Integrity Constraint

Keys

Super key
An attribute or a combination of attributes that is used to identify the records uniquely is known as Super Key. A table can have many Super Keys.

Example:

EmpSSN	**EmpNum**	**Empname**
1001	AB05	Shown
2001	AB06	Roslyn
3001	AB07	James

In the above-given example, EmpSSN and EmpNum name are superkeys.
The combination of “SSN” and “Name” is a super key of the following entity set customer.
Because:
The value of attributes “SSN” and “Name”, can uniquely identify that particular customer in the customer entity set, which is the pool of all customers.

Candidate key
The candidate key is a set of one or more attributes whose set of values can uniquely identify an entity instance in the entity set.
Any attribute in the candidate key cannot be omitted without destroying the unique property of the Candidate key.
It is a minimal Super Key.

Properties of Candidate key:

- It must contain unique values
- Candidate key may have multiple attributes
- Must not contain null values
- It should contain minimum fields to ensure uniqueness
- Uniquely identify each record in a table

Example:– ID

– Name, Address

ID	**Name**	**Address**
101	Robot	Boston
201	Mark	Newyork

For above table we have only two Candidate Keys (i.e. Irreducible Super Key) used to identify the records from the table uniquely.
ID Key can identify the record uniquely and a similar combination of Name and Address can identify the record uniquely, but neither Name nor Address can be used to identify the records uniquely as it might be possible that we have two employees with similar names or two employees from the same house.

Composite Key

If we use multiple attributes to create a Primary Key then that Primary Key is called Composite Key (also called a Compound Key or Concatenated Key).

Example of Composite Key, if we have used “Name, Address” as a Primary Key then it will be our Composite Key. ›

Alternate Key

Alternate Key can be any of the Candidate Keys except for the Primary Key.

Example of Alternate Key is “Name, Address” as it is the only other Candidate Key which is not a Primary Key.

Primary Key

A PRIMARY KEY constraint is a rule that the values in one column or a combination of columns must uniquely identify each row in a table.
No primary-key value can appear in more than one row in the table.
No column that is part of the primary key can contain a null.
A table can have only one primary key.

Example:

```
CREATE TABLE Persons (
        ID int NOT NULL,
        FirstName varchar(255),
        Age int,
   PRIMARY KEY (ID)
);
```

FOREIGN KEY (REFERENTIAL INTEGRITY) Constraints

FOREIGN KEY constraints are also called "referential integrity" constraints.
Foreign Key constraints designate a column or combination of columns as a foreign key.
A foreign key links back to the primary key (or a unique key) in another table, and this link is the basis of the relationship between tables.

DEPARTMENTS - Parent

DEPARTMENT_ID	**DEPT_NAME**	**MANAGER_ID**	**LOCATION_ID**
90	Executive	100	1700
110	Accounting	205	1700
190	Contracting	-	1700

EMPLOYEE - Child

EMPLOYEE_ID	**FIRST_NAME**	**LAST_NAME**	**DEPARTMENT_ID**
100	Steven	King	90

101	Neena	Kochhar	90
102	Lex	De Haan	90
205	Shelley	Higgins	110
206	William	Gietz	110

In the tables shown, the primary-key of the DEPARTMENTS table, department_id, also appears in the EMPLOYEES table as a foreign-key column.

Integrity constraint

Entity Integrity Constraint
The entity integrity constraint states that primary keys can't be null.
There must be a proper value in the primary key field.
This is because the primary key value is used to identify individual rows in a table. If there were null values for primary keys, it would mean that we could not identify those rows.

On the other hand, there can be null values other than primary key fields. Null value means that one doesn't know the value for that field. Null value is different from zero value or space.

Example:
In the Car Rental database in the Car table each car must have a proper and unique Reg_No. There might be a car whose rate is unknown - maybe the car is broken or it is brand new - i.e. the Rate field has a null value. See the picture below. The entity integrity constraints ensure that a specific row in a table can be identified.

Referential Integrity Constraint
The referential integrity constraint is specified between two tables and it is used to maintain the consistency among rows between the two tables.

The rules are:
1. You can't delete a record from a primary table if matching records exist in a related table. 2. You can't change a primary key value in the primary table if that record has related records.
3. You can't enter a value in the foreign key field of the related table that doesn't exist in the primary key of the primary table.
4. However, you can enter a Null value in the foreign key, specifying that the records are unrelated.

Examples
Rule 1
You can't delete any of the rows in the CarType table that are visible in the picture since all the car types are in use in the Car table.
Rule 2
You can't change any of the model_ids in the CarType table since all the car types are in use in the Car table
Rule 3
The values that you can enter in the model_id field in the Car table must be in the model_id field in the CarType table.
Rule 4
The model_id field in the Car table can have a null value which means that the car type of that car in not known