Database Fundamentals

Cybersecurity Department

Course Code: ***CBS213***

Lecture 3: Keys in the Relational Model

Halal Abdulrahman Ahmed

Lecture Outlines

- Why keys are important
- Types of keys

Learning Outcomes

By the end of this lecture, students will be able to:

- Explain the **purpose and importance of keys** in relational databases.
- Identify and differentiate between **types of database keys**, including super, candidate, primary, alternate, unique, foreign, composite, surrogate, natural, secondary, and partial keys.
- Describe how keys are used to **enforce data integrity** and **establish relationships** between tables.

Recap - Keys in the Relational Model

Key Type	Description	Example
Primary Key	Uniquely identifies each record	Student_ID in Students table
Foreign Key	Points to a primary key in another table	Course_ID in Enrollments table
Composite Key	Combination of attributes that uniquely identify a record	(Student_ID, Course_ID) in Enrollments
Candidate Key	Any field that could serve as a unique identifier	Email, Student_ID
Alternate Key	Candidate key not chosen as the primary	Email
Super Key	Any combination of attributes that ensures uniqueness	(Student_ID, Name)

What Are Keys?

What Are Keys?

- A **key** is one or more fields used to **uniquely identify each record** in a table and **connect tables together** in a relational database.
- Keys **prevent duplicate records** and **ensure accuracy**.

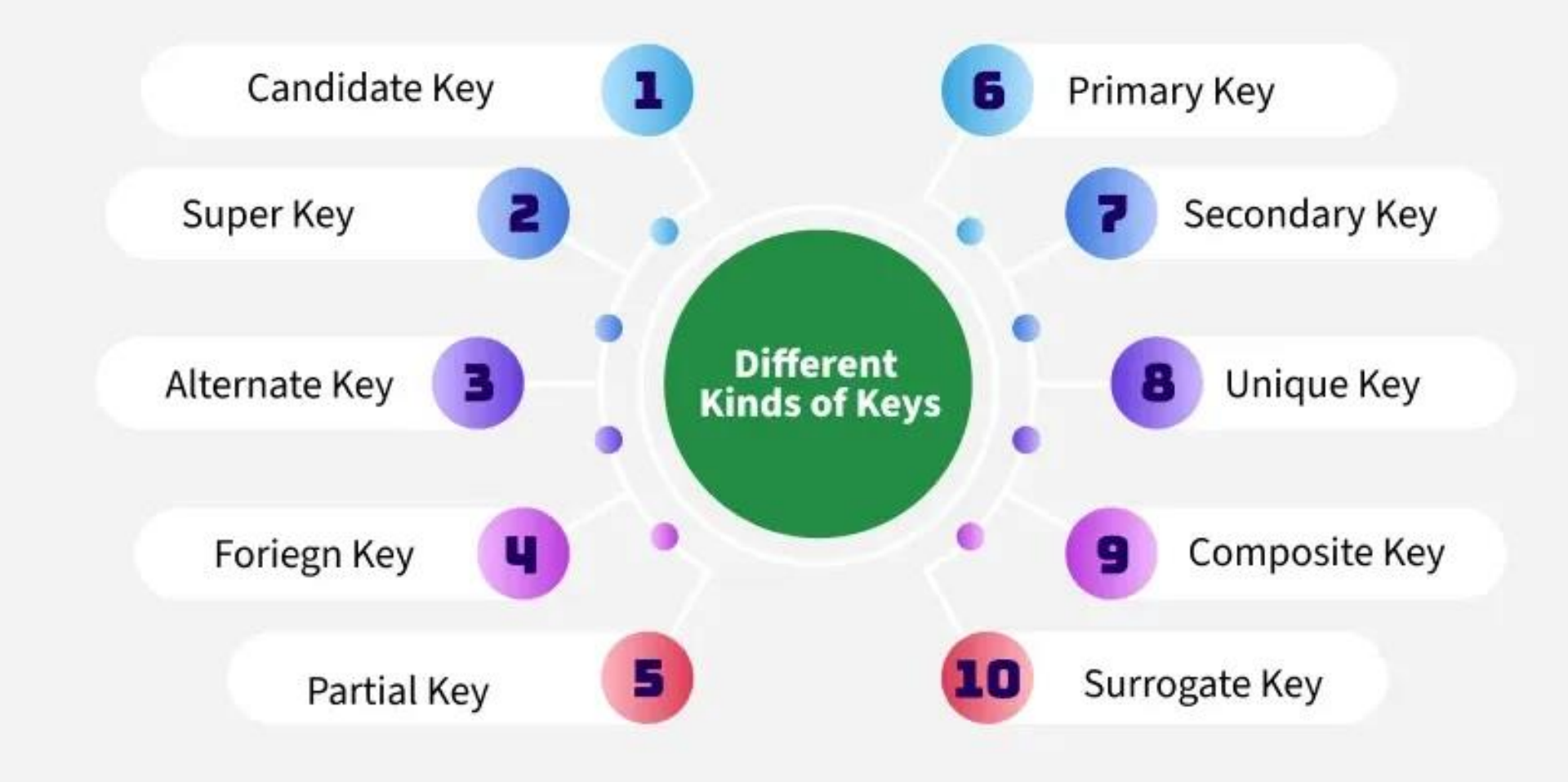
Keys are essential for:

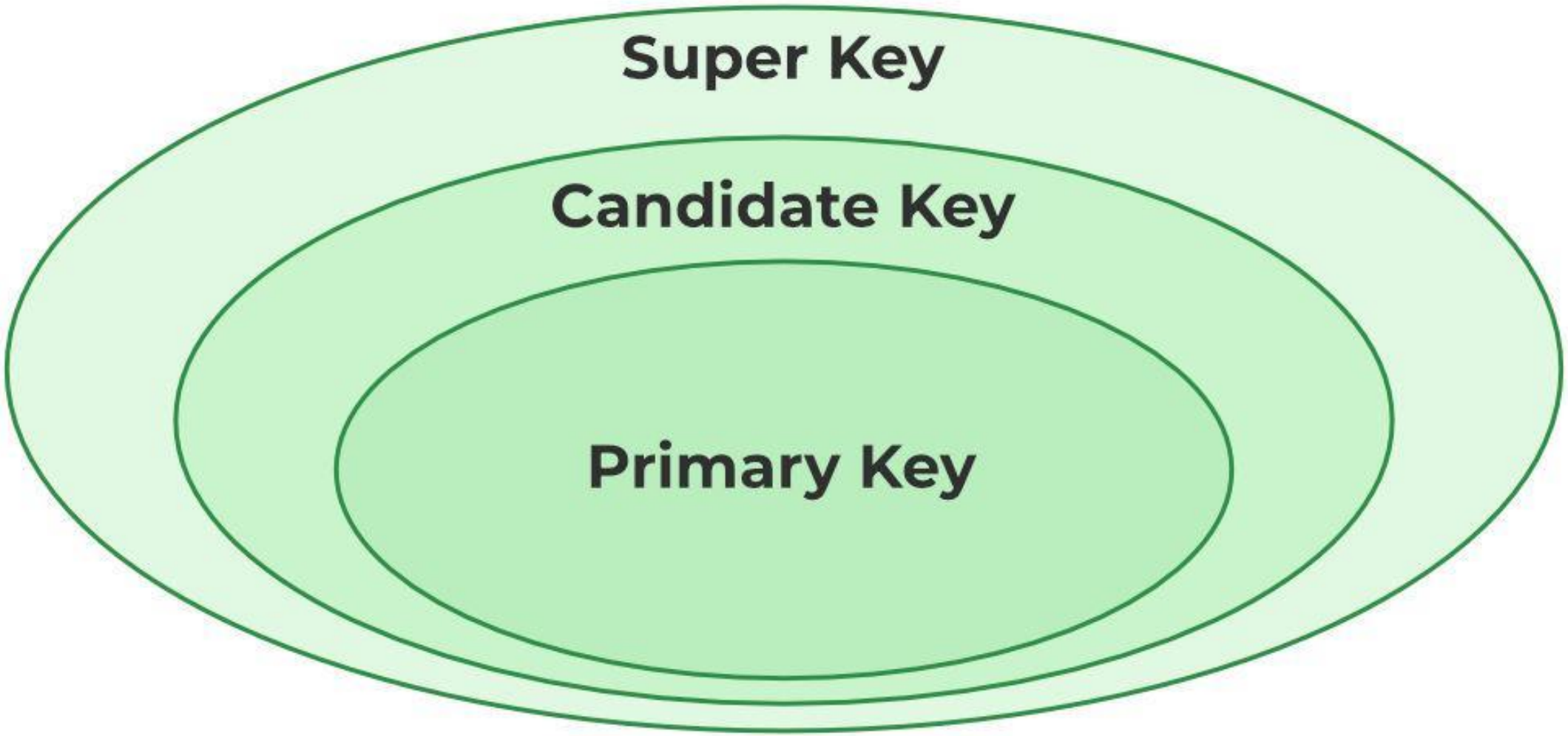
- Searching for a specific row
- Connecting tables
- Preventing duplicate data
- Enforcing rules in tables

Why Keys Are Important

- Every record must be identified uniquely
- Keys help **enforce integrity**
- Keys are used to **create relationships** between tables
- Keys prevent accidental duplicate data
- Keys protect data quality

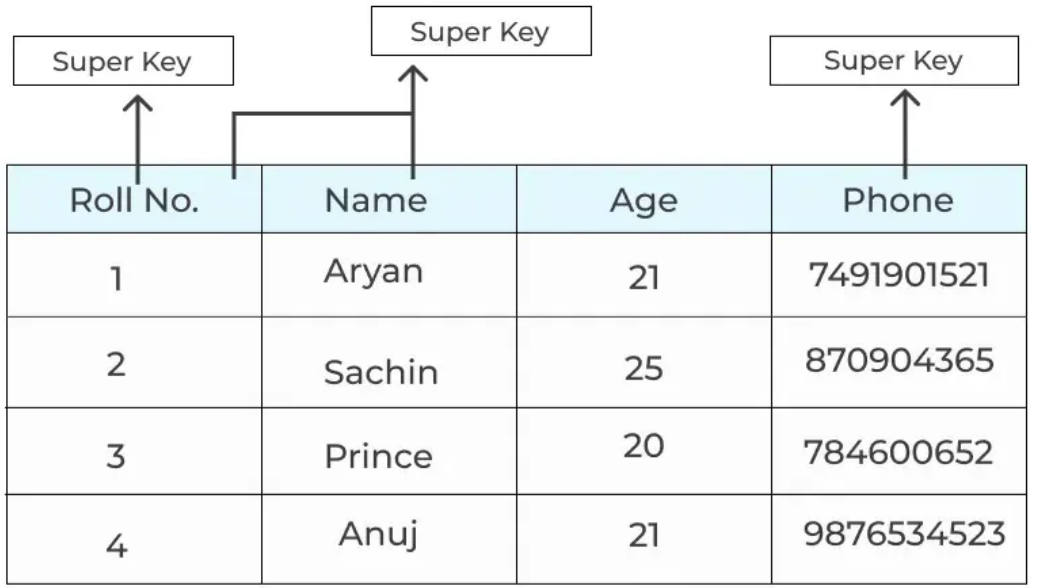
Types of Database Keys




Super Key
Candidate Key
Primary Key

Super Key

- A **Super Key** is **any field or combination of fields** that can **uniquely identify a record** in a table.
- Every **Primary Key is a Super Key**, but **not every Super Key is a Primary Key**.
- A Super Key **may contain extra unnecessary fields**.



Roll No.	Name	Age	Phone
1	Aryan	21	7491901521
2	Sachin	25	870904365
3	Prince	20	784600652
4	Anuj	21	9876534523

Candidate Key

- A **Candidate Key** is a field (or combination of fields) that can **uniquely identify a record** in a table.
- A table can have **more than one** candidate key
- Candidate keys must be **unique**, **minimal** and **not null**

CustomerID	Email	Phone
C01	a@gmail.com	07701234567
C02	b@yahoo.com	07509876543

Candidate Key

Primary Key (PK)

- Selected **one** candidate key
- Used to **uniquely identify each record**
- Must be **unique** and **not null**
- **Only one per table**

Customer ID	Forename	Surname
1	Simon	Jones
2	Emma	Price
3	Laura	Jones
4	Jonathan	Hale
5	Emma	Smith

Properties of a Good Primary Key

A primary key must:

✓Be unique

✓Never change

✓Not allow null values

✓Be simple and short

Bad primary keys:

- Names
- Phone numbers
- Emails

 They may change or repeat

Surrogate Key

- A **surrogate key** is a **system-generated primary key**.
- Has **no business meaning**
- Preferred in most database designs
- Simple and **never changes**

OrderID (Surrogate Key)	TotalAmount
5001	23,000

Natural Key

- A **Natural Key** is a key that comes from **real-world data** and can **uniquely identify a record** in a table.
- It already exists naturally **outside the database** and has a **real meaning**.
- It can be used as a **Candidate Key** and **may be chosen as a Primary Key**, but it is **not always recommended** because it **can change**.

Natural Key vs Surrogate Key

Feature	Natural Key	Surrogate Key
Meaning	Comes from real world (Email, SSN)	System-generated
Stability	Might change	Never changes
Use	Simple systems	Professional DB design

Composite Key

A **Composite Key** uses **two or more columns together** to uniquely identify a record.

OrderID	ProductID	Quantity
5001	101	2
5001	105	1

Composite Primary Key = (OrderID + ProductID)

Alternate Key

- An **Alternate Key** is a **candidate key not selected** as the primary key.

Example:

Candidate keys = {CustomerID, Email}

Selected PK = CustomerID

Alternate Key = Email

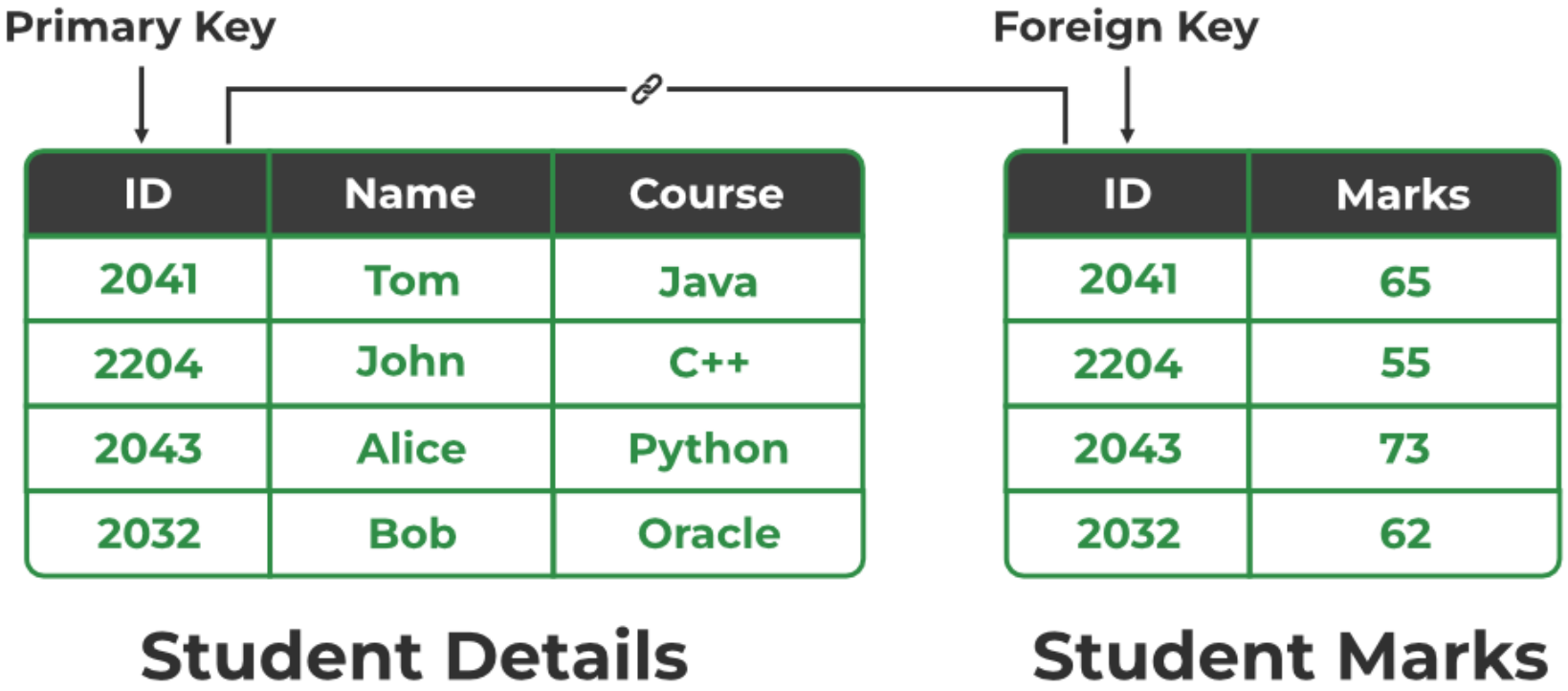
Unique Key

- A **Unique Key** ensures that all values in a column (or group of columns) are **unique** across the table.
- Similar to Primary Key, but **can accept one NULL value**
- It is not the main identifier of the table
- Enforces data integrity

UserID (PK)	Email (Unique)
1	a@gmail.com
2	b@yahoo.com

Foreign Key (FK)

- A **foreign key** connects tables. It is a **primary key from another table**.

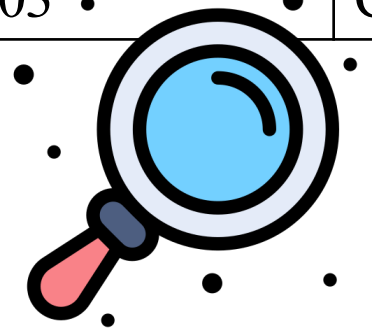


Secondary Key

A **Secondary Key** is used to **search or sort data**, but **cannot uniquely identify** a record.

- Not used to define table relationships
- Not unique
- Helps improve data retrieval

ProductID	ProductName	Category
101	Milk	Dairy
102	Rice	Grains
103	Cheese	Dairy



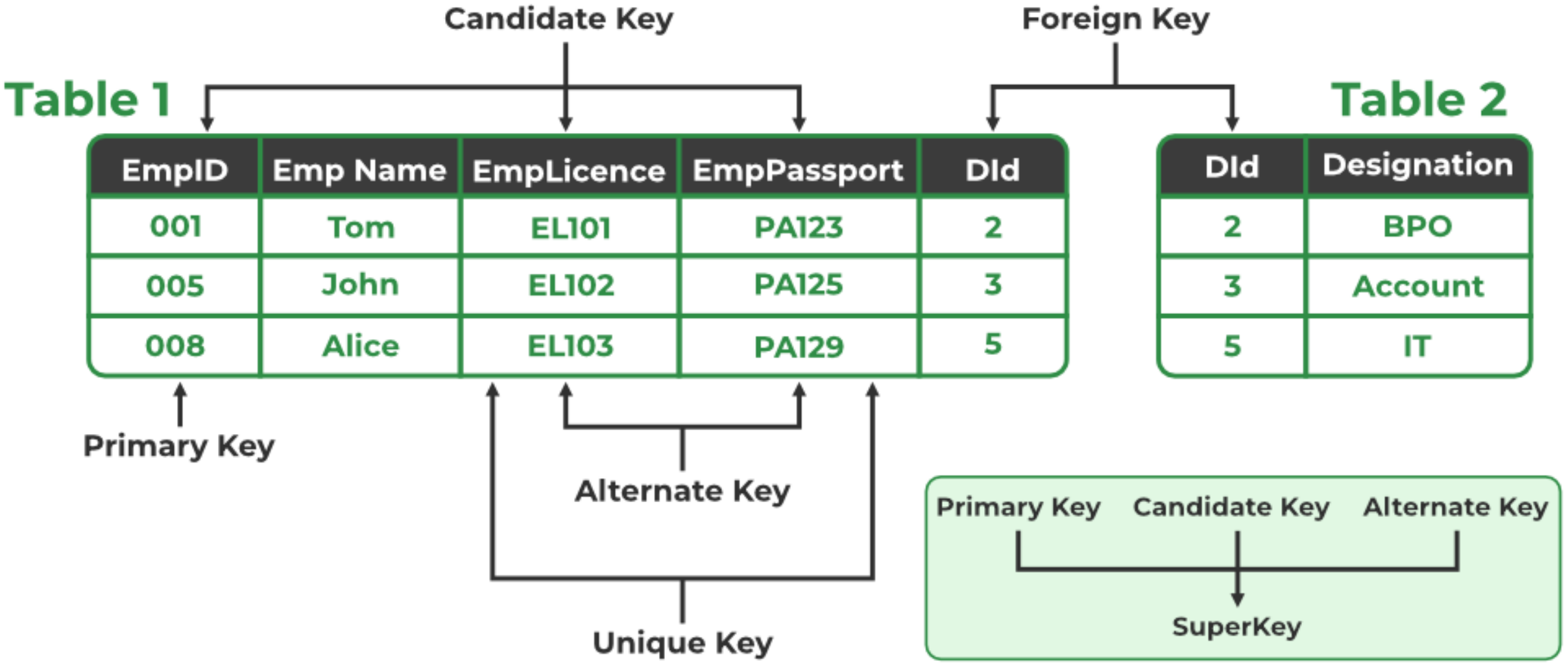
Secondary Keys:
•{ProductName}
•{Category}
These help search like: *"Show all products in Dairy category"*

Partial Key

- A **Partial Key** is a key that can **uniquely identify a record within a weak entity** but **only when linked to another table** (owner entity).
- Used in **weak entity relationships**.

OrderID	ProductID	Quantity
5001	101	2
5001	103	1

- **ProductID is a Partial Key**
- It identifies each item **only when combined with OrderID**


Candidate Key
Foreign Key
Table 1
Table 2
EmpID
Emp Name
EmpLicence
EmpPassport
DId
001
Tom
EL101
PA123
2
005
John
EL102
PA125
3
008
Alice
EL103
PA129
5
DId
Designation
2
BPO
3
Account
5
IT
Primary Key
Alternate Key
Unique Key
Primary Key
Candidate Key
Alternate Key
SuperKey

Key Type	Used in MySQL?	Purpose
Super Key	No	Only for **theoretical understanding**
Candidate Key	No	Used during **design thinking**
Primary Key (PK)	Yes	Enforced in SQL
Unique Key	Yes	Enforced in SQL
Foreign Key (FK)	Yes	Enforced in SQL
Composite Key	Yes	Enforced in SQL
Surrogate Key	Yes (AUTO_INCREMENT)	Enforced in SQL
Secondary Key	Yes (INDEX)	Enforced in SQL
Partial Key	No	ER modeling only

References

- Stallings, W. (2022). *Computer organization and architecture: Designing for performance* (11th ed.). Pearson.
- Elmasri, R., & Navathe, S. B. (2016). Fundamentals of database systems (7th ed.). Pearson.


Any
Question