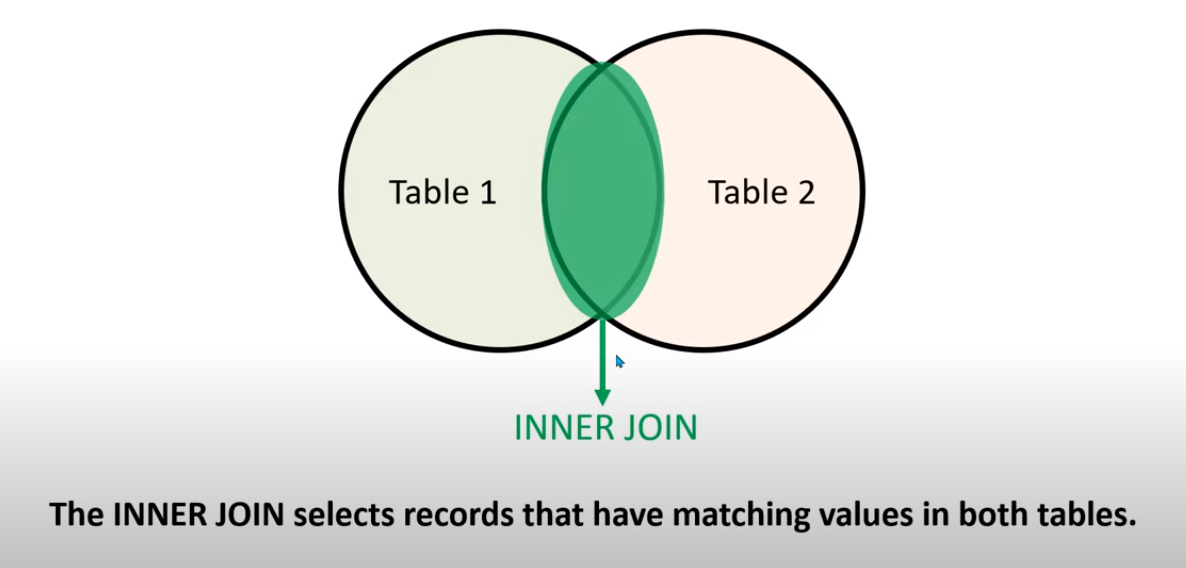
SQL JOIN

The SQL Join help in retrieving data from two or more database tables. The tables are mutually related, using primary keys and foreign keys.

Type of Join

1. **INNER JOIN –** Returns rows when there is a match in both tables.



1.1 EQUI JOIN in SQL

Equi Join in SQL is a type of INNER Join that displays the output by performing a join operation between two or more tables based on the common column between them. It uses the equality ( = ) symbol to compare the data between two columns

1.2 NON-EQUI JOIN in SQL

Non-Equi Join is also a type of INNER Join in which we need to retrieve data from multiple tables. Non-Equi Join matches the column values from different tables based on an inequality based on the operators like <,>,<=,>=,!=, BETWEEN, etc.

1.3 Natural Join

Natural Join in SQL joins two tables based on the same attribute name and datatypes. The resulting table will contain all the attributes of both tables but keep only one copy of each common column.

|  |  |  |  |
| --- | --- | --- | --- |
| **s.no** | **Natural Join** | **s.no** | **Inner Join** |
| 1 | Natural Join joins two tables based on same attribute name and datatypes. | 1 | Inner Join joins two table on the basis of the column which is explicitly specified in the ON clause. |
| 2 | In Natural Join, The resulting table will contain all the attributes of both the tables but keep only one copy of each common column | 2 | In Inner Join, The resulting table will contain all the attribute of both the tables including duplicate columns also |
| 3 | In Natural Join, If there is no condition specifies then it returns the rows based on the common column | 3 | In Inner Join, only those records will return which exists in both the tables |
| 4 | Syntax-  SELECT \*  FROM table1 NATURAL JOIN table2; | 4 | Syntax-  SELECT \*  FROM table1 INNER JOIN table2 ON table1.Column\_Name= table2.Column\_Name; |

1. **Outer – Join** 
   1. LEFT OUTER JOIN – Returns all rows from the left table, and only matched data from the right table.
   2. RIGHT OUTER JOIN – Returns all rows from the right table, and only matched data from the left table
   3. FULL OUTER JOIN – Returns rows when there is a match in one of the table.
2. **SELF JOIN** - is used to join a table to itself as if the table were two tables, temporarily renaming at least one table in the SQL statement.
3. **CROSS JOIN –** Produces a result set which is the number of rows in the first table multiplied by the number of rows in the second table. This kind of result is called as cartesian product.

**DBMS Architecture** :

DBMS architecture describes the structure and how the users are connected to a specific database system.

There are mainly Three Types of database architectures

1. 1-Tier Architecture
2. 2-Tier Architecture
3. 3-Tier Architecture

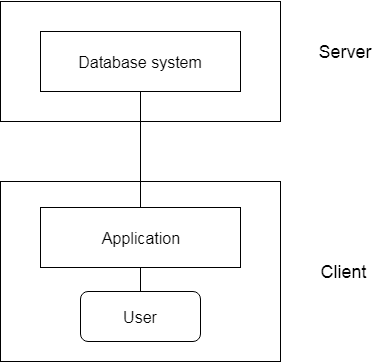
Here Tier means number of layers

1. **1-Tier Architecture :**

User Can Directly use Database.

Eg. Ms-Office

1. **2-Tier Architecture:**

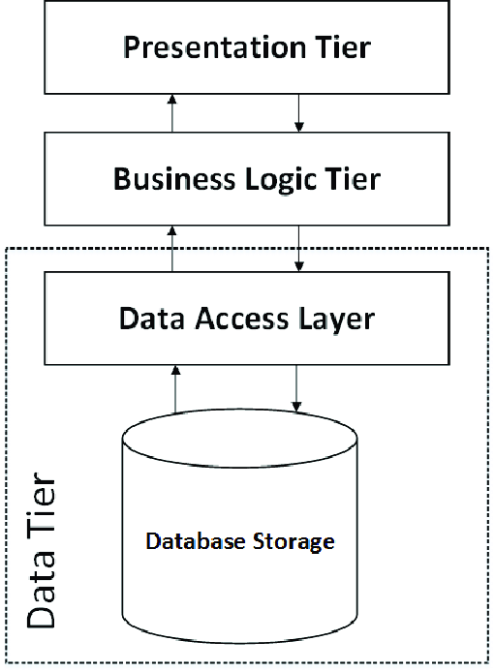


In 2 Tier architecture , User gives input to the API first then API directly access server side Database and gives the output to the client,

Security is less, less maintainance

Eg . Railway counter ticket booking system

1. **3-Tier Architecture :**

****

In 3 tier architectures .. client just gives input like mob number, OTP to API, then application programm is processed by the business layer , the business layer works as intermediate . eg. IRTC ticket booking