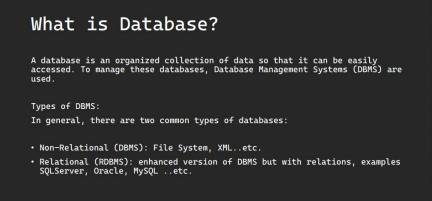
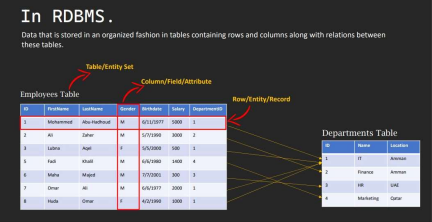
DATA BASE-Course15

What is Database? (20:55)

1

DBMS and RDBMS are NOT the same.

DBMS stands for Database Management Systems. DBMS has NOT relations between Data.

RDBMS has relations between data.

RDBMS stands for Relational Database

Management Systems .

DBMS has two types: None Relational Database and Relational Database.

2

File System, XML are samples of DBMS.

SQL Server , Oracle , MySQL are examples of RDBMS.

Dealing with RDBMS is much easier than

dealing with DBMS.

In RDBMS: Data that is stored in an organized fashion in tables containing rows and columns along with relations between these tables.

Column/Field/Attribute are the same.

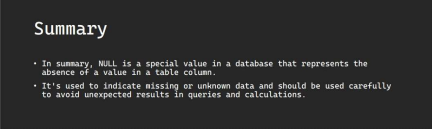
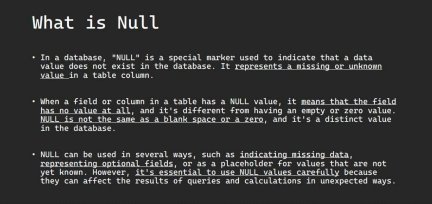
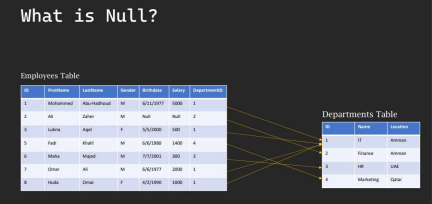
Row/Entity/Record are the same.

Table/Entity Set are the same.

Relationships are represented using references to data from other tables.

What is NULL? (8:33)

3

4

NULL is a special value in a database that represents the absence of a value in a table column.

NULL is used to indicate missing or unknown data and should be used carefully to avoid unexpected results in queries and calculations.

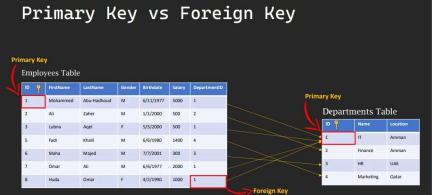
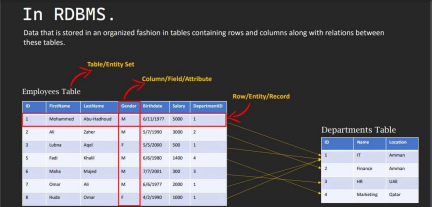
When a salary field in a table has a NULL value it means its value is missing or unknown.

When a field in a table has a NULL value it means its optional.

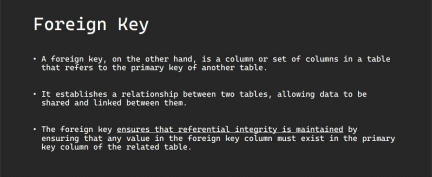
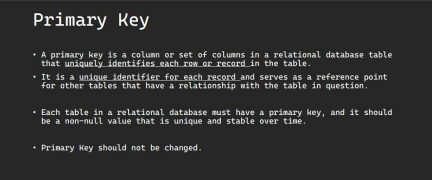
NULL is not the same as a blank space or a zero.

When a NumberOfChildren field in Employees table has a NULL value this means that we don't know yet that value, it's missing or unknown to us.

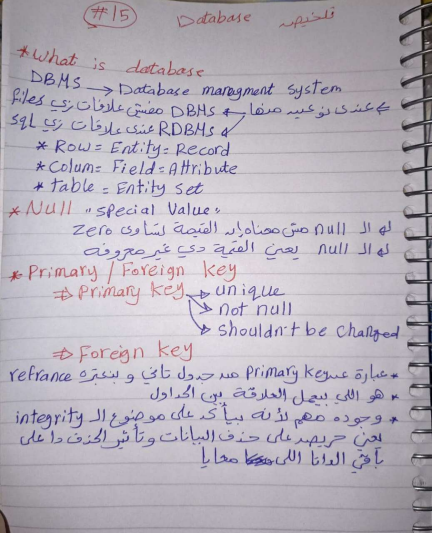
5

Primary Key vs Foreign Key / Referential Integrity (24:36) 

6



7



In summary, a primary key uniquely identifies a record in a table, while a foreign key establishes a relationship

8

between two tables by referencing the primary key of another table.

Primary Key can be more than one filed.

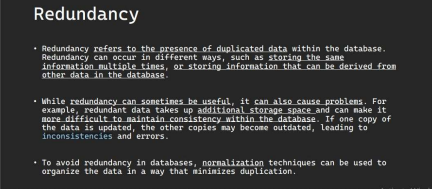
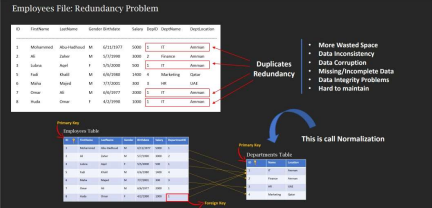
It's better to use numbers as primary keys because they are fast in search.

Primary Key should be Unique, Not NULL , and should not be changed.

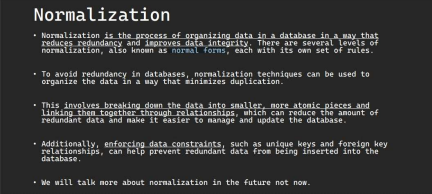
Value in the foreign key column must exist in the primary key column of the related table.

What is Redundancy? and why it's a problem? (19:22)

9



10



Redundancy refers to the presence of duplicated data within the database.

Redundancy can occur in different ways, such as storing the same information multiple times, or storing information that can be derived from other data in the database.

Redundancy can sometimes be useful.

Redundant data takes up additional storage space.

Redundancy can make it more difficult to maintain consistency within the database. If one copy of the data is

11

updated, the other copies may become outdated, leading to inconsistencies and errors.

To avoid redundancy in databases, normalization techniques can be used to organize the data in a way that minimizes duplication.

Normalization is the process of organizing data in a database in a way that reduces redundancy.

Normalization improves data integrity.

Normalization involves breaking down the data into smaller, more atomic pieces and linking them together through relationships, which can reduce the amount of redundant data and make it easier to manage and update the database.

Enforcing data constraints, such as unique keys and foreign key relationships, can help prevent redundant data from being inserted into the database.

Constraints are Restrictions/Rules on Data.

What is Data Integrity? and Why it's Important and Critical? (21:38)

12



13



Data integrity refers to the accuracy, consistency, and reliability of data over its entire life cycle, from creation to deletion.

Data Integrity refers to the assurance that data is complete, accurate, and trustworthy.

There are several factors that can impact data integrity, including human error, hardware or software failure,

14

security breaches, and data transfer errors.

There are different types of data integrity that organizations need to consider: Entity integrity, Referential integrity, Domain integrity, and Business integrity.

Entity integrity: This ensures that each row or record in a table is unique and can be uniquely identified. This is typically achieved through the use of primary keys.

Referential integrity: This ensures that relationships between tables are maintained and that there are no orphaned records. This is typically achieved through the use of foreign keys.

Domain integrity: This ensures that data is within acceptable ranges or values. For example, a date field should only contain valid dates, and a numeric field should only contain valid numbers.

Business integrity: This ensures that data meets business rules and requirements. For example, a bank might have rules around minimum and maximum account balances, or a hospital might have rules around patient data confidentiality.

15

What is Constraint? and Why it's Important? (13:47)

Constraints

• In the context of databases, constraints are rules or conditions that are

applied to the data to ensure its integrity and consistency. Constraints

can be applied to individual (فردي (columns or to entire tables, and they are used

to enforce various(عمتن (rules and restrictions (قيود (on the data.

• By using constraints, you can help ensure that your data is accurate (قةدق (,

consistent (ثابتة (, and easy to manage.

16

17

In the context of databases, constraints are rules or conditions that are applied to the data to ensure its integrity and consistency. Constraints can be applied to individual columns or to entire tables, and they are used to enforce various rules and restrictions on the data.

By using constraints, you can help ensure that your data is accurate, consistent, and easy to manage.

The constraint that ensures that a column or a set of columns uniquely identifies each row in a table. This constraint helps to enforce data integrity and ensure that there are no duplicate rows in the table is?

Primary Key Constraint

Which constraint ensures that the data in a column or set of columns meets a specified condition. This constraint helps to enforce data integrity and prevent invalid data from being inserted into the table?

Check Constraint

Which constraint establishes a relationship between two tables based on a key field. The foreign key constraint

18

ensures that data in one table matches data in another table, and it helps to maintain referential integrity in the database?

Foreign Key Constraint

Which constraint ensures that a column or set of columns cannot contain null (empty) values. This constraint helps to ensure that the data is complete and accurate, and it can help prevent errors in queries and calculations? Not Null Constraint

Which constraint ensures that the data in a column or set of columns is unique across all rows in the table. and allows Null as well, This constraint helps to enforce data integrity and prevent duplicate values from being inserted into the table?

Unique Constraint

Which constraint(s) ensures that the data in a column or set of columns is unique across all rows in the table. This constraint helps to enforce data integrity and prevent duplicate values from being inserted into the table.

Primary Key Constraint

19

Unique Constraint

Primary Key is Unique Constraint but does not allow NULL while Unique Constraint allows NULL.

Both Primary Key Constraint and Unique Constraint prevent duplicates

To achieve Entity integrity we use:

Primary Key Constraints

To achieve Referential integrity we use:

Foreign Key Constraints

To achieve Domain integrity we use:

Check Constraints

What is SQL? (17:51)

20

21

22

SQL stands for Structured Query Language.

SQL is Pronounced as “S-Q-L” or sometimes as “See-Quel”. SQL is used to communicate with a database.

Database management systems that use SQL are: Oracle, Sybase, Microsoft SQL Server, Access, Ingres, etc.

SQL can execute queries against a database. SQL can retrieve data from a database.

SQL can insert and update records in a database. SQL can delete records from a database.

SQL can create new databases, Tables, Views ..etc in the database

SQL can set permissions on tables, procedures, and views.

5 Types of SQL Statements :

Data Definition Language (DDL)

Data Manipulation Language (DML)

23

Data Control Language (DCL)

Transaction Control Language (TCL)

Data Query Language (DQL)

DBMs vs RDBMS Summary (8:50) 

24



25

Database Design: Conceptual Design

What is ERD? and Why? (20:22) 26

27



An Entity Relationship Diagram (ER Diagram) pictorially explains the relationship between entities to be stored in a database.

ER Diagram is a structural design of the database. ER Diagram is a conceptual design of the database.

ER diagram is created based on three principal 28

components: entities, attributes, and relationships.

An Entity-Relationship Model represents the structure of the database with the help of a diagram.

ER Modelling is a systematic process to design a database as it would require you to analyze all data requirements before implementing your database.

The cost of updating ERD is much cheaper than the cost of updating Database after you build it.

ER Diagram helps you conceptualize the database and lets you know which fields need to be embedded for a particular entity.

ER Diagram gives a better understanding of the information to be stored in a database.

ERD reduces complexity and allows database designers to build databases quickly.

ER Diagram in RDBMS is widely used to describe the conceptual design of databases.

ERD helps both users and database developers to preview 29

the structure of the database before

implementing the database.

ERD Symbols (12:33)

30



Which ERD symbol represents entity types? Ellipses

Rectangles

Diamonds

Double Ellipses

31

Which ERD symbol represents attributes? Rectangles

Diamonds

Ellipses

Lines

Which ERD symbol represents relationship types? Diamonds

Ellipses

Rectangles

Double Ellipses

Which ERD symbol links attributes to entity types and entity types with other relationship types?

Rectangles

Ellipses

Diamonds

Lines

32

How to represent primary key in ERD?

Underline the attributes name.

Strong Entity is the entity that has primary key(s).

Weak Entity is represented by double rectangles because it has no primary key for it and it depends on other entities to be identified.

Components of ERD (2:27)

33

Entity (Strong) and Weak Entity (9:25) 

34

35

There are two types of entities, strong entities and weak entities.

Strong Entity has to have primary key.

Weak Entity has no primary key(s)

An entity can be either a living or non-living component.

Which of the following assures the Entity Integrity? Strong Entities

Weak Entities

Strong Entity is represented by single rectangle in ERD. Weak Entity is represented by double rectangles in ERD.

Attributes (7:08)

36

37



An attribute exhibits the properties of an entity.

You can illustrate an attribute with an oval shape in an ER diagram.

Key attribute uniquely identifies an entity from an entity set.

Key Attribute is represented by underlining the text of a 38

key attribute.

An attribute that is composed of several other attributes is known as a composite attribute.

An oval showcases the composite attribute, and the composite attribute oval is further connected with other ovals.

Some attributes can possess over one value, those attributes are called multivalued attributes.

The double oval shape is used to represent a multivalued attribute.

It's not recommended to have multivalued attributes.

An attribute that can be derived from other attributes of the entity is known as a derived attribute.

In the ER diagram, the dashed oval represents the derived attribute.

At the end the attribute is a field or column in the database.

39

Relationships (16:39)



40

41



Relationship can be on one entity and we call it Self 42

Reference Relationship.

We can have more than one relationship between two entities.

One-to-One Relationship (16:21) 

43



44



When a single element of an entity is associated with a single element of another entity, it is called a one-to one relationship.

45

One-to-Many/Many-to-One

Relationship (15:01)



46



47



When a single element of an entity is associated with more than one element of another entity, it is called a one-to-many relationship.

When more than one element of an entity is associated with a single element of another entity, it is called a many to-one relationship.

Many-to-Many Relationship (6:26) 48



When more than one element of an entity is associated with more than one element of another entity, it is called a Many-to-Many Relationship.

᛿ل what : لازم سأل حالك لما دك تحدد طبعة العلاقة ب entities 2 is the maximum number I can take from bla bla وهكذا

49

Cardinality vs Ordinality (11:57)

50



Cardinality refers to the maximum number of times an instance in one entity can relate to instances of another entity.

Ordinality, on the other hand, is the minimum number of times an instance in one entity can be associated with an instance in the related entity. (in other words It specifies if it’s optional/mandatory/required or not).

(0,M) : the first number represents the ordinality and the second one represents the cardinality.

When you ask "What is the Max?" you are identifying the cardinality.

When you ask "What is the Min?" you are identifying 51

the ordinality.

Cardinality Symbols and Practices (29:44)

52



53

54

Total Vs Partial Participation (5:54) 55

Process of Creating ERD Step by Step - Small Project (26:57)

56

57

58



59



60



61

University ERD ProjectUniversity ERD.pdf

My University ERD Project

Uni small project.erdplus

62

Aggregation / Associative Entities (5:45)

63



An associative entity is a type of entity in a database that is used to model a many-to-many relationship between two other entities. It is also sometimes called a junction table, a linking table, or a cross-reference table.

By using an associative entity, we can represent complex relationships between entities in a structured and efficient way, without having to duplicate data or create confusing relationships between tables. It allows us to model many to-many relationships and avoid data redundancy, making it a useful concept in database design.

64

Representing relationship between an entity and a relationship which may be required in some scenarios. In those cases, a relationship with its corresponding entities is aggregated into a higher level entity. Aggregation is an abstraction through which we can represent relationships as higher level entity sets.

ا

ولا فهمت

Aggregation is a specialized form of association between two or more Entities in which each Entity has its own Existence.

In aggregation, the relation between two entities is treated as a single entity. In aggregation, relationship with its corresponding entities is aggregated into a higher level entity.

65

Generalization (13:26)



66



Generalization is the process of extracting common properties from a set of entities and create a generalized entity from it.

Generalization is Bottom-Up approach.

Specialization (15:38)

67



In specialization, an entity is divided into sub-entities based on their characteristics.

Specialization is a top-down approach.

Specialization is a top-down approach in which a higher 68

level entity is divided into multiple specialized lower level entities.

Relational Schema

What is Relational Schema? (13:58) 

69



A relational schema is a set of relational tables and associated items that are related to one another.

Relation schema defines the design and structure of the relation like it consists of the relation name, set of attributes/field names/column names. every attribute

70

would have an associated domain.

Convert Self Referential ==> Relational Schema (7:07)

71

72

Convert Composite-Multivalued Derived Attributes ==>

Relational Schema (15:19)

73

74



75



Convert One-to-One ==> Relational Schema

76



Convert One-to-Many/Many-to-One ==> Relational Schema

77



Convert Many-to-Many ==> Relational Schema

78

79



Convert Associative Entity to Relational Schema

80