 00:00:00
3	1	Junior Associate	Assist senior attorneys	New York	80000	full-time	2024-04-03 00:00:00
5	3	Paralegal	Assist attorneys with document preparation.	London	60000	full-time	2024-04-05 00:00:00
6	1	Legal Secretary	Provide administrative support	New York	70000	full-time	2024-04-06 00:00:00
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

/*
19. Retrieve a list of applicants and the jobs they have applied for, including those who have not applied and jobs without applicants.
*/

```
select a.firstname, a.lastname, c.companyname, j.jobtitle
from applicants a
left join applications app on a.id = app.applicants_id
left join jobs j on app.jobs_id = j.id
left join companies c on j.companies_id = c.id;
```

firstname	lastname	companyname	jobtitle
Harvey	Specter	Pearson Hardman	Senior Partner
Mike	Ross	Pearson Hardman	Junior Associate
Donna	Paulsen	Zane Specter Litt	Associate Attorney
Jessica	Pearson	Pearson Hardman	Senior Partner
Louis	Litt	Kaldor & Zane	Paralegal
Rachel	Zane	Pearson Hardman	Legal Secretary
Robert	Zane	NULL	NULL

/*
20. List all combinations of applicants and companies where the company is in a specific city and the applicant has more than 2 years of experience. For example: city=Chennai
*/

```
select a.firstname, a.lastname, c.companyname
from applicants a
cross join companies c
where a.experience > 2 and c.city = 'London';

select a.firstname, a.lastname, c.companyname
from applicants a
cross join companies c
where c.Location = 'New York';
```

	firstname	lastname	companyname	
	Harvey	Specter	Zane Specter Litt	
	Harvey	Specter	Pearson Hardman	
	Mike	Ross	Zane Specter Litt	
	Mike	Ross	Pearson Hardman	
	Donna	Paulsen	Zane Specter Litt	
	Donna	Paulsen	Pearson Hardman	
	Jessica	Pearson	Zane Specter Litt	
	Jessica	Pearson	Pearson Hardman	
	Louis	Litt	Zane Specter Litt	
	Louis	Litt	Pearson Hardman	
	Rachel	Zane	Zane Specter Litt	
	Rachel	Zane	Pearson Hardman	
	Robert	Zane	Zane Specter Litt	
	Robert	Zane	Pearson Hardman	