**PRIMARY KEY –** it’s a key given to only one column per table.

It can not be null and must be unique.

It uniquely identifies each row.

**FOREIGN KEY** - it refers to the PRIMARY KEY in another table.

It creates a relationship between two tables.

referential integrity: It ensures referential integrity means a foreign key value in one table must match an existing primary key value in another table.

It prevents having orphan records(rows that refer to something that doesn’t exist).

**COMPOSITE KEY:** It is a primary key made of two or more columns that together uniquely identify each row in a table.

If a single column is not enough to guarantee uniqueness, we combine multiple columns.

U cannot insert same values in composite key attributes/columns.

It cannot be null and there combination must be unique.

**SURROGATE KEY**:

A surrogate key is an artificial/auto-generated key that is used as the primary key of a table.

It has no business meaning in the real world.

Usually implemented as an auto-increment integer (like 1, 2, 3…) or a UUID.

Its only job -> uniquely identify a row.

UUID -> Universally Unique Identifier

A **128-bit unique value** (looks like a long string).

Doesn’t follow sequence 1,2,3…, but ensures uniqueness across machines.

**Example :**

CREATE TABLE Orders (

Order\_ID CHAR(36) PRIMARY KEY DEFAULT (UUID()), -- random unique key

Product VARCHAR(50)

);

**Output:**

Order\_ID | Product

-------------------------------------|---------

550e8400-e29b-41d4-a716-446655440000 | Laptop

1b4e28ba-2fa1-11d2-883f-0016d3cca427 | Phone