1.Use cases of joins in real time applications

- In healthcare:- Utilized to integrate patient data from various sources like electronic health records, laboratory results etc. By joining this it will give a comprehensive view of patients' medical history.
- Social media analytics:- Combining user data with their corresponding comments, likes and shares can provide insight into user behavior, preferences and the popularity of certain content.
- Online advertising
- E-commerce

2.Normalization

- It is the process of organizing data in a database.
- Minimize the redundancy from a relation or a set of relation.

Normal forms:-

❖ 1 NF

Each column should have atomic values.

A column should contain values that are of the same type

Each column should have unique name

Eg:-let us consider a table

student_id	name	subject
1	Abi	c++, python
2	Adithya	python
3	Akhila	c++

But this table is not normalized to normalize this table let us change the table

student_id	name	subject
1	Abi	c++
1	Abi	python

2	Adithya	python
3	Akhila	c++

- 2NF

Table should be in 1 NF.

Table should not have partial dependency

Eg:-

Let us consider 3 table:-

Student

sid	name	branch	regno	address
1	abi	cs	112	k
2	adithya	cs	223	j
3	akhila	cs	334	h

Subject

subject_id	name
1	c++
2	python

Score

score_id	student_id	subject_id	marks	teacher
1	1	1	30	reena
2	3	2	34	jibin

Here the score table student_id+subject_id makes the primary key . but the teacher attribute only depends on subject id so partial dependency arises so in order to remove it we can remove the teacher attribute from the score table and add this in the subject table.

subject_id	name	teacher
1	c++	reena
2	python	jibin

- <u>3 NF</u>

Table should be in 2 NF

There should be no transitive dependency that means a non prime attribute cannot determine any other attribute.

Eg:-

Score

score_id	student_id	subject_id	marks	exam_name	total_mark
1	1	1	20	series	20
2	1	2	30	practical	40

Here total_mark is a non prime attribute which depends only in exam_name which itself is a non prime attribute so in order to make it in 3 NF we can split the table as

Score table

score_id	student_id	subject_id	marks
1	1	1	20
2	1	2	30

Marks table

exam_name	total_mark
series	20
practical	40