Answer Key: JSQP

Section 1 (6 marks per question)

Q1. Explain the concept of data abstraction in the context of database systems, its levels, and their significance. Provide an example illustrating how data abstraction simplifies interaction with a database.

Keywords:

data abstraction, physical schema, logical schema, view level, data independence

Main Points:

• Hides complexity
• Levels of abstraction (physical, logical, view)
• Simplified interaction

Detailed Explanation:

Data abstraction simplifies interaction with the database by hiding the complex storage details and presenting a simplified view of the data at different levels. The levels include physical (how data is stored), logical (how data is organized), and view (how specific users see the data).

Examples:

• A user querying a database for student information doesn't need to know how the data is physically stored or the complex relationships between tables. They interact with a simplified view tailored to their needs.

Q2. Discuss the limitations of traditional file systems that led to the development of database management systems (DBMS). Provide specific examples of the problems caused by data redundancy and inconsistency in a file system.

Keywords:

file system, DBMS, data redundancy, inconsistency, data integrity, data security

Main Points:

• Data redundancy
• Inconsistency
• Data scattering
• Difficulty in access
• Integrity problems
• Security issues

Detailed Explanation:

Traditional file systems suffered from limitations like data redundancy, inconsistency, and difficulty in accessing data, which led to the development of DBMS. Data redundancy led to wasted storage and inconsistency issues where different copies of the same data held conflicting values.

Examples:

• In a file system storing student information, redundant data like addresses could be stored in multiple files. If a student moves, updating the address in every file becomes tedious and prone to errors, leading to inconsistencies.

Q3. Describe the components of the relational model and explain the significance of keys in ensuring data integrity. Illustrate with an example how primary and foreign keys establish relationships between tables.

Keywords:

relational model, relation, attribute, tuple, primary key, foreign key, data integrity

Main Points:

• Relations (tables)
• Attributes (columns)
• Tuples (rows)
• Keys (primary, foreign)
• Data integrity

Detailed Explanation:

The relational model uses relations (tables) with attributes (columns) and tuples (rows) to represent data. Keys are essential for data integrity. Primary keys uniquely identify tuples within a table, and foreign keys establish relationships between tables by referencing primary keys in other tables.

Examples:

• A 'Students' table has a primary key 'StudentID'. An 'Enrollments' table has a foreign key 'StudentID' referencing the 'Students' table, establishing a relationship between student information and enrollment details.

Q4. Explain the concept of mapping cardinalities in the Entity-Relationship (ER) model and their role in database design. Give examples of different mapping cardinalities (one-to-one, one-to-many, many-to-many) in a college database scenario.

Keywords:

ER model, entity set, relationship set, mapping cardinality, one-to-one, one-to-many, many-to-many

Main Points:

• Relationship between entity sets
• Cardinalities (1:1, 1:N, M:N)
• Database design

Detailed Explanation:

Mapping cardinalities in the ER model define the relationship between entity sets, indicating how many instances of one entity are associated with instances of another. These cardinalities (one-to-one, one-to-many, many-to-many) are crucial for database design, influencing table structures and relationships.

Examples:

• One-to-many: One department can have many students. Many-to-many: Students can enroll in many courses, and courses can have many students. One-to-one: Each student has one assigned advisor.

Q5. Compare and contrast specialization and generalization in the Extended ER (EER) model. Provide examples of how these concepts are used to model inheritance and hierarchical relationships in a database.

Keywords:

EER model, specialization, generalization, inheritance, hierarchy

Main Points:

• Specialization (top-down)
• Generalization (bottom-up)
• Inheritance
• Hierarchical relationships

Detailed Explanation:

Specialization (top-down) and generalization (bottom-up) are used in the EER model to represent inheritance and hierarchical relationships. Specialization creates sub-entities from a higher-level entity, while generalization combines similar entities into a higher-level entity.

Examples:

• Specialization: 'Person' entity can be specialized into 'Student' and 'Faculty' entities. Generalization: 'Car' and 'Truck' entities can be generalized into a 'Vehicle' entity.