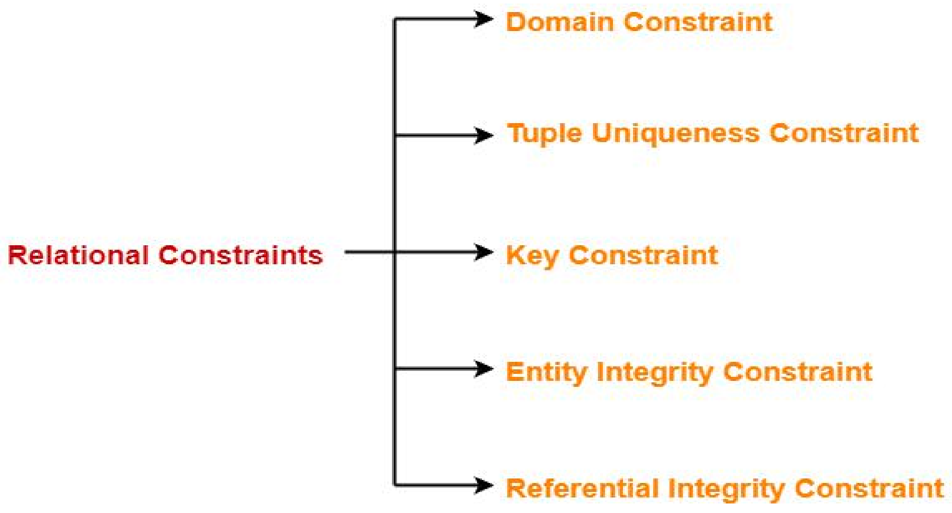
Constraints in DBMS | Types of Constraints in DBMS

Constraints in DBMS-

- Relational constraints are the restrictions imposed on the database contents and operations.
- They ensure the correctness of data in the database.

Types of Constraints in DBMS-

In DBMS, there are following 5 different types of relational constraints-


Relational Constraints
Domain Constraint
Tuple Uniqueness Constraint
Key Constraint
Entity Integrity Constraint
Referential Integrity Constraint

1. Domain Constraint-

- Domain constraint defines the domain or set of values for an attribute.
- It specifies that the value taken by the attribute must be the atomic value from its domain.

Example-

Consider the following Student table-

STU_ID	Name	Age
S001	Akshay	20
S002	Abhishek	21
S003	Shashank	20
S004	Rahul	**A**

Here, value **‘A’** is not allowed since only integer values can be taken by the age attribute.

2. Tuple Uniqueness Constraint-

Tuple Uniqueness constraint specifies that all the tuples must be necessarily unique in any relation.

Example-01:

Consider the following Student table-

STU_ID	Name	Age
S001	Akshay	20
S002	Abhishek	21
S003	Shashank	20
S004	Rahul	20

This relation satisfies the tuple uniqueness constraint since here all the tuples are unique.

Example-02:

Consider the following Student table-

STU_ID	Name	Age
S001	**Akshay**	**20**
S001	**Akshay**	**20**
S003	Shashank	20
S004	Rahul	20

This relation does not satisfy the tuple uniqueness constraint since here all the tuples are not unique.

3. Key Constraint-

Key constraint specifies that in any relation-

- All the values of primary key must be unique.
- The value of primary key must not be null.

Example-

Consider the following Student table-

STU_ID	Name	Age
S001	Akshay	20
S001	Abhishek	21
S003	Shashank	20
S004	Rahul	20

This relation does not satisfy the key constraint as here all the values of primary key are not unique.

4. Entity Integrity Constraint-

- Entity integrity constraint specifies that no attribute of primary key must contain a null value in any relation.
- This is because the presence of null value in the primary key violates the uniqueness property.

Example-

STU_ID	Name	Age
S001	Akshay	20
S002	Abhishek	21
S003	Shashank	20
	Rahul	20

This relation does not satisfy the entity integrity constraint as here the primary key contains a NULL value.

5. Referential Integrity Constraint-

- This constraint is enforced when a foreign key references the primary key of a relation.
- It specifies that all the values taken by the foreign key must either be available in the relation of the primary key or be null.

Important Results-

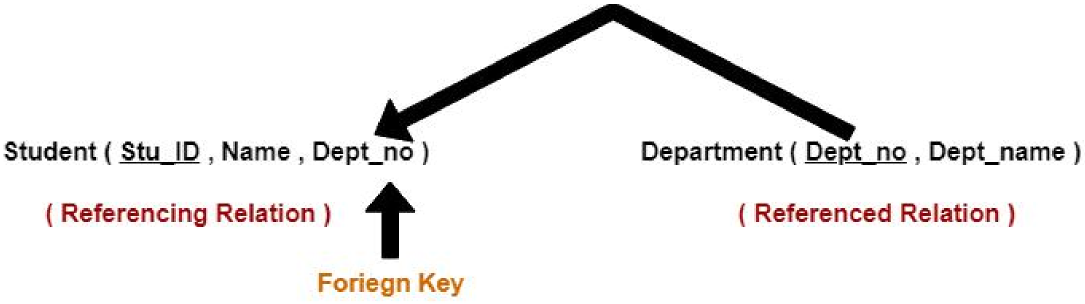
The following two important results emerges out due to referential integrity constraint-

- We can not insert a record into a referencing relation if the corresponding record does not exist in the referenced relation.
- We can not delete or update a record of the referenced relation if the corresponding record exists in the referencing relation.

Example-

Consider the following two relations- ‘Student’ and ‘Department’.

Here, relation ‘Student’ references the relation ‘Department’.


Student (Stu_ID , Name , Dept_no)
(Referencing Relation)
Foriegn Key
Department (Dept_no , Dept_name)
(Referenced Relation)

Student

STU_ID	Name	Dept_no
S001	Akshay	D10
S002	Abhishek	D10
S003	Shashank	D11
S004	Rahul	**D14**

Department

Dept_no	Dept_name
D10	ASET
D11	ALS
D12	ASFL
D13	ASHS

Here,

- The relation ‘Student’ does not satisfy the referential integrity constraint.

- This is because in relation ‘Department’, no value of primary key specifies department no. 14.
- Thus, referential integrity constraint is violated.

Handling Violation of Referential Integrity Constraint-

To ensure the correctness of the database, it is important to handle the violation of referential integrity constraint properly.

Different Types Of Keys in DBMS-

1. Super key
2. Candidate key
3. Primary key
4. Alternate key
5. Foreign key
6. Partial key
7. Composite key
8. Unique key

1. Super Key-

- A super key is a set of attributes that can identify each tuple uniquely in the given relation.
- A super key is not restricted to have any specific number of attributes.
- Thus, a super key may consist of any number of attributes.

Example-

Student (roll , name , sex , age , address , class , section)

Given below are the examples of super keys since each set can uniquely identify each student in the Student table-

- (roll , name , sex , age , address , class , section)
- (class , section , roll)
- (class , section , roll , sex)
- (name , address)

NOTE-

All the attributes in a super key are definitely sufficient to identify each tuple uniquely in the given relation but all of them may not be necessary.

2. Candidate Key-

A minimal super key is called as a candidate key.

OR

A set of minimal attribute(s) that can identify each tuple uniquely in the given relation is called as a candidate key.

Example-

Consider the following Student schema-

Student (roll , name , sex , age , address , class , section)

Given below are the examples of candidate keys since each set consists of minimal attributes required to identify each student uniquely in the Student table-

- (class , section , roll)
- (name , address)

NOTES-

All the attributes in a candidate key are sufficient as well as necessary to identify each tuple uniquely.

- Removing any attribute from the candidate key fails in identifying each tuple uniquely.
- The value of candidate key must always be unique.
- The value of candidate key can never be NULL.
- It is possible to have multiple candidate keys in a relation.
- Those attributes which appears in some candidate key are called as **prime attributes**.

3. Primary Key-

A primary key is a candidate key that the database designer selects while designing the database.

OR

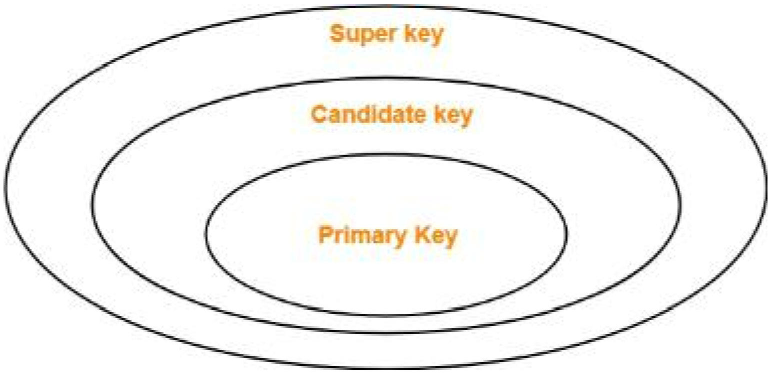
Candidate key that the database designer implements is called as a primary key.

NOTES-

The value of primary key can never be NULL.

- The value of primary key must always be unique.
- The values of primary key can never be changed i.e. no updation is possible.
- The value of primary key must be assigned when inserting a record.
- A relation is allowed to have only one primary key.

Remember-



4. Alternate Key-

Candidate keys that are left unimplemented or unused after implementing the primary key are called as alternate keys.

OR

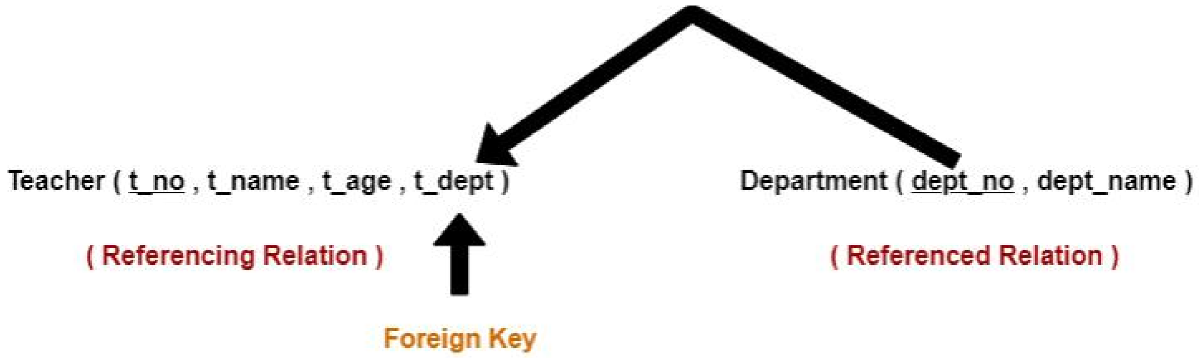
Unimplemented candidate keys are called as alternate keys.

5. Foreign Key-

- An attribute ‘X’ is called as a foreign key to some other attribute ‘Y’ when its values are dependent on the values of attribute ‘Y’.
- The attribute ‘X’ can assume only those values which are assumed by the attribute ‘Y’.
- Here, the relation in which attribute ‘Y’ is present is called as the **referenced relation**.
- The relation in which attribute ‘X’ is present is called as the **referencing relation**.
- The attribute ‘Y’ might be present in the same table or in some other table.

Example-

Consider the following two schemas-


Teacher (t_no , t_name , t_age , t_dept)
Department (dept_no , dept_name)
(Referencing Relation)
(Referenced Relation)
Foreign Key

Here, t_dept can take only those values which are present in dept_no in Department table since only those departments actually exist.

NOTES-

- Foreign key references the primary key of the table.
- Foreign key can take only those values which are present in the primary key of the referenced relation.
- Foreign key may have a name other than that of a primary key.
- Foreign key can take the NULL value.
- There is no restriction on a foreign key to be unique.
- In fact, foreign key is not unique most of the time.
- Referenced relation may also be called as the master table or primary table.
- Referencing relation may also be called as the foreign table.

6. Partial Key-

- Partial key is a key using which all the records of the table can not be identified uniquely.
- However, a bunch of related tuples can be selected from the table using the partial key.

Example-

Consider the following schema-

Department (Emp_no , Dependent_name , Relation)

Emp_no	Dependent_name	Relation
E1	Suman	Mother
E1	Ajay	Father

E2	Vijay	Father
E2	Ankush	Son

Here, using partial key Emp_no, we can not identify a tuple uniquely but we can select a bunch of tuples from the table.

7. Composite Key-

A primary key comprising of multiple attributes and not just a single attribute is called as a composite key.

8. Unique Key-

Unique key is a key with the following properties-

- It is unique for all the records of the table.
- Once assigned, its value can not be changed i.e. it is non-updatable.
- It may have a NULL value.

Example-

The best example of unique key is **Adhaar Card Numbers**.

- The Adhaar Card Number is unique for all the citizens (tuples) of India (table).
- If it gets lost and another duplicate copy is issued, then the duplicate copy always has the same number as before.
- Thus, it is non-updatable.
- Few citizens may not have got their Adhaar cards, so for them its value is NULL.