DATABASE DAY-2

Candidate Key

A **Candidate Key** is a **column** (or a set of columns) in a table that can **uniquely identify each row** in that table.

👉 A table can have **multiple candidate keys**, but **only one of them is chosen** as the **Primary Key**

Example Table: Students

name	Age	Address	amount
vishal	22	noida	22000
mohan	25	delhi	20000
ram	28	kanpur	5000000
Rohit	20	Lucknow	2
Mohan	25	Delhi	-20000
Gita	21	Noida	2000000

Possible Candidate Keys:

The following columns can uniquely identify each student:

- Adhar Number
- Email
- Phone_number

All 3 are Candidate Keys.

✅ You will choose one of these (like Adhar Number) as the Primary Key.

The rest are still Candidate Keys, just not selected as the main key.

Candidate Key Rules:

- Must be **unique** for each row
- Must **not contain NULLs**
- There can be **more than one** candidate key per table

Primary Key(Unique + NOT NULL)

Key Features of a Primary Key:

Feature	Description
Uniqueness	Every value in the primary key **must be unique** (no duplicates)
No NULLs	A primary key **cannot contain NULL** values
One per table	Each table can have **only one primary key**
Stability	The value of a primary key **should not change** frequently

Foreign Key

- the primary keys of another table in current table are referred as foreign key in this table
- It is used to create a relationship between two tables.

Key Points

Feature	Description
Reference	A Foreign Key always refers to the **Primary Key of another table**
Links tables	It creates a **connection** between rows in two tables
Maintains Integrity	Ensures that data stays **consistent** across related tables
Can have duplicates	Yes, unlike Primary Key, Foreign Key can have **repeated values**
Can have NULLs	Yes, if the relationship is optional

Trainer table

Id	Trainer	Course
1	Abhinav	AI/ML
2	Ashish	DS
3	Srajan	DBMS
4	Abhisek	Django FSD

→ In the table above “id” is the primary key of trainer table

Students table

Roll_No	Name	Age	BRANCH	Email	TrainerID
202	Ram	20	CSE	ram@gmail.com	2
204	Mohan	18	BCA	mohan@gmail.com	1
206	Gita	21	MCA	gita@gmail.com	1
207	Jay	20	CSE	jay@gmail.com	3
208	Ravi	23	BCA	ravi@gmail.com	1
209	Sita	21	MCA	sita@gamil.com	3
210	Rajat	22	CSE	rajat@gmail.com	3

→ In the table above “Roll_No” is the primary key of Students TABLE and TrainerID is foreign key that refer to primary key of trainer id

```
CREATE TABLE trainer (
    id INT PRIMARY KEY,
    name VARCHAR(50)
);

CREATE TABLE student (
    id INT PRIMARY KEY,
    name VARCHAR(50),
    trainer_id INT  -- This will be a foreign key
);
```

```
ALTER TABLE student
ADD CONSTRAINT fk_trainer
FOREIGN KEY (trainer_id)
REFERENCES trainer(id)
ON UPDATE CASCADE
ON DELETE SET NULL;
```

Example:1. Students Table (Parent)

student_id	name
101	Alice
102	Bob
103	Charlie

Marks Table (Child)

mark_id	student_id	subject	score
1	101	Math	90
2	102	Science	85
3	101	English	95

- Identify the PK and FK in both tables

Example:2 students Table

student_id	name	course_id
1	Rajat	101
2	Meena	102
3	Aman	NULL
4	Priya	103

courses Table

id	title	teacher_id
101	Data Structures	1001
102	DBMS	1002
103	Operating Systems	NULL

teachers Table

id	name
1001	Dr. Verma
1002	Ms. Sharma
1003	Mr. Anand

Identify the PK and FK in both tables

Exercise - 3

customers Table

customer_id	name
1	Rajat
2	Meena
3	Aman

products Table

product_id	product_name	price
201	Mouse	200
202	Keyboard	500
203	Monitor	5000

orders Table

order_id	customer_id	amount
101	1	500
102	2	800
103	1	300
104	3	450

order_items Table

order_id	product_id	quantity
101	201	2
101	202	1
102	203	1
103	202	1

Identify the PK and FK in both tables

Why Use a Foreign Key?

- Helps **connect related data** across tables
- Enforces **referential integrity** (you can't insert a mark for a student who doesn't exist)
- Avoids duplication of data

What a Foreign Key Prevents:

- You **cannot insert** a value in the foreign key column if that value **doesn’t exist** in the parent table
- You **cannot delete** a referenced row in the parent table unless you handle child rows (like with cascading)