

Short Description of Assumptions Made and Additional Information

The database has seven tables: candidate, candidate_skills, department, interview, position, position_skill_req and skills.

-Foreign key: Of these tables, candidate_skills, interview, position and position_skill_req have foreign key constraints:

1. Candidate_skills: The foreign keys constraints this table are the candidate identifiers in candidate and skills in skills.
2. Interview: Interview has foreign keys constraints for candidate identifier in candidate, and position identifier in position.
3. Position: Position has a foreign key constraint for department identifier in department.
4. Position_skill_req: position_skill_req has a foreign key constraint for skill in skill.

All the foreign keys have been set to cascade on update and delete. I decided to do this because it makes sense that if a candidate is removed from the database, any interviews they are down for should also be removed, or if we decided to rename the skill “administrative” to “administration”, it should be renamed in candidate skill and position skill requirements too.

-Primary keys:

1. Candidate: I chose candidate identifier here, as that should be unique for each candidate. I applied the same reasoning to the department, interview and position tables*.
2. Candidate_skills: I chose both skill and candidate identifier as the primary key here, as on this table a candidate's identifier will appear as many times as the number of skills they possess, but it will only appear once beside each of those skills.
3. Department*: Department identifier.
4. Interview*: Interview identifier.
5. Position*: Position identifier.
6. Position_skill_req: Both position_identifier and skill, as the position_identifier may not be unique as it will appear alongside each skill it requires, but only once beside each skill.
7. Skill: Just skill name, there is only one column in this table and a skill should not appear twice (set to unique just in case).

-In order to have it so that one candidate could have many skills and a position could require many skills, I decided to create three tables to handle skills and their relationships to candidates and positions: skills, candidate_skills and position_skill_req. Skills contains a list of all skills, candidate_skills is the relationship between candidate and skills and position_skill_req is the relationship between skills and position.

-When a query description in the assignment breakdown stated to find a 'candidate' or 'position', I interpreted this as an instruction to find the whole row for that particular candidate or position rather than a single value such as candidate or position identifier. When a single value was specified such as candidate identifier however, I have designed the procedure to only return the specified value.

Operating System Used: Windows 10

Diagram:

