INFSCI 2710 – Database Management – Spring 2018

Homework#1 – Relational Model

This assignment is worth 100 points

***Question 1 (20 points):** Consider the relational database below. 1) What are the appropriate primary keys for each table/relation? 2) after you decide the primary key, what are the appropriate foreign keys for each table/relation? For each question, please **explain your answer** and explain what assumptions you had to make to identify the primary key. You may add extra attribution, but you need to justify your decision in detail.

1. **Employee (employee_id, person_name, zip, phone1, fax, email)**

   It is a possibility that employees' names might be duplicated. I have added an attribution of employee_id for each employee and made an assumption that they are different with each other. And I deleted attributes of street_address, city and state all of which can be determined by zip.

2. **Employment (employeement_id, person_name, salary)**

   It is a possibility that employments' names might be duplicated. I have added an attribution of employment_id for each employment which is different with each other.

3. **Company (company_name, zip, phone, fax)**

   I choose the company_name and zip to be primary keys because the situation of duplicated company names rarely happens in the same zip area. And I deleted attributes of city and state because they are duplicated with zip.

4. **Products (productID, productName, shipperID, unit, price)**

   I choose productID to be the primary key and make an assumption that every productID is different with each other.
   The foreign key in this relation is shipperID because it is the primary key of Shipper relation.

5. **Shippers (shipperID, shipperName, phone)**

   I choose shipperID to be the primary key and make an assumption that every productID is different with each other.

***Question 2 (20 points)**: Identify the relationships (One-to-one (1:1); One-to-many (1:m); Many-to-many (m:n)) between the following entities. Provide detailed explanation for each answer.

1. **Prescription/medication**

   m:n: one prescription can include many medications and one medication can be listed in many prescription.

2. **Student/course**

   m:n: one student can choose many courses and one course can be taken by many students.

3. **Student/academic advisor**

   1:m: one academic advisor can have many students but one student can have only one academic advisor.

4. **Bank account/customer**

   1:m: one customer can have many bank accounts but one bank accounts can belong to only one customer.

5. **Rental car / renter** (the person who rents a car)

   1:1: one renter can rent one car at a time and one car can be rented by one person.

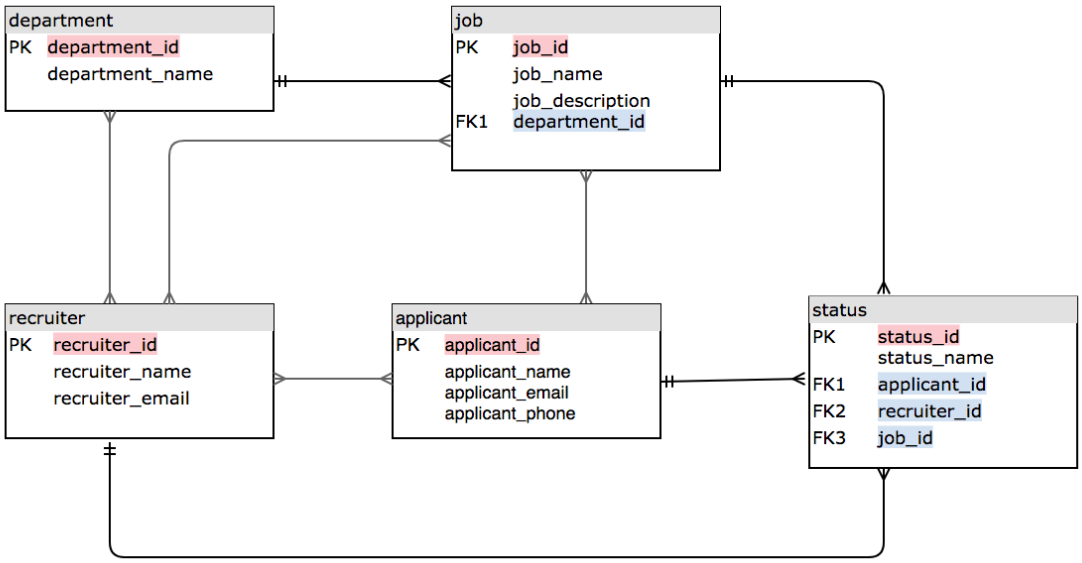
6. **Animal / food(s) that the animal consumes** (assume you are developing a database for a zoo)

   1:m: one animal can consume many foods but one food can be consumed by one animal.

***Question 3 (60 points):** Using draw.io diagram tool, design a database schema (at least five tables) for an application that tracks and manages job applications for a company. You have complete freedom in how you want to design your database, but you need to justify your decisions. Make sure to create proper primary/foreign key relationships between your tables. All entities and attributes must be named using the underscore convention. Highlight primary keys in red, foreign keys in blue. Explain your design decisions.

1. deparment (department_id, department_name)
2. job (job_id, job_name, job_description, department_id)
3. recruiter (recruiter_id, recruiter_name, recruiter_email)

4. applicant (applicant_id, applicant_name, applicant_email, applicant_phone)
5. status (status_id, status_name, applicant_id, recruiter_id, job_id)



I have designed 5 tables for this track and manage job applications application.

1. department: a company has many departments.
2. job: A department has many jobs to be applied for.
3. application: A job/department has many applications. An application can apply many jobs/departments in one company.
4. retruiter: A department/job has many retruiters and one retruiter is in charge of many departments/jobs. A retruiter is responsible for many appications' interview, and one application can be interviewed by many retruiters.
5. status: Using status table to track and manage application status which includes every application, recruiter and department. Every status can only correspond to one applicant, one recruiter and one job.