**Chapter 4: RELATIONAL DATA MODELS**

**Topic – 1: Relational Algebra Operations**

**Basic Operations**

* Selection
* Projection
* Cross product
* Rename

**Joins**

* Natural join
* Outer join

**Set Operators**

* Union
* Intersection
* Set Difference

**Note!**

**🡪 Aggregate functions is also a type of operation.**

**Topic – 2: Constraints**

**Introduction**

* Relational model says that database is collection of **relations**.
* Rows represent real world **entities**.
* Have nothing to do with **physical storage** of data.
* Table is also called ***relation***.

**Terms**

* **Cardinality:** Number of rows.
* **Relation instance:** Predefined tuples in RDBMS with no duplicates.
* **Relation key:** Attribute of a row.
* **Attribute domain:** Integrity constraint of an attribute.

**Attribute Types**

* There is never a **set** of attributes.

**All atomic members = Atomic domain**

* Null is member of each domain.

**RDBMS Constraints**

* Domain constraints
* Key constraints
* Referential integrity constrains

**Domain Constraints**

* Value of each tuple must be unique.
* **Key of the table** identifies a tuple uniquely.
* Key is helpful in finding unique row from the table.
* Foreign key is used to create relationship among tables.

**Topic – 3: Syntax**

**Making An Existing Column NOT NULL**

***ALTER TABLE Employee MODIFY Name VARCHAR(50) NOT NULL;***

**Check Constraint**

***CHECK (value not NULL);***

***CHECK (age>=18)***

***ALTER TABLE Persons ADD CHECK (age>=18);***

***ALTER TABLE Person DROP CHECK age;***

* **CHECK** is replaced by **CONSTRAINT** in SQL server/ Oracle/ MS Access.

**Relational Algebra Operation**

* Procedural query language.
* Relational algebra is theoretical only.

**Relational Algebra Operations From Set Theory**

• DIFFERENCE ( **-** )

• CARTESIAN PRODUCT ( **x** )

**Binary Relational Operations**

• JOIN

• DIVISION

**Facts**

* **pi** eliminates attribute values.

**Cross product = Student x Employee**

**Examples**

***σsubject = "database"(Books)***

***σsubject = "database" and price = "450"(Books)***

***σsubject = "database" and price = "450" or year > "2010"(Books)***

***ρ<New\_Relation\_Name> (Existing\_Relation\_name)***

***select distinct T.name***

***from instructor as T, instructor as S***

***where T.salary > S.salary and S.dept\_name = ‘Comp. Sci.’***

* Keyword **as** can be omitted.

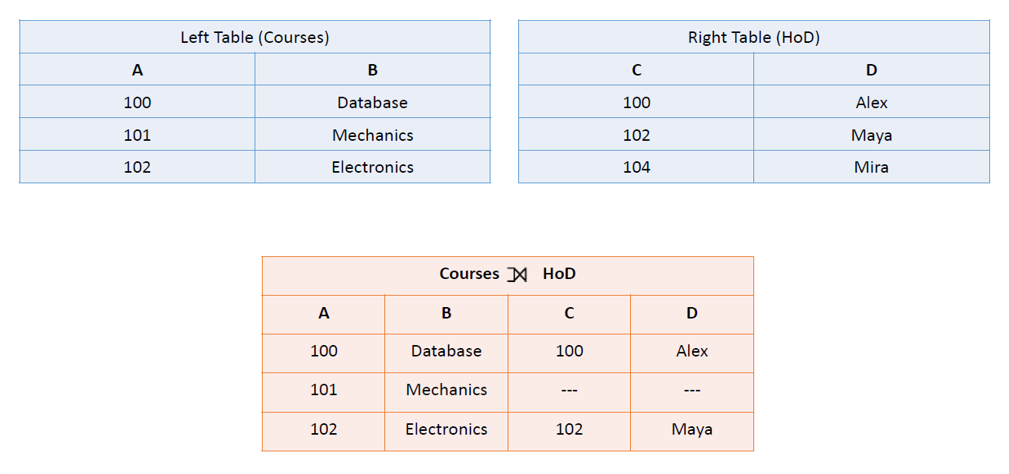
**Topic – 4: Types Of Joins**

**Inner joins**

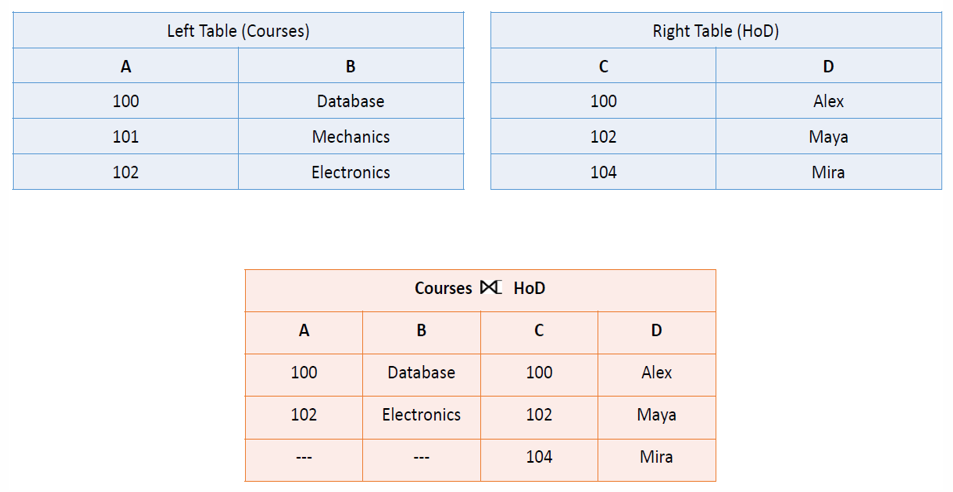
* Theta join
* EQUI join
* Natural join (joining column)

**Outer joins**

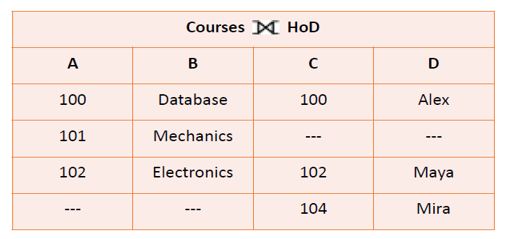
* Left outer join



* Right outer join



* Full outer join



**Topic – 5: Set Operators**

* In union (U), duplicate tuples are removed.
* **Aggregate functions –** avg, min, max, sum, count
* For only **null** values, output returns 0.