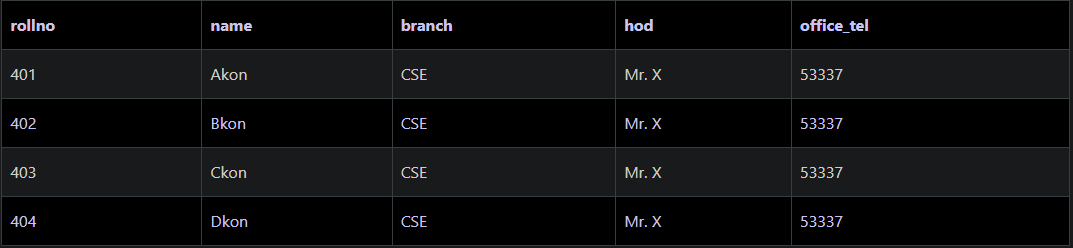
Normalization in DBMS

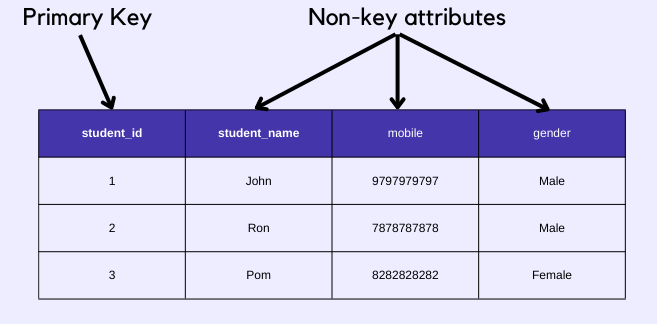
Removing duplicacy or redundancy from the table is called normalization

Normalization helps in keeping data consistent by storing the data in one table and referencing it anywhere else.

Problems without Normalization in DBMS

Insertion, Updation, and Deletion Anomalies are very frequent if the database is not normalized.





Non – Prime/Key attributes are those which are not participating in the formation of primary key.

Types of DBMS Normal forms

Normalization rules are divided into the following normal forms:

1. First Normal Form
2. Second Normal Form
3. Third Normal Form
4. BCNF(BOYCE CODE NORMAL FORM)
5. Fourth Normal Form
6. Fifth Normal Form
7. 1st Normal Form

It should only have single(atomic) valued attributes/columns. Values stored in a column should be of the same domain.

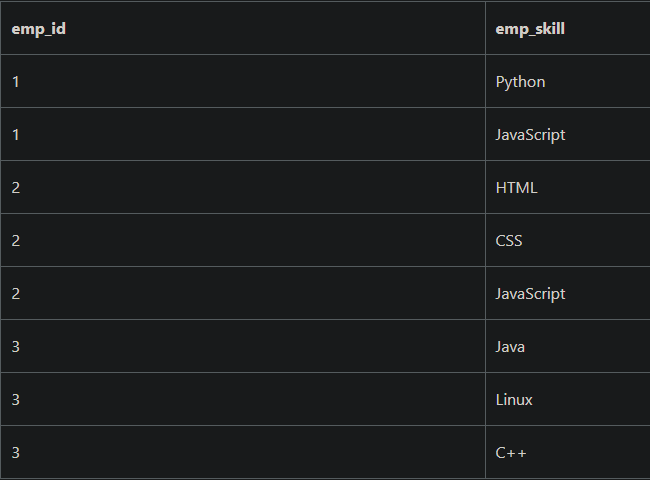


1. Create Separate tables for Employee and Employee Skills

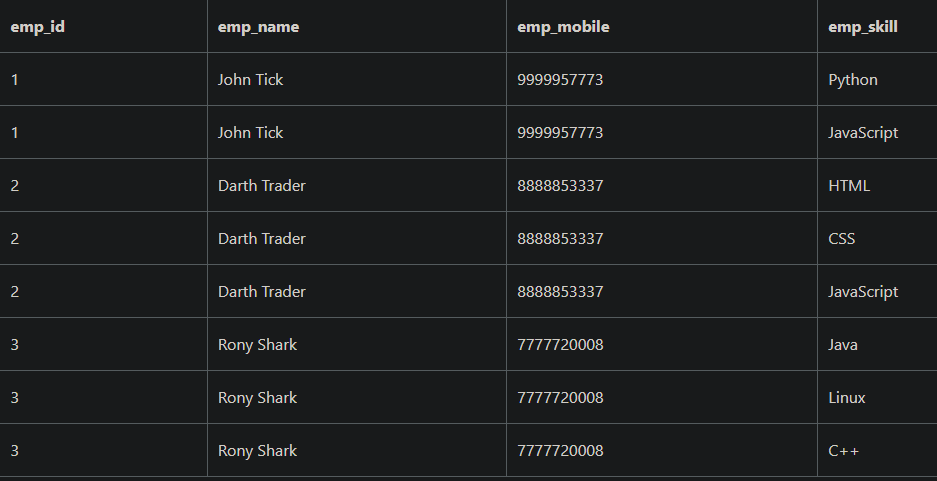
Employee Table



Employee\_Skill table:



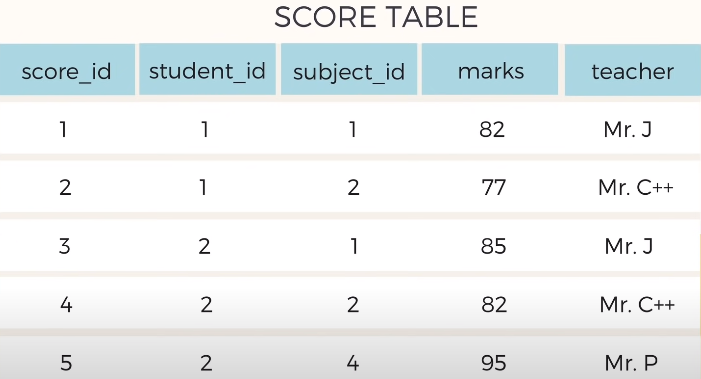
1. Add Multiple rows for Multiple skill



1. Second Normal Form (2NF)

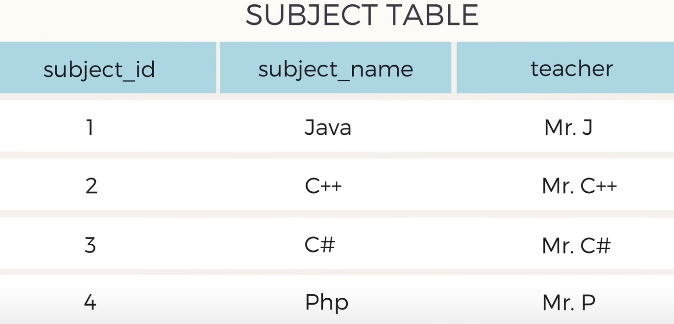
For a table to be in the Second Normal Form,

* It should be in the First Normal form.
* And, it should not have Partial Dependency.



Here teacher is partially dependent ..because it only belongs to subjects ..although the score\_id is primary key her but we can make best primary key by combining student\_id + subject\_id ..

To Remove partial dependency from this table , we should remove teacher column from score table and add it to the subject table



Now the table is in 2NF

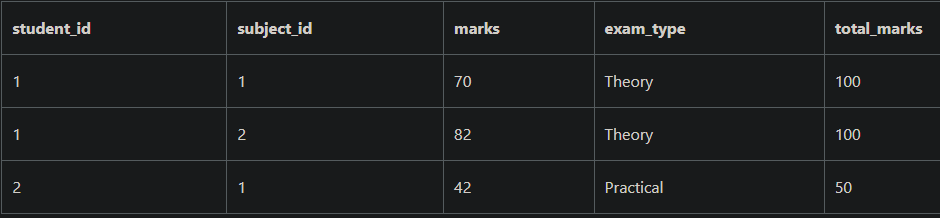
<https://www.youtube.com/watch?v=R7UblSu4744>

1. Third Normal Form (3NF)

Table is said to be in the Third Normal Form when,

* It satisfies the First Normal Form and the Second Normal form.
* And, it doesn't have Transitive Dependency.

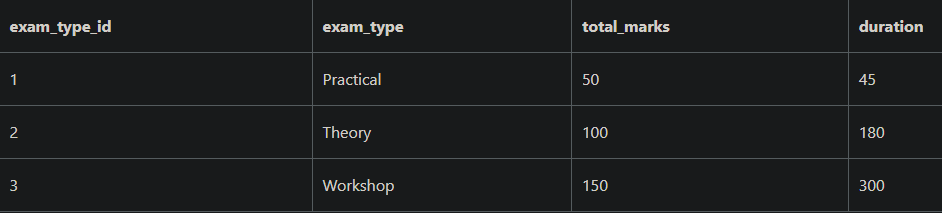
SCORE TABLE



How to Transitive Dependency?

You can create a separate table for ExamType and use it in the Score table.

New Exam\_Type table,



1. **DCL( Data Control Language ) :-** A Data Control Language or DCl , is a computer programming language, which is used to control user access to data stores in a database. This is a component of SQL which is related to the security issues of a database.

**There are two commands in the DCL :**

1 ) GRANT : - used to give permission to a user to access a database.

To allow an user to create table in a database

SYNTAX

GRANT CREATE ANY TABLE TO username ;

2 ) REVOKE : - Used to withdraw permission from a user to access a database.

To prevent an user from creating table in a database

REVOKE CREATE ANY TABLE TO username

Similarly from Droping the table.

**System operations** : - All the database system operations like Creating a database, Creating a table etc

**Object Operation :** - All the database table operation , such as viewing a table , altering a table etc.

1. **TCL( Transaction Control Language ) :-**

TCL is used to manage transaction in a database . This is used to manage changes made to the data in a table by DML statements . It allows statements to be grouped together into logical transaction

There are Three Commands in it

1 ) Commit : - used to save the changes made to the table permanentely

2 ) Rollback : - used to get back to the previous permanent status of the table , kind of similar to UNDO

Note : - Table can be rolled back only if it it temporary . If you committed your changes , it cannot be rolled by.

3 ) Savepoint : - Used along with the ROLLBACK command. It is used to mark a transaction in a table.

It is kind of similar to BOOKMARKS.



Application

1 ) Sorting 🡪 Ascending order , Descending order using order by.

2 ) Filtering 🡪 To fetch some values using operators like in, between , and , or

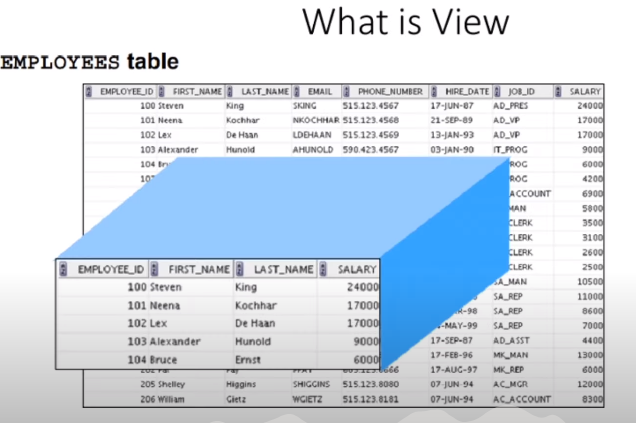
3 ) String Operation -🡪 replace() , like , char\_length() , concat(), lower(), upper(), repeat() , reverse()

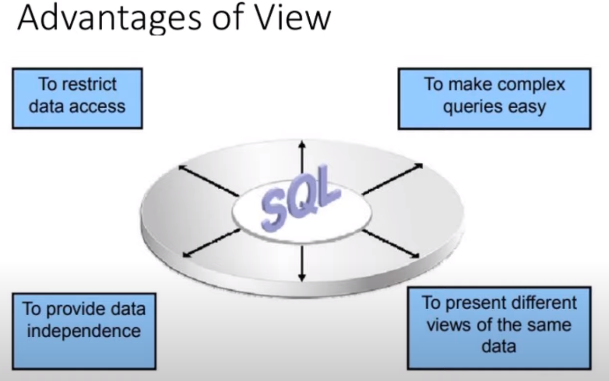
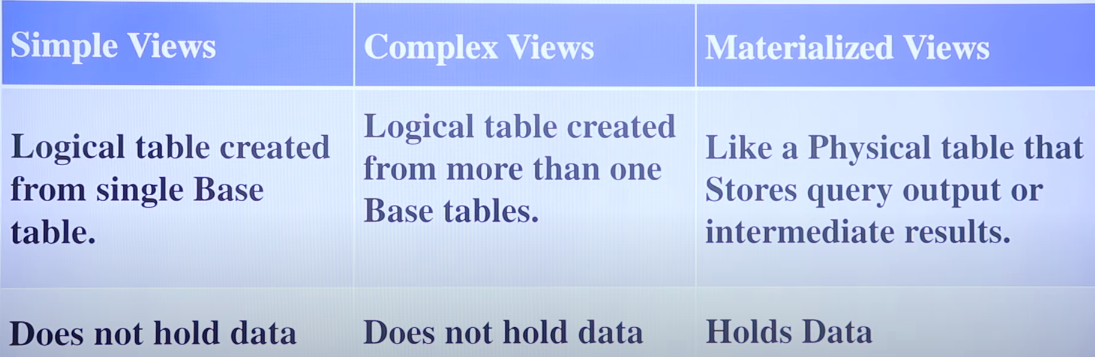
4 ) Math Operation 🡪 pi() , pow(), abs(), ceiling(), floor(), log(), sqrt(), mod()

SQL VIEW

* Virtual Table
* View is the result set of a stored query
* Read – only VS updatable views
* Materialized view

Base table and Views of table





Simple view syntax:

Create view V1 as select \* from Base\_table\_name:

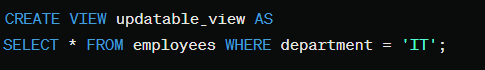
Complex view syntax:

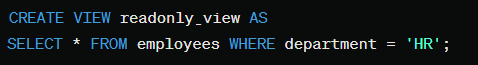
Create view V2 as select Id,Name,city from T1,T2 where T1.id = T2.id;

Materialized view syntax:

Create Materialized view V3 as select dept , max(salary) , min(salary) , average(salary) from T1 group by dept.

**Updatable View and Read only view:**

****

****