Database Fundamentals

The purpose of this set of Power Point slides and videos is to introduce relational database terms and concepts. Knowing these terms and concepts is essential to understanding how to work with relational databases—whether you need to build one, modify one, or use one.

Directly below is a list of the Power Point presentations and corresponding videos in the sequence they are designed to be used.

1. MEC Database Fundamentals – Intro
2. MEC Database Fundamentals – Data Model
3. MEC Database Fundamentals – Relationships
4. MEC Database Fundamentals – Defining Relationships
5. MEC Database Fundamentals – Physical Design Part 1 – Logical & Physical Model
6. MEC Database Fundamentals – Physical Design Part 2 – Primary & Foreign Keys
7. MEC Database Fundamentals – Physical Design Part 3 – 1-to-many, 1-to-1
8. MEC Database Fundamentals – Physical Design Part 4 – Many-to-many
9. MEC Database Fundamentals – Physical Design Part 5 – Multivalued attribute
10. MEC Database Fundamentals – Test Your Knowledge

The following pages list the terms and concepts covered in each presentation and some notes for teaches.

Concepts and terms covered in the Database Fundamentals series.

**MEC Database Fundamentals – Intro**

* Database
* Relational database
* E. F. Codd
* Peter Chen
* Data model
* Entity
* Relationship
* Entity-relationship diagram (ERD)
* Information versus Data

**MEC Database Fundamentals – Data Model**

* Crow’s feet notation
* Entity and entity type
* Entity instance
* Attribute
* Identifier
* Multivalued attribute
* Relationship
* Minimum and maximum cardinality

**MEC Database Fundamentals – Relationships**

* Relationships and cardinality
* Maximum cardinality
  + One-to-many
  + Many-to-many
  + One-to-one relationships
* Degree of the relationship
  + Unary
  + Binary
  + Ternary

**MEC Database Fundamentals – Defining Relationships**

* Questions to ask that help determine the minimum and maximum cardinality

**MEC Database Fundamentals – Physical Design Part 1**  **– Logical & Physical Model**

* Logical model
* Physical model
  + The example model created in Visio shows **required** attributes, ones that must be populated. This is a good opportunity to mention this characteristic about some attributes.
* Changes in terminology
  + Entity, table
  + Attribute, column or field
  + Entity instance, row or record
  + Identifier, primary key

**MEC Database Fundamentals – Physical Design Part 2 – Primary & Foreign Keys**

* How to implement the relationships show in a data model relationship in the database.
  + The example presented is for a one-to-many, binary relationship.
* Foreign key
* Data redundancy
* Foreign key name – The foreign key name doesn’t have to match the primary key name.

**MEC Database Fundamentals – Physical Design Part 3** **– 1-to-many, 1-to-1**

* Adding a primary key
  + This is a good time to add something about **candidate** keys and **surrogate** keys.
* Implementing relationships shown in the data model
  + Unary, one-to-many relationship
    - The EMPLOYEE table example has an example of two people with the same name. This is a good time to remind student why attributes, such as a person’s name, can’t be used as the primary key.
    - This is a situation when the name of the foreign key cannot match the primary key because they’re in the same table.
  + One-to-one relationship
    - Note that the one-to-one relationship example in this presentation does not fit with the auto repair shop database used in many examples. This is just a stand-alone example using LOCATION and EMPLOYEE tables.

**MEC Database Fundamentals – Physical Design Part 4 – Many-to-many**

* Implementing a many-to-many relationship
  + The example data model is based on the auto repair model. However, the REPAIR DETAIL entity is not shown initially and an INVENTORY entity has been added.
* Associative entity
  + This is an opportunity to discuss what **weak** entities are and **identifying** versus **non-identifying** relationships.

**MEC Database Fundamentals – Physical Design Part 5 – Multivalued attribute**

* Implementing a multivalued attribute.
  + This is another opportunity to discuss what **weak** entities are.
  + **Parent** and **child** entities
* Concatenated (composite) primary key

**MEC Database Fundamentals – Test Your Knowledge**

* This presentation asks questions about terminology and how to interpret the data models presented.

Questions & Answers:

1. What is the data model term that has this definition? *“A person, place, thing, or event about which we need to keep information.”*  **ENTITY**
2. What is a field called if it is used to link to a primary key field in another table? **FOREIGN KEY**
3. A field (or fields) that uniquely identifies each row in a table is called a **PRIMARY KEY**? **CANDIDATE KEY** would also be a valid answer.
4. If a relationship in the data model has a maximum cardinality of “one” on one end and “crows feet” on the other end, what type of relationship is this? **ONE-TO-MANY**
5. Which of the following could be a primary key for the EMPLOYEE table?

a. First Name + Last Name

b. Phone Number

c. Job Title

**d. Social Security Number**

e. Birth Date

*COMMENT: Though the SSN could serve as a primary key, it probably wouldn’t. It’s more likely that an employee ID would be assigned and the SSN would only be used for tax purposes.*

1. E. F. Codd proposed the principles for \_\_\_\_\_\_\_\_\_\_ databases, which is the type of database most widely used today. **RELATIONAL**
2. An attribute that might have more than one value for each entity \_\_\_\_\_\_\_\_\_, is called a \_\_\_\_\_\_\_\_\_\_ attribute. **INSTANCE, MULTIVALUED**

For the following questions, the student is first asked how tables would be built in the database given the data model shown. The second question is which entities/tables would have a foreign key. When a new table name is provided for the associate table, it could, of course, be whatever you deem appropriate for the name.

1. How many tables? **4** (customers, orders, **order details**, inventory)

*COMMENT: The many-to-many relationship between ORDERS and INVENTORY requires another table be added in the database to connect these two tables.*

*.*

1. Which gets a foreign key? **ORDER**
2. How many tables? **3** (employees, assignments, projects)
3. Which gets a foreign key? **ASSIGNMENT**
4. How many tables? **6** (vendors, **vendor-phones**, **vendor-parts**, parts, **BOM-parts**, bills of materials)

*COMMENT: This one is tricky. There is a multivalued attribute in VENDOR that becomes a separate table. The two many-to-many relationships require two additional tables. If students answer 6 tables, that’s sufficient for this level of coverage. However, the many-to-many relationship between VENDOR and PARTS would likely require another table because VENDOR and PARTS would be related through PURCHASE and PURCHASE DETAIL rather than a single vendor-parts table.*

1. Which gets a foreign key? **None**

*COMMENT: This version of the data model shows only the entity names, no attributes. Because the two relationships shown are many-to-many, neither entity in the relationship gets a foreign key. The foreign keys would be in the associative entities.*