COMP20070 MySQL DB assignment (2023-24): Hospital Database

Student Name : Shlok Patel

Student Number : 22715709

Operating System : MacOS

Short Clear Description :

1. **Skills Details:**
    - **skill_id** (Primary Key): A unique identifier for a skill.
    - **skill name**: The name of the skill.
2. **Position_Skill:**
    - **position_id** and **skill_id** (Composite Primary Key): Ensures that each position can have multiple skills, and each skill can be associated with multiple positions.
    - **position** (Foreign Key): Links to the **Position Details** table, establishing a relationship between positions and skills.
    - **skill** (Foreign Key): Links to the **Skills Details** table, associating a skill with a position.
3. **Position Details:**
    - **position_id** and **hospital_hiring** (Composite Primary Key): Uniquely identifies a position in a hospital.
    - **type_of_pos**: Describes the type of position.
    - **position_hospital** (Foreign Key): Links to the **Hospital Details** table, specifying the hospital hiring for the position.
4. **Interview Details:**
    - **interview_id** (Primary Key): Uniquely identifies an interview.
    - **Position_id** and **Candidate_id** are both Not Null and allow foreign keys
    - **Interview_pos** and **interview_can** (Foreign Keys): Link to the **Position Details** and **Candidate Details** tables, respectively.
    - **Interview_date**: Records the date and time of the interview.
    - **offered_position**: Indicates whether a position was offered after the interview.
5. **Hospital Details:**
    - **Hospital Identifier** (Primary Key): Uniquely identifies a hospital.
    - **Hospital Name**: The name of the hospital.
    - **Address**: The address of the hospital.
    - **Telephone Number**: The contact number of the hospital.
6. **Candidate_Skill:**
    - **candidate_id** and **skill_id** (Composite Primary Key): Ensures that each candidate can have multiple skills, and each skill can be associated with multiple candidates.
    - **Cand_id** (Foreign Key): Links to the **Candidate Details** table, establishing a relationship between candidates and skills.
    - **Skill_id** (Foreign Key): Links to the **Skills Details** table, associating a skill with a candidate.

7. **Candidate Details:**
    - **candidate_id** (Primary Key): Uniquely identifies a candidate.
    - **firstname** and **surname**: The first name and surname of the candidate.
    - **address**: The address of the candidate.
    - **telephone**: The contact number of the candidate.

Assumptions or Additions :

Added 3 separate tables :
Position_Skill – to represent the fact that each position can require many skills
Candidate_Skill – to represent the fact that each candidate can have any skills
Skills Details – To represent each skill with a skill_id

I linked both the candidate_skill and position_skill to the skill details table. Originally I did not have this but realised this table is useful to prevent the duplication of data leaving it redundant as I would have to add skill_name to both candidate_skill and position_skill.

Another Design feature I noticed was originally I had a separate table for weather the candidate was offered a job but realised it would be easier to have an attribute in the interview details which would let you know whether or not the job was offered using a Boolean.

In my Interview Details I have the interview_id alongside the position_id and candidate_id which link the 2 tables so we know which candidate and which position they are applying for which intern tells you which hospital the position is for. The reason for not having the hospital_id as well is to prevent data redundancy.

The Candidate_skill table has a composite foreign key as one candidate can have many skills and the same skill is not repeated.

The Interview Details table has only 1 primary key as each interview_id must be uniqe but the candidate_id and position_id are not primary keys as you can have different people applying for the same position and you can have different candidates for a position.

My Position Details table has only 1 primary key as the position_id initially I had hospital_hiring as a primary key but later realised it was a mistake as each position_id can only be applied to one hospital.

Similar to the Candidate_skill table the position_skill table has a composite primary key as each candidate can have many skills.

Reaction Policies :

I added the “ON DELETE CASCADE” and “ON UPDATE CASCADE” clauses to the foreign key constraints. This means that if a referenced row is deleted or updated in the parent table, the corresponding rows in the child table will also be deleted or updated to maintain referential integrity

Candidate_skill :
Both are cascade in this situation because if a new candidate wants to add a new skill this should be allowed and if a candidate would like to remove a skill this should also be allowed.

Interview Details :
Both foreign keys are set to Cascade in this situation as well because if a new candidate wants to get an interview for a position this should be allowed and if a candidate wants to remove himself from an interview this should also be allowed.

Position Details :
Both cases are set to Cascade as if a position for a new hospital appears this should be allowed and if you want to delete a position for a hospital this should also be allowed.

Position_skill :
Both are cascade in this situation because if a position requires a new skill you should be able to add it and if you would like to delete a skill from a position this should also be enabled.

The Entity-Relationship (ER) diagram of your