

Tech ABC Corp - HR Database

Ba Khuong Nguyen, 12.12.2024



How to use this Template

- Make a copy of this Google Slide deck.
- We have provided these slides as a guide to ensure that you submit all the required components to successfully complete your project.
- When presenting your project, please only think of this as a guide. We encourage you to use creative freedom when making changes, as long as the required information is present.
- **Remember to delete this and all** of the other example slides before you submit your project.
- **Remember to add your name and the date** to the cover slide

Reference slide remove
before you submit

Business Scenario

Business requirement

Tech ABC Corp saw explosive growth with a sudden appearance onto the gaming scene with their new AI-powered video game console. As a result, they have gone from a small 10 person operation to 200 employees and 5 locations in under a year. HR is having trouble keeping up with the growth, since they are still maintaining employee information in a spreadsheet. While that worked for ten employees, it has become increasingly cumbersome to manage as the company expands.

As such, the HR department has tasked you, as the new data architect, to design and build a database capable of managing their employee information.

Dataset

The [HR dataset](#) you will be working with is an Excel workbook which consists of 206 records, with eleven columns. The data is in human readable format, and has not been normalized at all. The data lists the names of employees at Tech ABC Corp as well as information such as job title, department, manager's name, hire date, start date, end date, work location, and salary.

IT Department Best Practices

The IT Department has certain Best Practices policies for databases you should follow, as detailed in the [Best Practices document](#).



Step 1

Data Architecture

Foundations

Step 1: Data Architecture Foundations

Hi,

Welcome to Tech ABC Corp. We are excited to have some new talent onboard. As you may already know, Tech ABC Corp has recently experienced a lot of growth. Our AI powered video game console WOPR has been hugely successful and as a result, our company has grown from 10 employees to 200 in only 6 months (and we are projecting a 20% growth a year for the next 5 years). We have also grown from our Dallas, Texas office, to 4 other locations nationwide: New York City, NY, San Francisco, CA, Minneapolis, MN, and Nashville, TN.

While this growth is great, it is really starting to put a strain on our record keeping in HR. We currently maintain all employee information on a shared spreadsheet. When HR consisted of only myself, managing everyone on an Excel spreadsheet was simple, but now that it is a shared document I am having serious reservations about data integrity and data security. If the wrong person got their hands on the HR file, they would see the salaries of every employee in the company, all the way up to the president.

After speaking with Jacob Lauber, the manager of IT, he suggested I put in a request to have my HR Excel file converted into a database. He suggested I reach out to you as I am told you have experience in designing and building databases. When you are building this, please keep in mind that I want any employee with a domain login to be have read only access the database. I just don't want them having access to salary information. That needs to be restricted to HR and management level employees only. Management and HR employees should also be the only ones with write access. By our current estimates, 90% of users will be read only.

I also want to make sure you know that am looking to turn my spreadsheet into a live database, one I can input and edit information into. I am not really concerned with reporting capabilities at the moment. Since we are working with employee data we are required by federal regulations to maintain this data for at least 7 years; additionally, since this is considered business critical data, we need to make sure it gets backed up properly.

As a final consideration. We would like to be able to connect with the payroll department's system in the future. They maintain employee attendance and paid time off information. It would be nice if the two systems could interface in the future

I am looking forward to working with you and seeing what kind of database you design for us.

Thanks,
Sarah Collins
Head of HR

Data Architect Business Requirement

- **Purpose of the new database:**

The purpose of the new database is firstly to replace the old way of managing data. Secondly, with the new database, users are able to input and edit information. The new database also needs satisfy the data integrity and data security requirements as the data is being shared amongst different parties who might have different level of ownership or access rights.

- **Describe current data management solution:**

Current data management is solely a shared excel spreadsheet, which everyone can have access to. There is a serious concern about data integrity and security. Moreover, as the company grows at a relatively strong rate, excel spreadsheet seem to be a not good solution in the long term.

- **Describe current data available:**

Current available data is in table form with columns consisting of all information from the employees.

- **Additional data requests:**

Data must be stored and carefully backed up for at least 7 years. The reporting capabilities are not considered at the moment. The database also needs to have capabilities to connect to another system in the future.

- **Who will own/manage data**

HR and management level employees.

- **Who will have access to database**

All employees with domain login, including HR and management level.

Data Architect Business Requirement

- **Estimated size of database**

Current data has only 206 rows, which are also 206 employees.

- **Estimated annual growth**

Growth rate is expected to be 20% (around 40 rows/employees) a year for the next 5 years.

- **Is any of the data sensitive/restricted**

Salary information is only available to HR, management level employees. Normal employees with do not have access to salary information.

Data Architect Technical Requirement

- **Justification for the new database**

Since the data is expected to grow by 20% a year in the next 5 years, the new database is suitable as a scalable solution.

The new database will resolve the concerns with data integrity and security.

The new database can be easily expanded to connect with other system.

- **Database objects**

- Employee
- Department
- Salary
- Education
- Job
- Manager
- Address
- Location
- State
- Employee_Info

- **Data ingestion**

ETL is a good method to ingest data into the database and it can be efficiently automated with the cloud infrastructure. In this project, the data is provided in an excel files.

Data Architect Technical Requirement

- **Data governance (Ownership and User access)**

Ownership: HR

User Access:

- Employee: read-only access without access to salary information
- HR: full read & write access

- **Scalability**

Replication is expected for the database as the company grows in the number of employees. Furthermore, replication will later enable the load distribution on various database.

- **Flexibility**

- Similar structure of the database should be communicated amongst different departments to ensure future compatibility when corporations are needed.

- **Storage & retention**

Storage (disk or in-memory): check [IT best practices document](#)

Retention: data must be store for at least 7 years

- **Backup**

- Full backup: once a week
- Incremental backup: daily

Step 2

Relational Database Design

Step 2: Relational Database Design

This step is where you will go through the process of designing a new database for Tech ABC Corp's HR department. Using the [dataset](#) provided, along with the requirements gathered in step one, you are going to develop a relational database set to the 3NF.

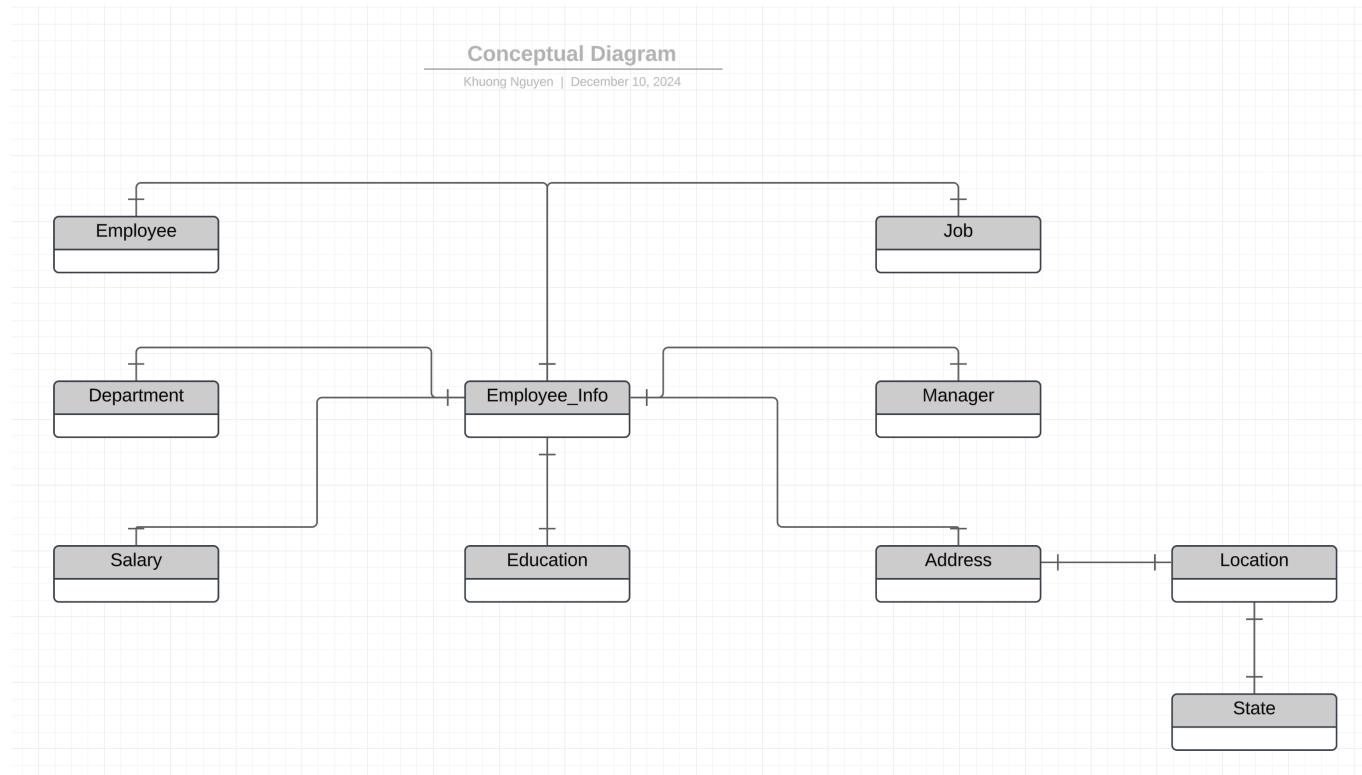
Using Lucidchart, you will create 3 entity relationship diagrams (ERDs) to show how you developed the final design for your data.

You will submit a screenshot for each of the 3 ERDs you create. You will find detailed instructions for developing each of the ERDs over the next several pages.

ERD

- **Conceptual**

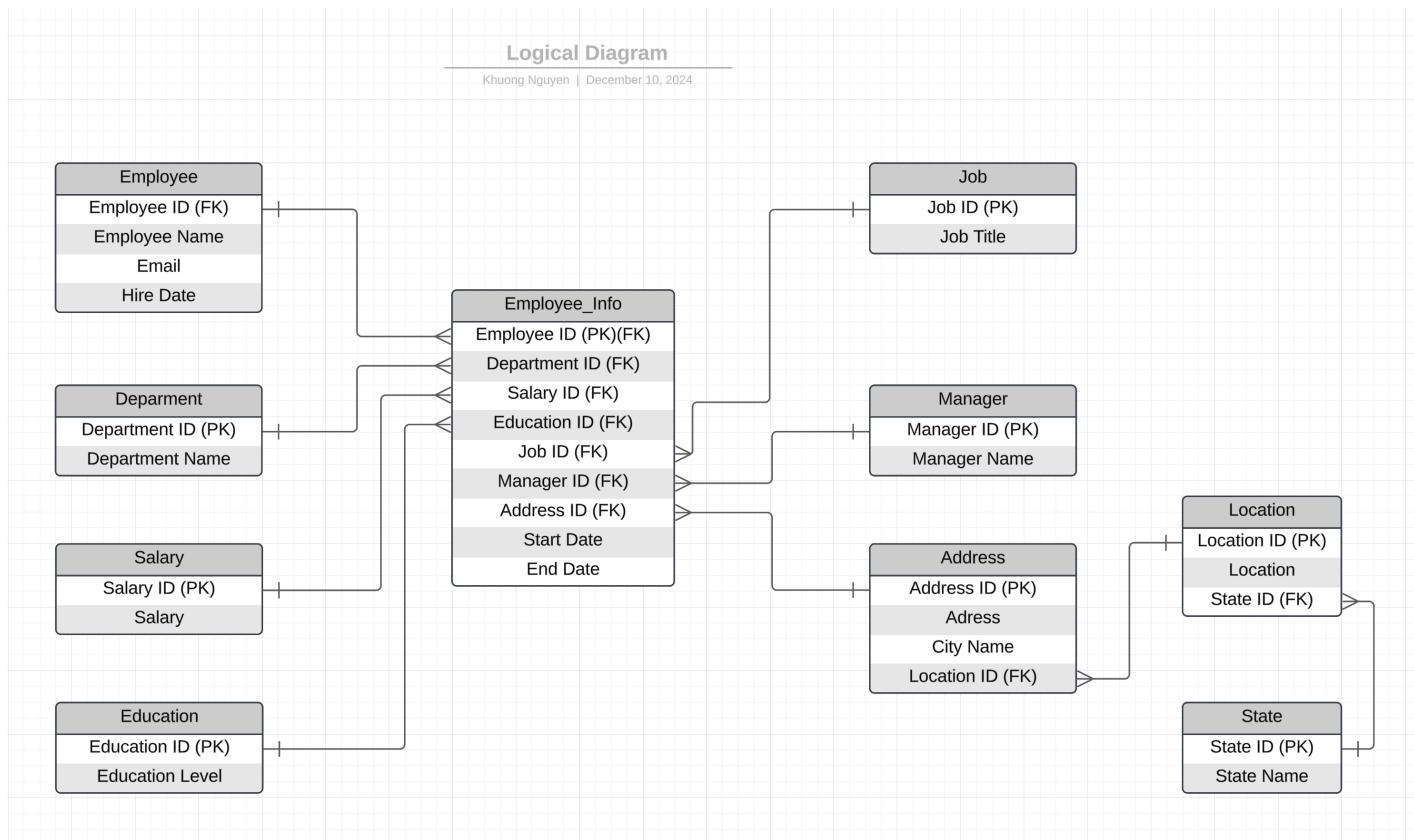
The following is the physical model of the proposed database:



ERD

- **Logical**

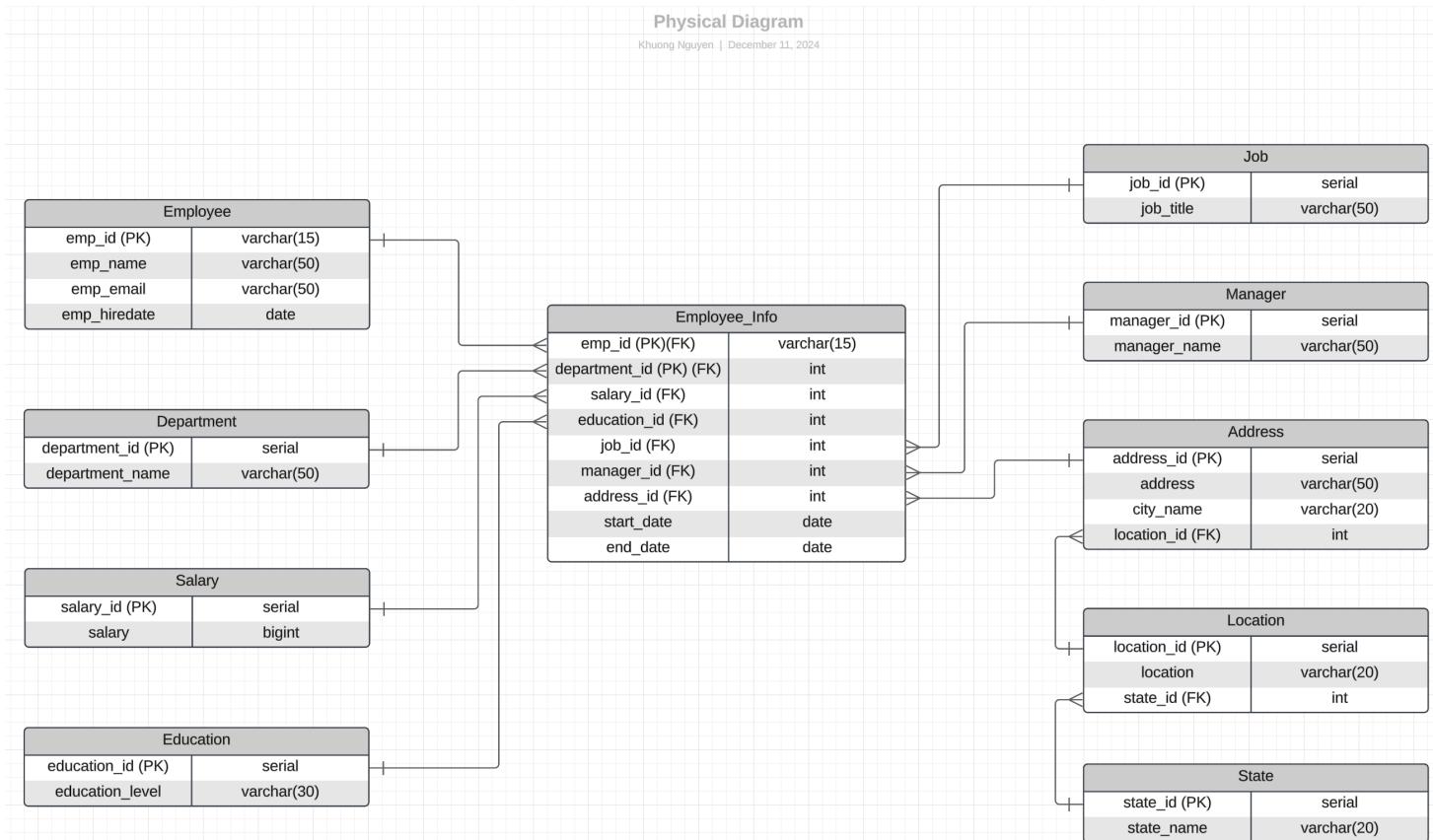
The following is the logical model of the proposed database:



ERD

- **Physical**

The following is the physical model of the proposed database:



Step 3

Create A Physical Database

Step 3: Create A Physical Database

In this step, you will be turning your database model into a physical database.

You will:

- Create the database using SQL DDL commands
- Load the data into your database, utilizing flat file ETL
- Answer a series of questions using CRUD SQL commands to demonstrate your database was created and populated correctly

Submission

For this step, you will need to submit SQL files containing all DDL SQL scripts used to create the database.

You will also have to submit screenshots showing CRUD commands, along with results for each of the questions found in the starter template.

Hints

Your DDL script will be graded by running the code you submit. Please ensure your SQL code runs properly!

Foreign keys cannot be created on tables that do not exist yet, so it may be easier to create all tables in the database, then to go back and run modify statements on the tables to create foreign key constraints.

After running CRUD commands like update, insert, or delete, run a SELECT* command on the affected table, so the reviewer can see the results of the command.

DDL

The following scripts was implemented to create the elements of the proposed database:

- Employee

```
1  create table if not exists Employee(
2    emp_id varchar(15) primary key,
3    emp_name varchar(50),
4    emp_email varchar(50),
5    emp_hiredate date
6  );
7
8  insert into Employee (
9    emp_id, emp_name, emp_email, emp_hiredate
10 )
11 select distinct emp_id, emp_nm, email, hire_dt
12 from proj_stg
13
14
```

- Department

```
1  create table if not exists Department (
2    department_id serial primary key,
3    department_name varchar(50)
4  );
5
6  insert into Department (
7    department_name
8  )
9  select distinct(department_nm)
10 from proj_stg
```

DDL

- Salary

```
1  create table if not exists Salary (
2    salary_id serial primary key,
3    salary bigint
4 );
5
6  insert into Salary (
7    salary
8 )
9  select distinct(salary)
10 from proj_stg
11
```

- Education

```
1  create table if not exists Education (
2    education_id serial primary key,
3    education_level varchar(50)
4 );
5
6  insert into Education (
7    education_level
8 )
9  select distinct(education_lvl)
10 from proj_stg
11
```

DDL

- Job

```
1  create table if not exists Job (
2    job_id serial primary key,
3    job_title varchar(50)
4 );
5
6  insert into Job (
7    job_title
8  )
9  select distinct job_title
10 from proj_stg
```

- Manager

```
1  create table if not exists Manager (
2    manager_id serial primary key,
3    manager_name varchar(50)
4 );
5
6  insert into Manager (
7    manager_name
8  )
9  select distinct manager
10 from proj_stg
```

DDL

● Address

```
1  create table if not exists Address (
2    address_id serial primary key,
3    address varchar(50),
4    city_name varchar(50),
5    location_id int references Location(location_id)
6  );
7
8  insert into Address (
9    address, city_name, location_id
10 )
11 select distinct proj_stg.address, proj_stg.city, Location.location_id
12 from proj_stg
13 inner join Location
14 on proj_stg.location = Location.location
```

● Location

```
1  create table if not exists Location (
2    location_id serial primary key,
3    location varchar(50),
4    state_id int references state(state_id)
5  );
6
7  insert into Location (
8    location, state_id
9  )
10 select distinct(proj_stg.location), State.state_id
11 from proj_stg
12 inner join State
13 on proj_stg.state = State.state_name
```

● State

```
1  create table if not exists State (
2    state_id serial primary key,
3    state_name varchar(50)
4  );
5
6  insert into State (
7    state_name
8  )
9  select distinct(state)
10 from proj_stg
```

DDL

● Employee_Info

```
1  create table if not exists EmployeeInfo (
2      emp_id varchar(15) references Employee(emp_id),
3      department_id int references Department(department_id),
4      salary_id int references Salary(salary_id),
5      education_id int references Education(education_id),
6      job_id int references Job(job_id),
7      manager_id int references Manager(manager_id),
8      address_id int references Address(address_id),
9      start_date date,
10     end_date date
11 );
12
13 insert into EmployeeInfo (
14     emp_id,
15     department_id,
16     salary_id,
17     education_id,
18     job_id,
19     manager_id,
20     address_id,
21     start_date,
22     end_date
23 )
24 select distinct
25     Employee.emp_id,
26     Department.department_id,
27     Salary.salary_id,
28     Education.education_id,
29     Job.job_id,
30     Manager.manager_id,
31     Address.address_id,
32     proj_stg.start_dt,
33     proj_stg.end_dt
34 from proj_stg
35 inner join Employee
36 on proj_stg.emp_id = Employee.emp_id
37 inner join Department
38 on proj_stg.department_nm = Department.department_name
39 inner join Salary
40 on proj_stg.salary = Salary.salary
41 inner join Education
42 on proj_stg.education_lvl = Education.education_level
43 inner join Job
44 on proj_stg.job_title = Job.job_title
45 inner join Manager
46 on proj_stg.manager = Manager.manager_name
47 inner join Address
48 on proj_stg.city = Address.city_name
49 inner join Location
50 on proj_stg.location = Location.location
51 inner join State
52 on proj_stg.state = State.state_name
53
```

CRUD

- Question 1: Return a list of employees with Job Titles and Department Names

```
1  select distinct
2      Employee.*/,
3      Job.job_title,
4      Department.department_name
5  from EmployeeInfo
6  join Employee
7  on EmployeeInfo.emp_id = Employee.emp_id
8  join Department
9  on EmployeeInfo.department_id = Department.department_id
10 join Job
11 on EmployeeInfo.job_id = Job.job_id
```

emp_id	emp_name	emp_email	emp_hiredate	job_title	department_name
E10033	Jermaine Massey	Jermaine.Massey@TechCorp.c...	2016-03-07	Software Engineer	Product Development
E10407	Darshan Rathod	Darshan.Rathod@TechCorp.com	2018-10-08	Sales Rep	Product Development
E11678	Colleen Alma	Colleen.Alma@TechCorp.com	2001-12-26	Network Engineer	Product Development
E11920	Sharon Gillies	Sharon.Gillies@TechCorp.com	2006-06-19	Sales Rep	Sales
E12397	Daniel Matkovic	Daniel.Matkovic@TechCorp.com	2013-11-17	Network Engineer	Product Development
E12562	Keith Ingram	Keith.Ingram@TechCorp.com	1996-04-14	Administrative Assist...	Product Development
E12890	Robert Brown	Robert.Brown@TechCorp.com	2010-06-06	Software Engineer	Product Development
E13085	Susan Cole	Susan.Cole @TechCorp.com	2017-05-01	Shipping and Receivi...	Distribution
E13160	Eric Baxter	Eric .Baxter@TechCorp.com	2008-10-06	Database Administra...	IT
E13160	Eric Baxter	Eric .Baxter@TechCorp.com	2008-10-06	Network Engineer	Product Development
E13596	Kenneth Dewitt	Kenneth.Dewitt@TechCorp.com	2012-04-09	Sales Rep	Product Development
E14737	Juan Cosme	Juan.Cosme@TechCorp.com	2012-07-22	Shipping and Receivi...	Distribution
E14913	Aaron Gordon	Aaron.Gordon @TechCorp.com	1998-07-15	Network Engineer	Product Development
E15267	Shanteel Jackson	Shanteel.Jackson@TechCorp.c...	1998-02-22	Shipping and Receivi...	Distribution
E15292	Melinda Fisher	Melinda.Fisher@TechCorp.com	2011-02-06	Shipping and Receivi...	Distribution
E15292	Melinda Fisher	Melinda.Fisher@TechCorp.com	2011-02-06	Software Engineer	IT
E16276	Analyn Braza	Analyn.Braza@TechCorp.com	1996-03-07	Sales Rep	Sales
E16346	Jill Fram	Jill.Fram@TechCorp.com	2007-12-16	Administrative Assist...	Product Development
E16678	Abby Lockhart	Abby.Lockhart@TechCorp.com	2005-11-25	Database Administra...	IT
E16678	Abby Lockhart	Abby.Lockhart@TechCorp.com	2005-11-25	Network Engineer	IT
E16995	Wilson Martinez	Wilson.Martinez@TechCorp.com	2001-02-28	Sales Rep	Product Development
E17054	Tyrone Hutchison	Tyrone.Hutchison@TechCorp.c...	2006-09-07	President	HQ
E17372	Greg Pratt	Greg.Pratt@TechCorp.com	2009-06-08	Sales Rep	Sales
E17469	Haifa Hajiri	Haifa.Hajiri@TechCorp.com	2003-12-17	Administrative Assist...	Distribution
E18659	Arnold Hanson	Arnold.Hanson@TechCorp.com	2008-01-27	Software Engineer	Product Development
E18697	Anita Deluise	Anita.Deluise@TechCorp.com	1995-06-01	Administrative Assist...	HQ
E20101	Zoey Bulter	Zoey.Bulter@TechCorp.com	2012-12-03	Sales Rep	Sales
E20848	Edward Eslser	Edward.Eslser@TechCorp.com	2006-07-26	Shipping and Receivi...	Distribution
E20848	Edward Eslser	Edward.Eslser@TechCorp.com	2006-07-26	Software Engineer	IT
E21348	Nital Thaker	Nital.Thaker@TechCorp.com	2016-09-28	Software Engineer	Product Development
E21696	Mallory Russo	Mallory.Russo@TechCorp.com	1997-12-06	Legal Counsel	Product Development
E22197	Oliver Jia	Oliver.Jia@TechCorp.com	2013-09-22	Network Engineer	IT
E22680	Cassidy Bancroft	Cassidy.Bancroft@TechCorp.c...	2013-10-26	Sales Rep	Product Development
E22785	Tami Smith	Tami.Smith@TechCorp.com	1997-12-08	Sales Rep	Sales
E23295	Lori Scatchard	Lori.Scatchard@TechCorp.com	2004-05-08	Sales Rep	Sales
E23295	Lori Scatchard	Lori.Scatchard@TechCorp.com	2004-05-08	Software Engineer	IT
E23429	Andrew Yoon	Andrew.Yoon@TechCorp.com	2018-11-14	Sales Rep	Product Development
E23590	Dennis Wooten	Dennis.Wooten@TechCorp.com	2013-04-06	Software Engineer	IT
E23669	Karen Brown	Karen.Brown@TechCorp.com	2017-04-23	Network Engineer	Product Development

CRUD

- Question 2: Insert Web Programmer as a new job title

```
1 1 insert into Job (
2   job_title
3 )
4 values ('Web Programmer')
5
```

job_id	job_title
abc Filter...	abc Filter...
1	Shipping and Receiving
2	Sales Rep
3	Administrative Assistant
4	Design Engineer
5	Database Administrator
6	Software Engineer
7	Manager
8	Legal Counsel
9	President
10	Network Engineer
11	Web Programmer

CRUD

- Question 3: Correct the job title from web programmer to web developer

```
1 update job
2 set job_title = 'Web Developer'
3 where job_title = 'Web Programmer'
```

job_id	job_title
abc Filter...	abc Filter...
1	Shipping and Receiving
2	Sales Rep
3	Administrative Assistant
4	Design Engineer
5	Database Administrator
6	Software Engineer
7	Manager
8	Legal Counsel
9	President
10	Network Engineer
11	Web Developer

CRUD

- **Question 4: Delete the job title Web Developer from the database**

```
1 delete from job
2 where job_title = 'Web Developer'
```

job_id	job_title
abc Filter...	abc Filter...
1	Shipping and Receiving
2	Sales Rep
3	Administrative Assistant
4	Design Engineer
5	Database Administrator
6	Software Engineer
7	Manager
8	Legal Counsel
9	President
10	Network Engineer

CRUD

- **Question 5: How many employees are in each department?**

```
1 select Department.department_name, count(Employee.emp_name)
2 from employeeinfo
3 join employee
4 on employeeinfo.emp_id = employee.emp_id
5 join department
6 on employeeinfo.department_id = department.department_id
7 group by department.department_name
```

department_na...	count
a bc Filter...	a bc Filter...
Product Development	70
HQ	13
Distribution	27
Sales	41
IT	54

CRUD

- **Question 6: Write a query that returns current and past jobs (include employee name, job title, department, manager name, start and end date for position) for employee Toni Lembeck.**

```
1  select
2      Employee.emp_name,
3      Job.job_title,
4      Department.department_name,
5      Manager.manager_name,
6      Employeeinfo.start_date,
7      Employeeinfo.end_date
8  from Employeeinfo
9  join Employee
10 on Employeeinfo.emp_id = Employee.emp_id
11 join Job
12 on Employeeinfo.job_id = Job.job_id
13 join Department
14 on Employeeinfo.department_id = Department.department_id
15 join Manager
16 on Employeeinfo.manager_id = Manager.manager_id
17 where Employee.emp_name = 'Toni Lembeck'
18 order by start_date
```

emp_name	job_title	department_na...	manager_name	start_date	end_date
abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc Filter...	abc Filter...
Toni Lembeck	Network Engineer	IT	Jacob Lauber	1995-03-12	2001-07-18
Toni Lembeck	Database Administrator	IT	Jacob Lauber	2001-07-18	2100-02-02

CRUD

- **Question 7: Describe how you would apply table security to restrict access to employee salaries using an SQL server.**

In order to apply the table security to restrict access to employee salaries, there are several possible solutions:

- Restricting the access to the database by providing the authorization only to the allowed people.
- Create 2 separate tables, 1 with salaries and 1 without salaries, then restrict the access to the table with salaries information.