DATABASE MANAGEMENT SYSTEM

Database

The collection of organized interrelated data/information to any purpose is known as database.
It is used to store, organize and extract data. Database can be computerized or non-computerized (manual).
A database may have single or multiple tables. The data in database are organized in rows and columns.
e.g. Telephone directory, marks ledger, attendance register, dictionary etc.

Data → Information
Processing

Database Management System (DBMS)

DBMS is a method that stores data, process them and provides the information in an organized form using database management software. It also allows user to modify, update, organize and retrieve information from the database

E.G.

- MS- Access
- FoxPro
- MySQL
- Oracle
- SQL Server
- dBase

Advantages

- Reduce data redundancy (duplication)
- Increase data consistency (constant nature)
- Greater data integrity (completeness: complete data can be retrieved)
- Improved data security
- High storage
- Concurrent data access
- File Sharing
- Time Save

Disadvantages

- Complex to understand and implement
- Many rules to follow
- Need of frequent modification and update because of changing technology.
- Possibility of data leakage as cyber crime increases
- Initial investment cost in hardware, software and training is high
- Cost for maintenance of software is high
- Complexity of backup and recovery
- Dependency of central database in centralized database may arise problem

RDBMS

Relational Database Management System (RDBMS) is most widely used software or program that stores database in multiple tables in the basis of a key field. It allows a user to view or create or retrieve records from multiple linked tables

- MS- Access
- Oracle
- MS SQL Server
- MySQL
- Microsoft Access

MS-Access

MS-Access is a relational database management system developed by Microsoft Corporation. It is used to store and manipulate large amount of data in the form of tables. It allows the user to create database and store data in multiple tables

Objects of MS Access

- Table
- Form
- Pages
- Modules
- Query
- Report
- Macro
- Table

Table

Table is the primary object of MS-Access. It is the building block of database that stores large volume of data in the form of rows of columns.

Creating Table

Different ways to create a table in ms access database are:

- Creating Table in Design View
- Creating Table by Using Wizard
- Creating Table by entering Data

Elements of Table

Field

A column of a table which stores a particular type of data is known as a field. It is the smallest unit of the database. It is also known as attribute.

field

SN	Fname	Date of birth	Address
SIT01	Kamal	1/11/2048	Kathmandu
SIT02	Rojina	5/9/2052	Lalitpur
SIT03	Mohan	9/13/2045	Kalimati
SIT04	Sujata	9/15/2045	Kirtipur
SIT05	Ronish	6/11/2052	Kathmandu

Record

The collection of multiple related fields which give a complete information of a particular object or person is known as record. It is also known as tuple.

record

SN	Fname	Date of birth	Address
SIT01	Kamal	1/11/2048	Kathmandu
SIT02	Rojina	5/9/2052	Lalitpur
SIT03	Mohan	9/13/2045	Kalimati
SIT04	Sujata	9/15/2045	Kirtipur
SIT05	Ronish	6/11/2052	Kathmandu

Primary Key and Foreign Key

The special field of table that uniquely identifies each record from database is known as the primary key of the table. It does not accept duplicate value and cannot be left blank.

Foreign Key is a field in a table that matches the primary key field (column) of another table in relation.

Importance of primary key

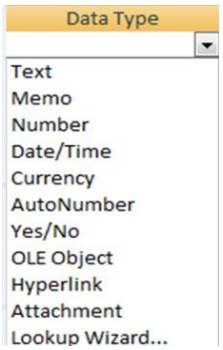
- To identify each record uniquely
- To control duplication data entry
- To set relationship between tables

Some other keys

- **Candidate key**: A Candidate Key can be any column or a combination of columns that can qualify as unique key in database. There can be multiple Candidate Keys in one table which can be taken as a primary key
- **Alternate key**: An alternate key is a key associated with one or more columns whose values uniquely identify every row in the table, but which is not the primary key.
- **Composite (Compound) Key**: A composite key, in the context of relational databases, is a combination of two or more columns in a table that can be used to uniquely identify each row in the table. Uniqueness is only guaranteed when the columns are combined; when taken individually the columns do not guarantee uniqueness.

Data Types

It is the feature of a field that determines what kinds of data can be entered in the field.



Text
Default data type which stores alphanumeric value.
Default size: 50 characters
Maximum size: 255 characters

Memo
Stores alphanumeric values upto 65535

Number
Stores only number
Default numeric data type: Long Integer

Sub data type for number

- **Byte** - For integers that range from **0** to **255**. Storage requirement is a single byte.
- **Integer** - For integers that range from **-32,768** to **+32,767**. Storage requirement is two bytes.
- **Long Integer** - For integers that range from **-2,147,483,648** to **+2,147,483,647**. Storage requirement is four bytes.
  [*Use the Long Integer data type when you create a foreign key to relate a field to another table's AutoNumber primary key field.*]
- **Single** - For numeric floating point values that range from **-3.4×10^{38}** to **$+3.4 \times 10^{38}$** and up to seven significant digits. Storage requirement is four bytes.
- **Double** - For numeric floating point values that range from **-1.797×10^{308}** to **$+1.797 \times 10^{308}$** and up to **15** significant digits. Storage requirement is **eight** bytes.
- **Replication ID** - for storing a **GUID** (Globally Unique Identifier) that is required for replication. Storage requirement is **16** bytes.
  [*Note that replication is not supported using the .accdb file format.*]
- **Decimal** - For numeric values that range from **$-9.999... \times 10^{27}$** to **$+9.999... \times 10^{27}$**. Storage requirement is **12** bytes. [**-10×10^{27}** to **$+10 \times 10^{27}$**]

Date/Time
Stores date and time
Size: 8 bytes

Currency
Stores currency value
Storage size: 8 bytes

Auto Number
Generates unique sequential number automatically
Size: 4 bytes

Yes/No
Stores logical value that can be only one of two possible values: yes/no, true/false, on/off etc.

OLE Object
Stores picture, video, sound
Size: 1 GB

Hyperlink
Stores link of other documents
Size: 2048 characters

Attachment (only in ms access 2007)
Attaches any document or file in the table

Lookup Wizard
Displays data from another table in list box or combo box.
Storage size: 4 bytes

Common Field Properties

Field Size: Maximum Number of characters
Format: Appearance of value to be displayed
Input Mask: It specifies a pattern and controls value of records for all data to be entered

Caption: alternate name for the field upto 2048 characters
Default Value: specifies the value entered automatically
Validation Rule: Controls and limits the value to be entered
Validation Text: Text messaged to be displayed when data entered is not matched to validation rule
Required: specifies data must be entered or not
Indexed: speeds up sorting and searching data
Some Validation rule

- <>0: not equal to zero
- 1000: equal to 1000
- <1000: less than 1000
- Between 5000 and 10000

Query

Query is an object of database that is used to view, retrieve, change and analyzed records from a table or multiple linked tables. When a user changes data in the query, the data in the table also get changed.

Different ways to create query are:

- Creating query in design view
- Creating query by using wizard

Types of Query

Select Query: Retrieves records from table
Crosstab Query: Presents data in a sheet that summarizes data from the fields in one or more tables and displays the dynaset (stands for dynamic set) in a datasheet format
Parameter Query: Retrieves data as parameter given and works in conjunction with other types of queries
SQL Query: Communicates with other types of database
Action Query: Performs action (makes changes) on the data retrieved

- **Update Query**: Updates specified value on the table
- **Append Query**: Adds or appends data from one table to another
- **Delete Query**: Deletes records from a table or multiple tables.
- **Make Table Query:** Makes new database table from the result of the query

FORMS

It is a database object that is used to create an interface for entering data in a table or multiple tables. A form displays a complete record at a time. Forms are comprised of following three sections:

- **Header**: Header may be Page header or Form header. Page Header is displayed when the form is printed and form header is displayed when the form is viewed in the form view.
- **Detail**: Detail area contains the entire contents of your form design.
- **Footer**: Footer may be page footer or form footer. Page footer is displayed when the form is printed and form footer is displayed when the form is viewed in the form view.

Different ways to create a form are:

- Creating form using Design View
- Creating form by using wizard

REPORTS

Report is a type of access database object that presents data in effective way in a printed format. It allows a user to print documents according to the user specifications of the summarized information through queries or tables. Similar to Forms Reports is also comprised of three sections:

Report Header: Appears at the top of the first page and displays the report title.
Page Header: Appears at the top of every page and displays the headings (field labels) for each column.
Page Footer: Appears at the bottom of every page and displays the page number and total number of pages.
Detail Section: Appears between the page header and page footer and displays the records from the table or query.
Report Footer: This section is optional. Appears on the last page of the report and displays summary information such as grand totals.

Different ways to create a report are:

- Creating report using Design View
- Creating report by using wizard

PAGES

Pages allow you to view and update the data in your database from within a browser. Although they are stored outside the Access database (.mdb) file, they are created and maintained in a manner similar to that of forms. Although the data access pages are targeted toward browser, they can also be previewed within the Access application environment. Data access Pages can also be viewed and modified in Design View. Design View of a Data access page is similar to that of a form, which makes working with the data access Pages and deployment of your application over an Intranet, very easy.

MACROS

Macros in Access differ from the macros of the other languages. They can't be recorded as in Word or Excel and are not saved as VBA code. With Access macros, you

can perform most of the task that you perform manually from keyboard, menus and toolbars. Generally **VBA** (**Visual Basic for Applications**) code contained in module are used rather than macros to do the task that your application must perform. VBA code gives more flexibility and power than macro.

MODULES

Modules are the foundation of any application and let you create ***libraries of functions*** that can be used throughout your application. Modules contains the VBA code and you can do the following using it:

- Perform error handling
- Declare and use variables
- Loop through and manipulate record sets
- Call Windows API and other library functions
- Create and modify system objects, like tables and queries
- Perform transaction processing
- Perform many functions not available with macros
- Test and debug complex process
- Create library database

BASIC DATABASE TERM

Entity (table): Entity is the distinguishable objects of real world. e.g. Student, Customer, Employee

Attribute (column): Set of properties possessed by an entity.: e.g. Name, Class, Section of a student

Tuple (row): Each record row in a table

Sorting: Arrangement of all the records in ascending or descending order either alphabetically or numerically

Database index: is a data structure that improves the speed of data retrieval operations on a database table at the cost of additional writes and the use of more storage space to maintain the extra copy of data.

Filtering of Data: Process of retrieving required record of a table based on given criteria

Formatting the table: Changing the appearance of to make it attractive. E.g. changing row height or column width, changing border color and style, changing font and its size

Design View: A window that shows the design of database objects: table, query, form, reports etc.
In this view, we can create new database objects and modify them.

Datasheet View: A window that displays data from a table, query, form in a row and column format. In this view, we can edit fields, add and delete data and search for data.

Wizard: Step by step process to perform any action

Relationship: An association among different related tables so that we can retrieve information from different tables easily.

Data Normalization: A process to make the table efficient and compact as possible to eliminate the possibility of inconsistencies, duplicate and error.

Database Administrator (DBA)

An information specialist who has responsibility for managing the database is called a DBA. His/ Her duties fall into four major areas: Database planning, implementation, operation and security

Front-End and Back-End Application:

Back-end application contains tables while front end application contains other objects like, queries, forms and reports.

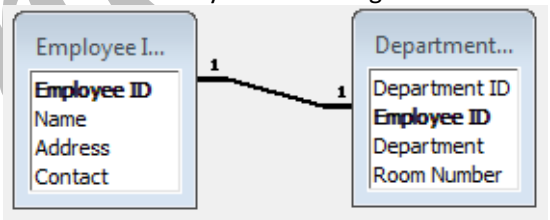
Control: Graphical objects like text box, check box, command button which performs certain action on form or report

Toggle Button: control which acts as on/off button

Relationship: relationship between two or more tables

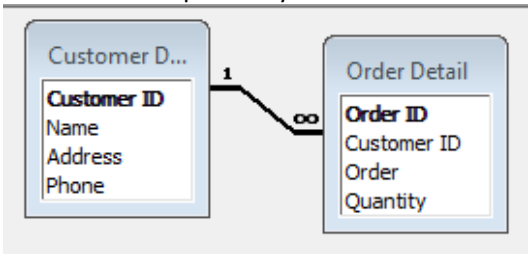
- One to One:
- One to Many
- Many to Many

One to One relationship: In a one-to-one relationship, each record in the first table can have only one matching record in the second table, and each record in the second table can have only one matching record in the first table.



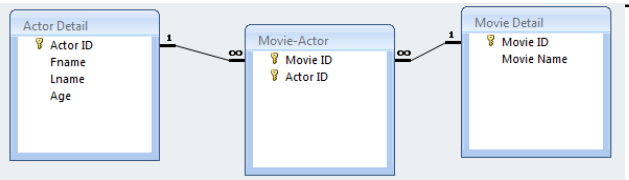
One to Many relationship: customer → order

A customer can place any number of orders.



Many to Many relationship: Actor → Movie

A single actor can play more than one movie. On the other hand, a single movie can have more than one actor. To represent a many-to-many relationship, we need create a third table, often called a junction table, that breaks down the many-to-many relationship into two one-to-many relationships. Insert the primary key from each of the two tables into the third table.



Data Models: A model is a representation of reality, 'real world' objects and events, and their associations

Data integrity: Data integrity refers to maintaining and assuring the accuracy and consistency of data over its entire life-cycle.

Confidentiality: Confidentiality is a set of rules or a promise that limits access or places restrictions on certain types of information.

Data redundancy: Data redundancy occurs in database systems which have a field that is repeated in two or more tables.

Data consistency: Data consistency is the accuracy, validity, usability and integrity of associated data. It ensures that users observe a continuous view of data and any changes made. It may fail at any time but can be recovered when backup copies substitute's original data.

Replication : Replication is the process of copying and maintaining database objects in multiple databases that make up a distributed database system

DATABASE MODELS

a. **Hierarchical Data Model**: Data are organized in tree structure (parent-child structure) in different hierarchy (level)
b. **Network Database Model**: In a network data model, record is stored with a link to other records.
c. **Relational Database Model**: Data are organized in more than one table related to one another.
d. **Object Oriented Database Model**: It assumes data as an object.
e. **Object Relational Database Model**: Objects oriented with relational database model.

Generation of Database Model

First Generation: Based on hierarchical and network database model

Second Generation: Based on relational database model

Third Generation: Based on object-oriented and object-relational database model

A Database Schema

A database schema of a database system is its structure described in a formal language supported by the database management system (DBMS) and refers to the organization of data as a blueprint of how a database is constructed (divided into database tables in case of Relational Databases)

a. **Physical Schema** describes the database design at the physical level. It describes how data are actually stored in database.

b. **Logical Schema** describes the database design at the logical level. It describes what data are stored in database and what relationship exists among them.

Instance: collection of information stored in database at a particular time is called instances of the database.

Constraints:

Sometimes we have set some rules for the database, which have to be followed, known as constraints. For example in a university database application, following rules may have to be followed.

 Domain Constraints: Here data associated with domains typically include integers, floating point numbers, date, time etc. Restrictions on the set of values of the attribute can take can be specified.

Course number should be an integer between 0 to 999.
Maximum Number of characters in cname should be limited to 30.

 Key Constraints: A relation is defined to be set of tuples, since a set does not contain duplicates, no two tuples can be identical.

 Entity Integrity Constraints: Are the set of rules called by the application and these are applicable on all instances of the relations. For example in a university database application, following rules may have to be followed.

Students should not be allowed to take more than two courses at a time.
Students should complete all their assignments before taking the final exam.

 Referential Integrity Constraint: This is a special type of integrity constraint that relates two relations and maintains consistency across the relations. To make this clear, consider two relations R and S. Let PK be the Primary Key of R. Set of attributes FK of S is said to be a Foreign Key of S.

The attributes in FK have the same domain as the attributes of PK
A non null value of FK in a tuple of S also occurs as a value for PK in some tuple of R.

DATA INTEGRITY

1. Entity integrity: Entity integrity ensures that each row can be uniquely identified by an attribute called the Primary key. The Primary key cannot have a NULL value.

2. Domain integrity: Domain integrity refers to the range of valid entries for a given column. It ensures that there are only valid entries in the column.

3. Referential integrity: Referential integrity ensures that for every value of a Foreign key, there is a matching value of the Primary key.

3 Level Architecture of Database

The **external level** of a database has the users' views of the database.

The **conceptual level** describes the logical structure of the entire database, including descriptions of the data and relationships among the data.

The **internal level** gives the details of the physical storage of the database on the computer.

Flat file: It is a static document, spreadsheet or textual file that typically contains data that is not structurally related.

Database Design

1. **Conceptual database design:** This process requires a model of all of the information in a system to be produced
2. **Logical database design:** This process involves the design of a database, but does not take into consideration which database system will be used, or which hardware platform the database will be run on
3. **Physical database design:** This stage involves the design of the database which will be specific to the software and hardware to be used.

Steps in database design

Step 1: Define the Purpose of the Database (Requirement Analysis) Gather the requirements and define the objective of your database.

Step 2: Gather Data, Organize in tables and Specify the Primary Keys.

Step 3: Create Relationships among Tables.

Step 4: Refine & Normalize the Design.

Hierarchy of data

1. Database
2. Table
3. Record
4. Field
5. Character (byte)
6. Bit

Database Management/ Processing Tasks:

- Data Capture
- Data classification
- Data storage
- Data arranging
- Data retrieval
- Data maintenance
- Data verification
- Data coding
- Data editing
- Data transcription
- Data transmission

Types of Table

1. Master Table:

a. It has unique record in a table

b. It contains relatively permanent records and rarely changed

c. It is created only once

2. Transaction Table:

a. May have unique record but generally has duplicate records

b. It contains relatively temporary records and changeable

3. Linker/Junction Table:

a. Used to relate two or more tables

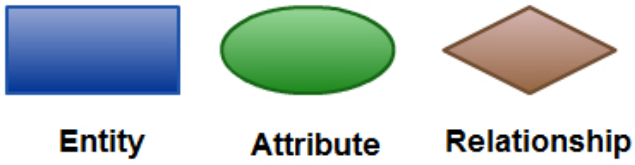
b. It has two common columns (Key fields)

ER Diagram

An entity-relationship diagram (ERD) is a data modeling technique that graphically illustrates an information system's entities and the relationships between those entities. An ERD is a conceptual and representational model of data used to represent the entity framework infrastructure.

The elements of an ERD are:

- Entities
- Relationships
- Attributes



Shortcuts in MS Access

Ctrl + F2: Invoke Builder
Ctrl + "+": Insert new record
F11 or Alt + F1: Bring database window to the front
F12 or Alt + F2: Save As
F6: Navigate between panes

Different Opening options in MS Access:

Open: to open the database for shared access in a multi-user environment so that you and other users can read and write to the database.

Open Read-Only: open the database for read-only access so that you can view but not edit it. Other users can still read and write to the database.

Open Exclusive: When you have a database open with exclusive access, anyone else who tries to open the database receives a "file already in use" message.

Open Exclusive Read-Only: Other users can still open the database, but they are limited to read-only mode.

DATABASE LANGUAGES

DDL

Data Definition Language (DDL) statements are used to define the database structure or schema. Some examples:

- **CREATE** - to create objects in the database
- **ALTER** - alters the structure of the database
- **DROP** - delete objects from the database
- **TRUNCATE** - remove all records from a table, including all spaces allocated for the records are removed
- **COMMENT** - add comments to the data dictionary
- **RENAME** - rename an object

DML

Data Manipulation Language (DML) statements are used for managing data within schema objects. Some examples:

- **SELECT** - retrieve data from the a database
- **INSERT** - insert data into a table
- **UPDATE** - updates existing data within a table
- **DELETE** - deletes all records from a table, the space for the records remain
- **MERGE** - UPSERT operation (insert or update)
- **CALL** - call a PL/SQL or Java subprogram
- **EXPLAIN PLAN** - explain access path to data
- **LOCK TABLE** - control concurrency

DCL

Data Control Language (DCL) statements. Some examples:

- **GRANT** - gives user's access privileges to database
- **REVOKE** - withdraw access privileges given with the GRANT command

TCL

Transaction Control (TCL) statements are used to manage the changes made by DML statements. It allows statements to be grouped together into logical transactions.

- **COMMIT** - save work done
- **SAVEPOINT** - identify a point in a transaction to which you can later roll back
- **ROLLBACK** - restore database to original since the last COMMIT
- **SET TRANSACTION** - Change transaction options like isolation level and what rollback segment to use

Microsoft Access 2007 - 2010 Maximum Capacities

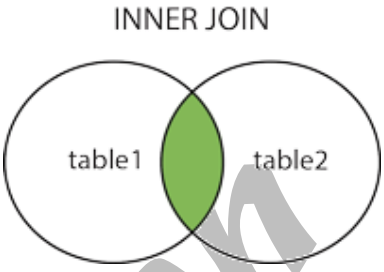
Database specifications	
Attribute	**Maximum**
Database (.mdb) file size	2 GB total for all objects in the database (data, forms, reports, indices, macros, modules, etc.) Because your database application can be linked to tables in other mdb files or to SQL Server, application data storage can be considerably larger.

Number of objects in a database	32,768
Modules (including forms and reports with the HasModule property set to True)	1,000
Number of characters in an object name	64
Number of characters in a password	14 (for MS Access 2003) 20 (for MS Access 2007+)
Number of characters in a user name or group name	20
Number of concurrent users	255
Table specifications	
Attribute	**Maximum**
Number of characters in a table name	64
Number of characters in a field name	64
Number of fields in a table	255
Number of open tables	2048. The actual number will be fewer because of tables opened internally by Microsoft Access.
Table size	2 GB minus the space needed for the system objects.
Number of characters in a Text field	255
Number of characters in a Memo field	65,535 when entering data through the user interface; 1 GB when entering data programmatically.
Size of an OLE Object field	1 GB
Number of indexes in a table	32
Number of fields in an index	10
Number of characters in a validation message	255
Number of characters in a validation rule	2048
Number of characters in a table or field description	255
Number of characters in a record (excluding Memo and OLE Object fields)	4,000; 2,000 for Access 2000 and 2002
Number of characters in a field property setting	255
Query specifications	
Attribute	**Maximum**
Number of enforced relationships	32 per table minus the number of indexes that are on the table for fields or combinations of fields that are not involved in relationships

Number of tables in a query	32
Number of fields in a recordset	255
Recordset size	1 GB
Sort limit	255 characters in one or more fields
Number of levels of nested queries	50
Number of characters in a cell in the query design grid	1,024
Number of characters for a parameter in a parameter query	255
Number of ANDs in a WHERE or HAVING clause	99
Number of characters in a SQL statement	approximately 64,000
Form and report specifications	
Attribute	**Maximum**
Number of characters in a label	2,048
Number of characters in a text box	65,535
Form or report width	22 in. (55.87 cm)
Section height	22 in. (55.87 cm)
Height of all sections plus section headers (in Design view)	200 in. (508 cm)
Number of levels of nested forms or reports	7; 3 f+or Access 2000 and 2002
Number of fields or expressions you can sort or group on in a report	10
Number of headers and footers in a report	1 report header/footer; 1 page header/footer; 10 group headers/footers
Number of printed pages in a report	65,536
Number of controls and sections you can add over the lifetime of the form or report	754
Number of characters in an SQL statement that serves as the **Recordsource** or **Rowsource** property of a form, report, or control (both .mdb and .adp)	32,750
Macro specifications	
Attribute	**Maximum**
Number of actions in a macro	999
Number of characters in a condition	255
Number of characters in a comment	255
Number of characters in an action argument	255

SQL JOIN

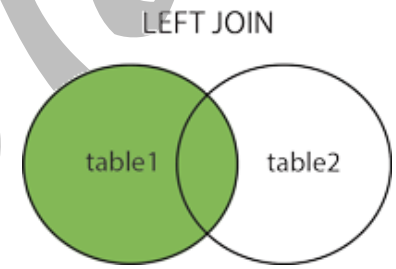
A SQL **join** clause combines records from two or more tables in a relational database. It creates a set that can be saved as a table or used as it is. A JOIN is a means for combining fields from two tables (or more) by using values common to each.



Inner join
This type of join is also known as the Equi join. This join returns all the rows from both tables where there is a match**.**

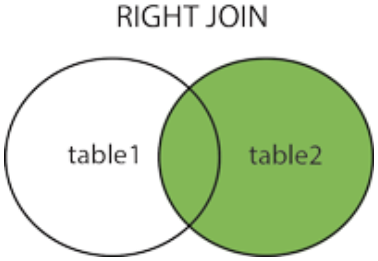
Self Join
Sometime we need to join a table to itself. This type of join is called Self join. In this Join, we need to open two copies of a same table in the memory.

Outer Join
This type of join is needed when we need to select all the rows from the table on the left (or right or both) regardless of whether the other table has common values or not. The Outer join can be of three types: **Left Outer Join, Right Outer Join and Full Outer Join.**

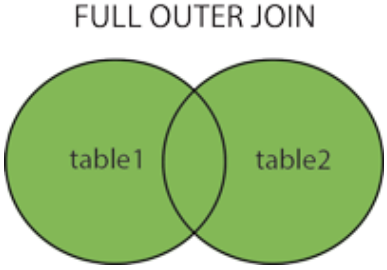


Left Outer Join: If we want to get employee id, employee first name, employees' last name and their department name for all the employees regardless of whether they belong to any department or not, then we can use the left outer join

Right Outer Join: If we want to get all the departments name and employee id, employee first name, and employees last name of all the employees belonging to the department regardless of whether a department have employees or not, then we can use the right outer join



Full Outer Join: If we want to get all the departments name and the employee id, employee first name, employees' last name of all the employees regardless of whether a department have employees or not, or whether a employee belong to a department or not, then we can use the full outer join



Cross Join: This join combines all the rows from the left table with every row from the right table. It returns Cartesian product of rows from tables in the join