Pakistan Institute
of Public Finance Accountants

Model Solutions

Database Management System (Application)

Summer Exam-2023

MODEL SOLUTIONS – DISCLAIMER

INTRODUCTION

The Model Solutions are provided to students for clear understanding of relevant subject and it helps them to prepare for their examinations in organized way.

These Model Solutions are prepared only for the guidance of students that how they should attempt the questions. The solutions are not meant for assessment criteria in the same pattern mentioned in the Model Solution. The purpose of Model Solution is only to guide the students in their future studies for appearing in examination.

The students should use these Model Solutions as a study aid. These have been prepared by the professionals on the basis of the International Standards and laws applicable at the relevant time. These solutions will not be updated with changes in laws or Standards, subsequently. The laws, standards and syllabus of the relevant time would be applicable. PIPFA is not supposed to respond to individual queries from students or any other person regarding the Model Solutions.

DISCLAIMER

The Model Solutions have been developed by the professionals, based on standards, laws, rules, regulations, theories and practices as applicable on the date of that particular examination. No subsequent change will be applicable on the past papers solutions.

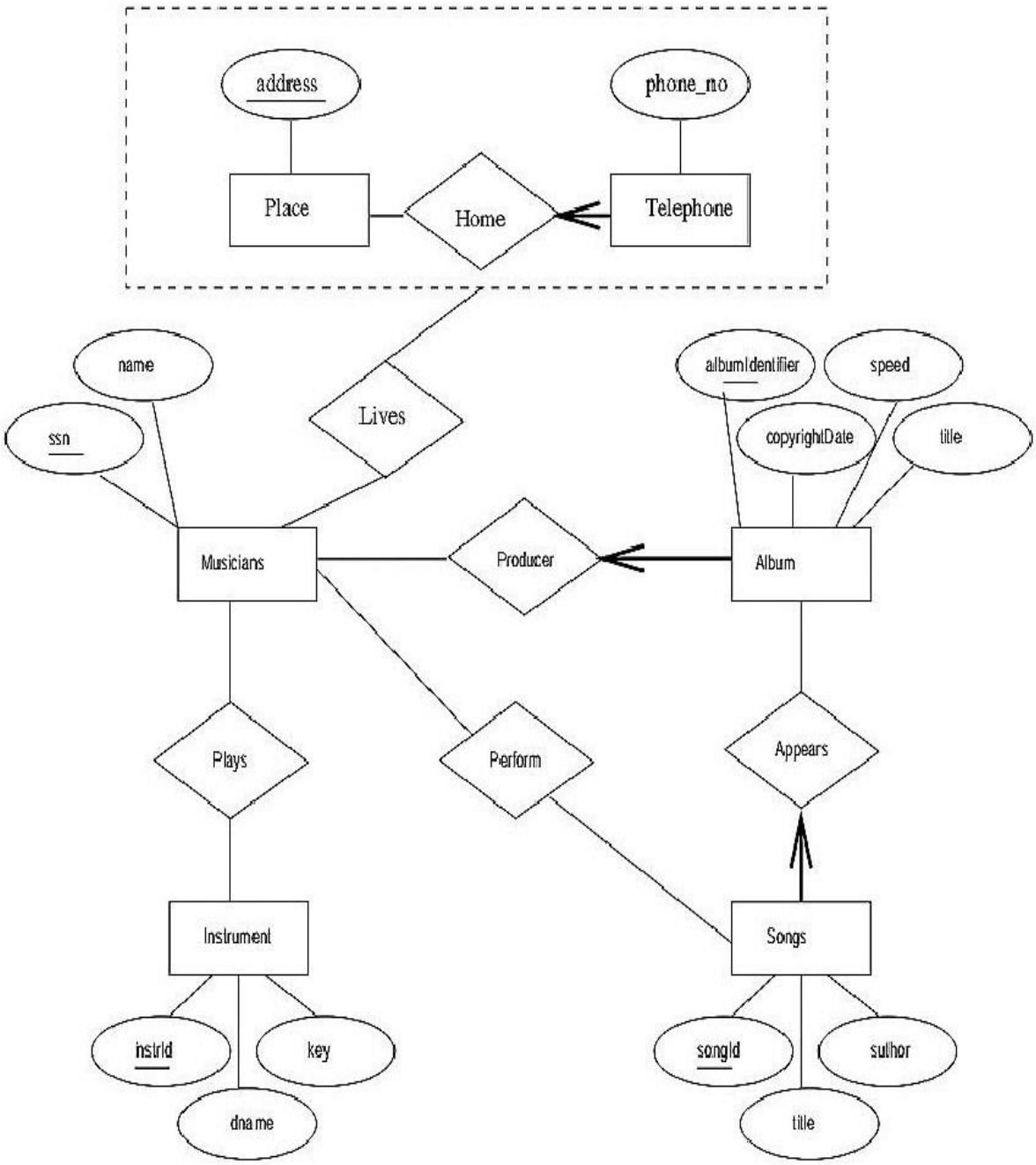
Further, PIPFA is not liable in any way for an answer being solved in some other way or otherwise of the Model Solution nor would it carry out any correspondence in this regard.

PIPFA does not take responsibility for any deviation of views, opinion or solution suggested by any other person or faculty or stake holders. PIPFA assumes no responsibility for the errors or omissions in the suggested answers. Errors or omissions, if noticed, should be brought to the notice of the Executive Director for information.

Summer Exam-2023
Solutions – Database Management System (Application)

Q.1. 10

(a)



Total Marks 14

Q.2. a To some extent it depends on your application, each database model has its own strength. For example, the document model is suitable for text or semi-structured data. On the other hand, if you have atomic data, the relational model is your best option. It also depends on which DBMS you use. Many DBMSs are built to work only with one particular model and the user does not have any other choices.

Summer Exam-2023
Solutions – Database Management System (Application)

Q.2. b

Description	Issue	Input	Output	Challenge
a. Conceptual Design Create model that captures major entities, relationships among entities, and attributes of entities required for a particular system.	– Capturing all data – Capturing relationships – Data integrity	– Functional specs – General understanding of problem	– ER diagram	-
b. Logical Design Transform the major entity/attribute /relationship requirements into high level specification for database	– Providing Location for all data – Data integrity	– ER diagram	– Relational database schema	-
c. Improving Logical Design Improve the high-level database specification.	– Minimizing redundancy – Minimizing ambiguity	– Relational database schema	– Relational database schema	-
d. Physical Design Transform the high-level specifications for database into detailed specifications for how to construct actual database in specific relational database software.	– Performance – Data integrity	– Relational database schema – Meaning of data	– Technical specifications for construction of the database	-

Total Marks 10

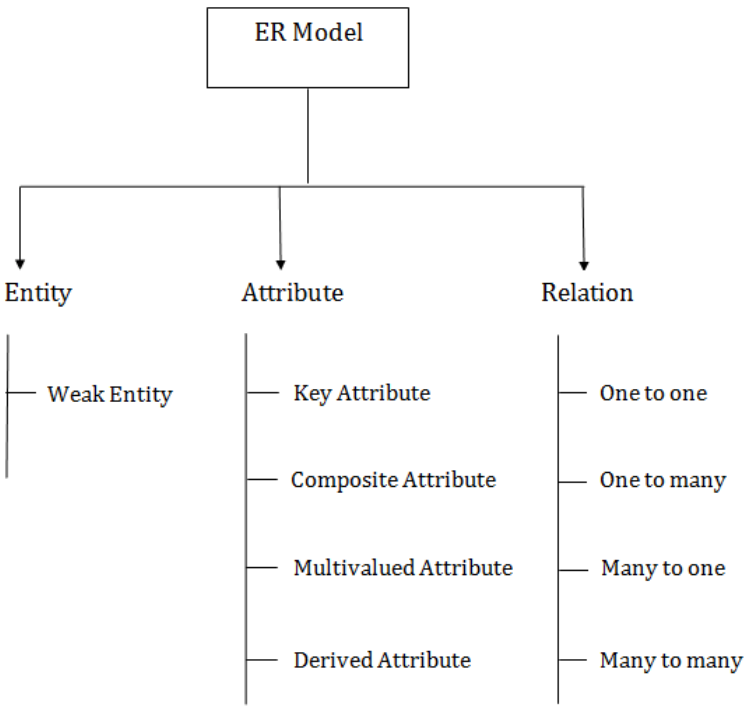
Q.3. (a) ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system.

It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.

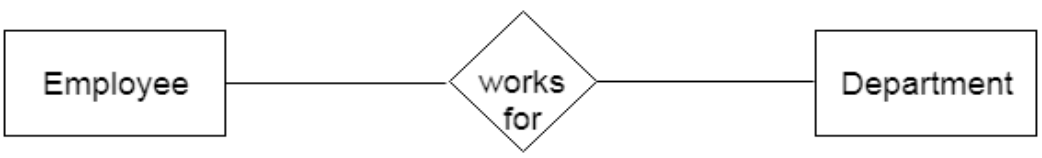
In ER modeling, the database structure is portrayed as a diagram called an entity-relationship diagram.

For example, Suppose we design a school database. In this database, the student will be an entity with attributes like address, name, id, age, etc. The address can be another entity with attributes like city, street name, pin code, etc and there will be a relationship between them.

Q.3. (b) 07

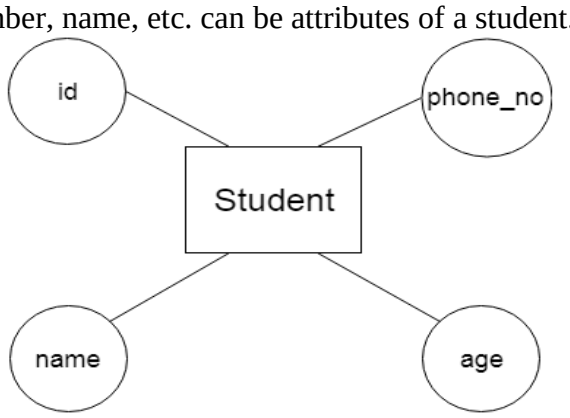


1. Entity:
An entity may be any object, class, person or place. In the ER diagram, an entity can be represented as rectangles.
Consider an organization as an example- manager, product, employee, department etc. can be taken as an entity.



a. Weak Entity
An entity that depends on another entity called a weak entity. The weak entity doesn't contain any key attribute of its own. The weak entity is represented by a double rectangle.

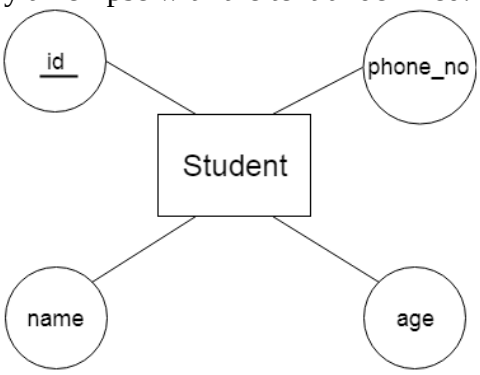
2. Attribute
The attribute is used to describe the property of an entity. Eclipse is used to represent an attribute. **For example**, id, age, contact number, name, etc. can be attributes of a student.



Summer Exam-2023
Solutions – Database Management System (Application)

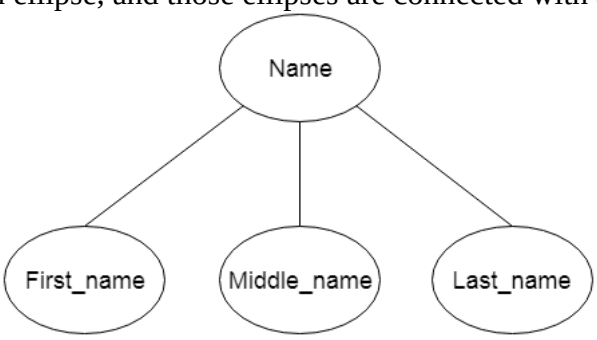
a. Key Attribute

The key attribute is used to represent the main characteristics of an entity. It represents a primary key. The key attribute is represented by an ellipse with the text underlined.



b. Composite Attribute

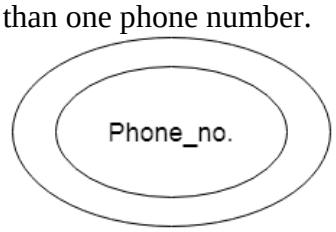
An attribute that composed of many other attributes is known as a composite attribute. The composite attribute is represented by an ellipse, and those ellipses are connected with an ellipse.



c. Multivalued Attribute

An attribute can have more than one value. These attributes are known as a multivalued attribute. The double oval is used to represent multivalued attribute.

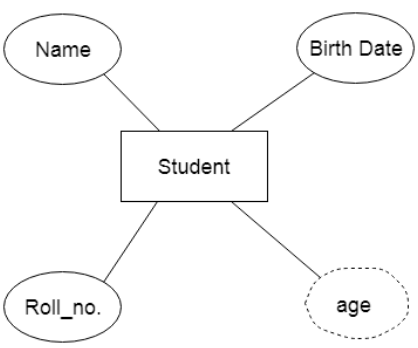
For example, a student can have more than one phone number.



d. Derived Attribute

An attribute that can be derived from other attribute is known as a derived attribute. It can be represented by a dashed ellipse.

For example, A person's age changes over time and can be derived from another attribute like Date of birth.



3. Relationship

A relationship is used to describe the relation between entities. Diamond or rhombus is used to represent the relationship.

Types of relationship are as follows:

a. One-to-One Relationship

When only one instance of an entity is associated with the relationship, then it is known as one to one relationship.

For example, A female can marry to one male, and a male can marry to one female.

b. One-to-many relationship

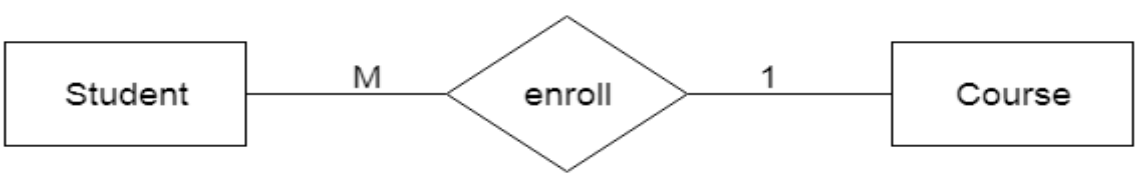
When only one instance of the entity on the left, and more than one instance of an entity on the right associates with the relationship then this is known as a one-to-many relationship.

For example, Scientist can invent many inventions, but the invention is done by the only specific scientist.

c. Many-to-one relationship

When more than one instance of the entity on the left, and only one instance of an entity on the right associates with the relationship then it is known as a many-to-one relationship.

For example, Student enrolls for only one course, but a course can have many students.



d. Many-to-many relationship

When more than one instance of the entity on the left, and more than one instance of an entity on the right associates with the relationship then it is known as a many-to-many relationship.

For example, Employee can assign by many projects and project can have many employees.

Total Marks 10

Q.3. (c) **DATABASE DESIGN PROCESS**

The database design is very important and creative activity. The major objective of database design is to map the conceptual data model to an implementation model that a particular DBMS can process. The database and database system must be acceptable to the organization and all its users. The users can easily perform different operations on the database and no problem can be created for users.

Phases in Database Development Process

The database development process includes a series of phases. The major phases are: planning, analysis, design, and implementation. Each phase is divided into steps. The phases of database development process are:

- **Planning**

  The database planning phase begins when a customer requests to develop a database system. It is a set of tasks or activities. It decides the resources required in the database development. It also decides the time limits for the completion of the system.

- **Analysis**

  Analysis is done in order to understand or study the current system. It is very important activity for the development of database system. In this phase, the requirements and expectations of the users are collected and analyzed. The collected requirements help to understand the current system and for the improvement of that system (or for designing the new system).

- **Database Design**

  The database design is very important step of database development process. In this phase, the database structure is designed.

  Database design is divided into two steps:

  - **Logical Database Design**

    In logical database design, the conceptual data model (or logical data model) is converted into database structure for a specific DBMS. If there is a relational DBMS, then the conceptual data models are mapped to the normalized relations.

  - **Physical Database Design**

    In Physical database design, the logical database design is converted into physical storage structures such as files and tables. The indexes and access methods are also specified. Similarly, physical design is also concerned with security, backup and recovery etc.

- **Implementation**

  After the design phase and selecting a suitable DBMS, the database system is implemented. The purpose of this phase is to install and run the database system.

  In database implementation phase, the ***database administrator (DBA)*** normally requires a server computer. The *DBA* may also need the services of network administrator to connect the users with the server. The users can share information through the server (database server).

Total Marks 06

Q.4. (a) In terms of inter-dependency, there are two types of sub-queries, in one of them, the inner query is dependent to the value of the outer query, we call this kind of queries “correlated” queries and in the other one, inner and outer queries are independent, which we call them “non-correlated” queries.

It is needless to mention that correlated sub-queries are very slow because it requires the inner sub-query to run once for every row in the outer query

Q.4. (b) There are three major algorithms to perform JOIN. Here I try to describe them briefly and mention their advantages. 05

Nested Loop: It compares all values of outer and inner tables against each other. It is the only algorithm that is capable of cross join (many-to-many joins). It serves as a fallback option in the absence of better algorithms.

Hash Join: This is the most versatile join method. In a nutshell, it builds an in-memory hash table of the smaller of its two inputs, and then reads the larger input and probes the in-memory hash table to find matches. Hash joins can only be used to compute equi-joins. It is typically more efficient than nested loops, except when the probe side of the join is very small.

Sort-Merge Join: This algorithm, first, sorts two tables based on the join attributes. Then it finds the first match and scrolls down on two tables and merge the rows for the matching attributes

Q.5. (a) 12

Reduction rules:

Strong entity – all attributes of strong entity will be attributes of relation schema.
M-to-M relationship – separate table need to be created with the primary keys of all participating strong entity sets.
1-to-M relationship – primary key of one side entity is included as foreign key in many side entity set.

ER Component	Reduced into Relational Schemas
Strong Entity	
Students	Students(SID, Name, Address)
Teams	Teams(TID, Name, Ranking)
Games	Games(GID, Score, Date)
M-to-M Relationship	
Binary Membership	Membership(SID, TID)
1-to-M Relationship	
is Captain	No separate schema. But the many side's strong entity set is added with one side's primary key. Hence teams becomes as follows. SID in teams schema is given a special name. Teams(TID, Name, Ranking, **Captain**)
Host	The relationship is to represent the host team for a game. Hence, we add tid (***one side pk***) in games (***as foreign key***) with a special name host or host_team as follows; Games(GID, Score, Date, **Host_Team**)
Guest	The relationship is for mentioning the guest team for a game. Hence, we add tid (***one side pk***) in games (***as foreign key***) with a special name guest or guest_team as follows; Games(GID, Score, Date, Host_Team, **Guest_Team**)

Final set of relations as follows;

Students (SID, Name, Address)

Teams (TID, Name, Ranking, Captain) – here, **Captain is foreign key** refers SID of Students relation.

Membership (SID, TID) – here, **SID and TID are foreign keys** refer SID of Students and TID of Teams relations respectively

Q.5. (b) A database model is a logical structure of a database describing relationships and constraints to store and access data. Some common database models include:

- Relational model
- Hierarchical model
- Entity-Relationship (ER) model
- Document model
- Object-Oriented (OO) model

Total Marks 15

Q.6. i Reject, key constraint (6), referential constraint (as there is no value = 7 in site table) and domain constraint (115 not a string) are violated.

Q.6. ii Multiple entries have popularity =4, thus when we update their userid to 55, the key constraint will be violated

Q.6. iii Referential constraint violated no site with site id 1 exists

Q.6. iv Accept (first two tuple are deleted)

Q.6. v referential action is cascade on delete, this will lead to deletion of site 3 and 6 in sites and deletion of entries 1, 2, 5 from Entries table.

Total Marks 10

****