
Homework 1

Your Name: Naga Satya Silpa Annadevara

Student ID: A20517818

1. Understanding Concepts related to DB

a). Compare and introduce the differences between relational data models and ER data models (10 points)

Relational Data model:

Relational data model uses Tables(entities) to represent data and the relation between this data. These tables are referred to as 'Relations' In relational model.

Tables can have an unlimited number of rows, but they must have a specific number of columns that can link to each other by using Primary keys and foreign keys.

Each row in a table is known as a "Tuple," and it holds all the details about a certain table item. Records are collections of tuples; hence the relational model is sometimes known as a record-based model.

A table's columns are referred to as its attributes since they describe the table's characteristics (relation). Each attribute must have a domain that specifies the kind of data it may hold.

All three types of relations—one to many, many to one, and many to many—are often accommodated by relational data architectures.

ER Data model:

Entity relation models are used for Graphical representation of relation models.

A collection of entities, often known as real world objects, and the relationships between those entities may be described as the E-R Model. There should be no identikit entities. The database's conceptual view is provided through the E-R Model.

Entity set, relationship set, and characteristics combine to form the E-R Model. In this scenario, all entities together form an entity set, all relationships between entities together form an entity relationship set, and the attribute represents the attributes of the entities.

| Relational Data Model | ER Data Model |
|---|---|
| 1. It is used to display a group of tables as well as the connections between them. | 1. It is used to describe a group of things called Entities (which are also called Tables) as well as the connections between them. |
| 2. It is Table specific | 2. It Is entity specific |
| 3. This model does not cover cardinality mapping. | 3. The mapping of Cardinalities is described by this model. Cardinality is the measure of how distinct a row of data items is. |
| 4. It uses domain, attributes, and tuples to represent components. | 4. Entity, Entity Type, and Entity Set are represented as components. |
| 5. It is a linear data model without any modeling diagrams. | 5. It uses a mapping diagram for its representation. |
| 6. In Relational model, it is less difficult or more complex to understand the relation between tables. | 6. Understanding the relationships between entities is simpler using the E-R Model. |
| 7. It is the representational or implementation model. | 7. This model is theoretical or high-level. |

b). Explain the differences: strong entity, weak entity, associative entity (10 points)

Entity:

An Entity is nothing but a table which consists of a set of entity instances (rows) and Attributes (columns).

Example: Student Entity, Course Entity, Department Entity.

| Strong Entity | Weak Entity | Associative Entity |
|--|---|--|
| 1. Strong Entity is not dependent on any other entity. They are Independent. | 1. Weak entity cannot exist on its own, so it depends on strong entity | 1. The associated entity is also a weak entity since it lacks an independent existence. |
| 2. A primary key exists for every strong entity | 2. A weak entity is one that can't be uniquely recognized by only its qualities; as a result, it must combine a foreign key with those properties to get a primary key. | 2. An associating entity is one that is inserted between two other entities to break a M:N link. |
| 3. A single rectangle serves as a strong entity's indicator. | 3. With a double rectangle, a weak entity is indicated. | 3. A single rectangle serves as an associative entity's indicator. |

| | | |
|---|--|---|
| 4. Strong entities in ERD are represented by single line boxes. | 4. Weak entities are represented by double line boxes in ERD. | |
| 5. A single diamond denotes the relationship between two strong entities. | 5. A double diamond is used to symbolize the relationship between two opposing entities—one powerful and one weak. | |
| 6. Strong entity might or might not be fully involved in the connection. | 6. The identifying relationship depicted by the double line is always fully participated in by the weak entity. | 6. It is used to depict M: N connections more accurately. It can aid in the conversion of M: N connections to multiple 1:M relationships. |

c). Why we use index? What are the differences between primary key and unique index? (10 points)

Index:

- An index is a schema object that holds an entry for each value that appears in the table or cluster's indexed column(s) and allows for direct, quick access to rows.
- Indexes can help enhance SQL query performance.
- It may be set on a single attribute or a composite of attributes. Allowing duplicate and NULL values

Primary key:

It is a constraint placed on a table's attribute to prevent duplicate rows and to prevent Null values from being present. Every row value for every characteristic is required to be unique. It serves as a unique identifier for every record.

Unique Index:

Unique indexes make sure that no two rows of data in a database have the same key values, which helps to ensure data integrity.

| Primary Key | Unique Index |
|---|---|
| 1. a Primary key is a chosen candidate key that is used to distinctly identify any specific row; | 1. An index that can identify every row that is distinct, such as the unique values in a unique index, designed for rapid searches or queries |
| 2. One can make the primary key a unique index. If a primary key has already been generated, the database will set it as a unique index by default. | 2. Nonetheless, a unique index need not be a main key. |

| | |
|--|--|
| 3. There should only be one primary key per table. | 3. However, a table may have more than one unique index. |
| 4. Primary key don't allow Null values. | 4. Unique Index allow Null values. |
| 5. Primary key don't allow duplicate values | 5. Unique Index allows Duplicate values but only 1. |

3. (70 points) Given the following tables:

Customer Table [assume email is unique for each customer]

CustomerID, Name, Age, Gender, Email

Order Table

OrderID, CustomerID, OrderDate, OrderTotal, OrderPayment

OrderItems Table [describe items in an order. Note that there may be several items in a single order]

OrderID, ItemID, ItemQuantity, ItemUnitPrice

1). (15 points) For each table, list the following keys: super key, candidate key, primary key, secondary key, foreign key

Customer Table:

Super key:

Customer ID
 Customer ID + Name
 Customer ID + Age
 Customer ID + Gender
 Customer ID + Email
 Customer ID + Name + Age + Gender + Email
 Customer ID + Name + Age
 Customer ID + Name + Gender
 Customer ID + Name + Email
 Customer ID + Age + Gender
 Customer ID + Age + Email
 Customer ID + Gender + Email
 Email
 Email + Customer ID
 Email + Name
 Email + Age
 Email + Gender
 Email + Name + Age
 Email + Name + Gender
 Email + Age + Gender

Customer ID + Email

Candidate key:

Email,
Customer ID

Primary key:

Customer ID

Secondary key:

Email

Foreign key:

None

Order Table:

Super key:

Order ID
Order ID + Customer ID
Order ID + Customer ID + Order Date
Order ID + Customer ID + Order Date + Order Total
Order ID + Customer ID + Order Date + Order Total + Order Payment
Order ID + Order Date
Order ID + Order Date + Order Total
Order ID + Order Date + Order Total + Order Payment
Order ID + Order Total
Order ID + Order Total + Order Payment
Order ID + Order Payment
Order ID + Order Payment + Customer ID

Candidate key:

Order ID

Primary key:

Order ID

Secondary key:

None

Foreign key:

Customer ID

OrderItems Table:

Super key:

Item ID

Item ID + Order ID

Item ID + Order ID + Item Quantity

Item ID + Order ID + Item Quantity + Item Unit Price

Item ID + Item Quantity

Item ID + Item Quantity + Item Unit Price

Item ID + Item Unit Price

Item ID + Item Unit Price + Order ID

Order ID

Order ID + Any Other Attribute

Candidate key:

Item ID

Primary key:

Item ID

Secondary key:

None

Foreign key:

Order ID

Item ID

2). (20 points) List relationships between Customer and Order, Order and OrderItems, and explain why. Note: you do not need to draw the diagram

Relationship between Customer and Order:

Customer → Order

from left to right: 1 customer can have many orders. So the relationship between Customer and Order from left to right is 1(mandatory) : M(optional)

From right to left: 1 Order can be handed over to only 1 Customer. So the relationship between Order and Customer from right to left is 1(mandatory) : 1

So, the relation between Customer and Order is 1(mandatory) : M(optional)

Relationship between Order and Order Items:

Order → Order Items

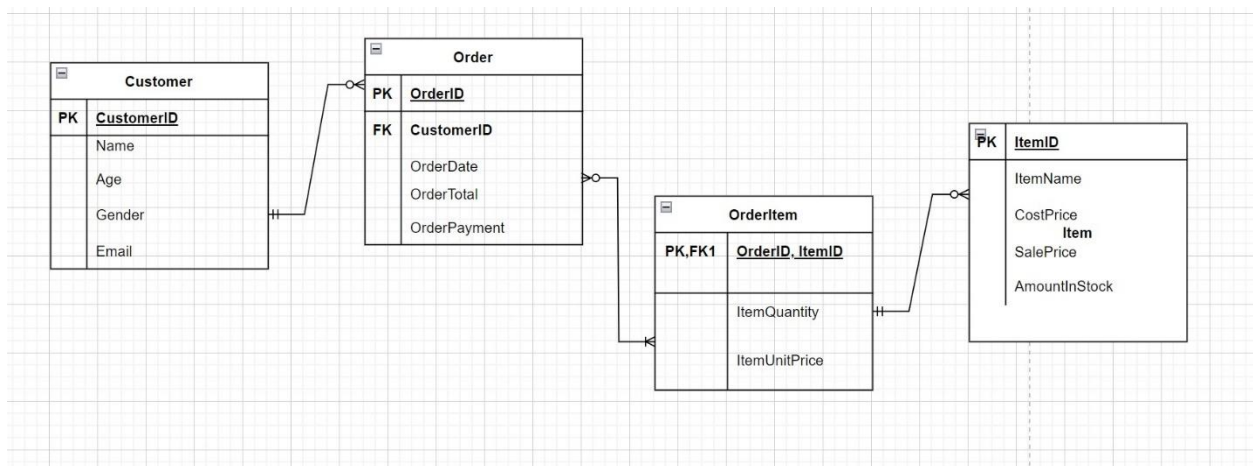
From left to right: 1 order can have many order items. So, the relationship between order and order items from left to right is 1(mandatory) : M(optional)

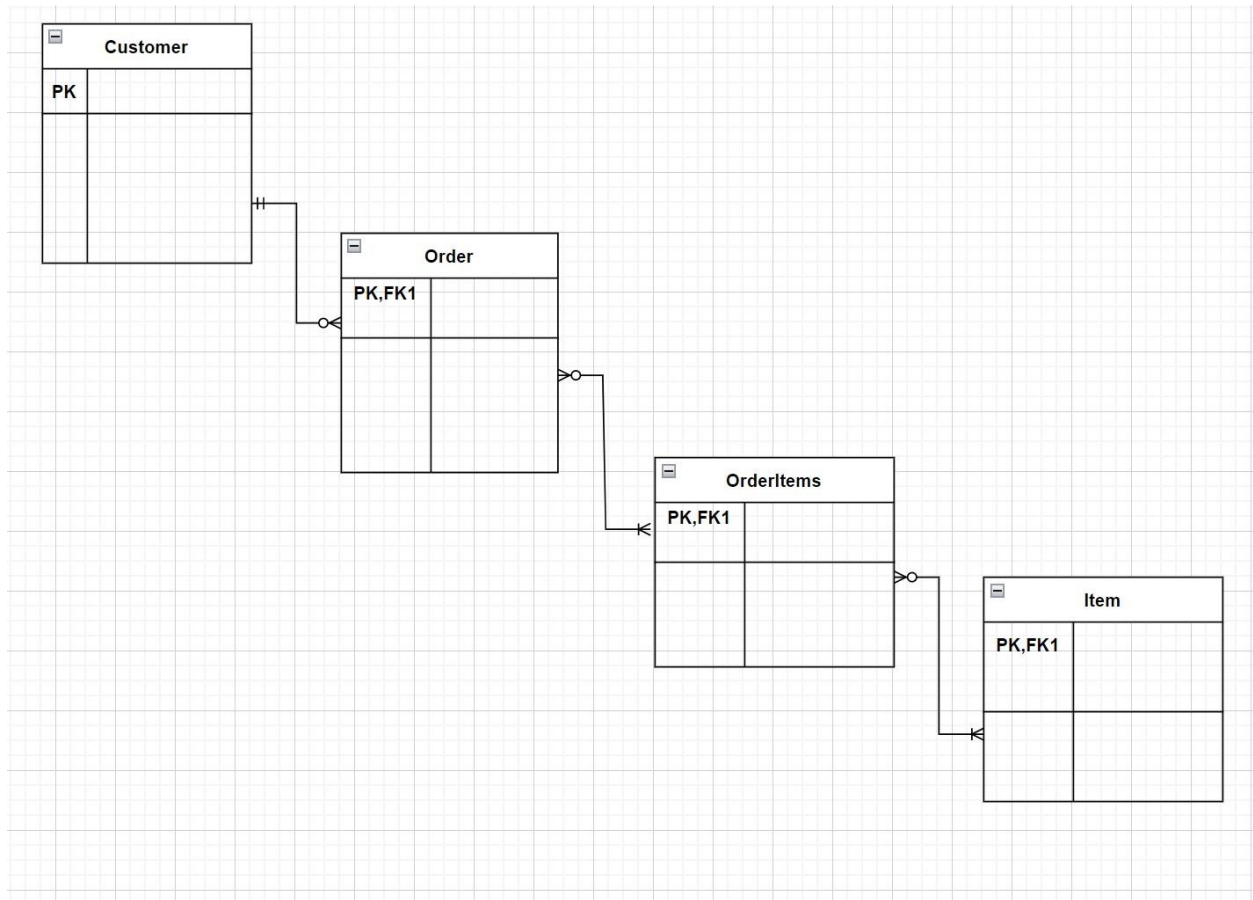
From right to left: 1 item can be in many orders. So the relation between order and order items from right to left is 1: M

So, the relation between order and order items is M(optional) : N(mandatory)

3). (35 points) Assume we have another table – Item with attributes (ItemID, ItemName, CostPrice, SalePrice, AmountInStock), complete DB conceptual design (based on all of these entities) by using ERD with Crow's foot notations. Note: you can use draw.io to draw ERD and you must provide ERD images (PNG or JPG) in the document

DB Conceptual design:





My Grading for this assignment is 91.

Incorrect answers:

HW1

Q3. 1. -5 Incorrect keys provided for the below tables

OrderItems Table : Foreign Key, Super Key, Secondary Key, Primary Key, Candidate Key

Q3. 2. -2 Minimum cardinality (constraints) for order and OrderItem is incorrect

Q3. 3. -2 Crow's foot notations between order and orderitems is Incorrect.