**Normalisation** - Organising the data into multiple related tables, to minimize data redundancy.

**Data redundancy** - repetition of multiple data at multiple places, reption of data increases the size of db and other problems like insertion , deletion and updation.

**Example:** A student table of a branch without Normalisation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Roll No | Name | Branch | HOD | Telphone\_No |
| 1 | Amith | Mech | AK | 1111 |
| 2 | Rahul | Mech | AK | 1111 |
| 3 | Vivek | Mech | AK | 1111 |
| 4 | Ramu | Mech | AK | 1111 |
| 5 | Raj | Mech | AK | 1111 |

**Insertion Problem** – To insert the data for every new row we need to add same Branch, HOD, Telephone\_No (Increases the size of db )

**Deletion Problem** – If we delete all student information for any new batch we are deleting branch information along with, since we are storing student and branch info in same table.(Loss of another related dataset when some other dataset is deleted )

**Updation Problem –** If the HOD /Telephone\_No of the branch change we need to update the info in every table.

Normalisation - Student Table = Student Table + Branch table

|  |  |  |
| --- | --- | --- |
| Roll No | Name | Branch |
| 1 | Amith | Mech |
| 2 | Rahul | Mech |
| 3 | Vivek | Mech |

|  |  |  |
| --- | --- | --- |
| Branch | HOD | Telphone\_No |
| Mech | AK | 1111 |

**1NF – First Normal Form**

1.Each Column should contain atomic values(Single values)

2. A column should contain value of same type

3. Each column should have unique name,

|  |  |  |
| --- | --- | --- |
| **~~Name~~** | **~~Name~~** | **DOB** |
| Vivek | Kota | 31-12-1999 |
|  | ~~Ramu, K~~ | ~~R~~ |
| ~~30-06-1992~~ |  | ~~raju~~ |

|  |  |  |
| --- | --- | --- |
| F\_Name | L\_Name | **DOB** |
| Vivek | Kota | 31-12-1999 |
| Ramu | K |  |
| Raju |  | 30-06-1992 |

2NF :Second Normal Form

1. It should be 1NF
2. No partial Dependency(All Non Prime attribute columns depend on prime attribute columns (Canditate Key Columns))

Example : Score Table of student

|  |  |  |  |
| --- | --- | --- | --- |
| Student\_ID | Subject\_ID | Marks | Teacher |
| 1 | 1 | 56 | A |
| 1 | 2 | 59 | B |
| 2 | 1 | 72 | A |
| 2 | 2 | 77 | B |

**Primary Key** : (Student\_ID+ Subject\_ID)

In the above table, Marks dependent on both student\_ID and Subject\_ID but **Teacher Dependent on only Subject\_ID this is partial dependency.**

|  |  |  |
| --- | --- | --- |
| Student\_ID | Subject\_ID | Marks |
| 1 | 1 | 56 |
| 1 | 2 | 59 |
| 2 | 1 | 72 |
| 2 | 2 | 77 |

|  |  |
| --- | --- |
| Subject\_ID | Teacher |
| 1 | A |
| 2 | B |

3NF: Third Normal Form

1. It should be 1NF
2. No Transitive Dependency(When one Non Prime attribute column dependent on another Non Prime attribute column)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Student\_ID | Subject\_ID | Marks | Exam\_Name | Total\_Marks |
| 1 | 1 | 26 | Mid | 30 |
| 1 | 2 | 59 | Final | 70 |
| 2 | 1 | 22 | Mid | 30 |
| 2 | 2 | 67 | Final | 70 |

**Primary\_Key** : (Student\_ID+Subject\_ID)

**Prime Attribute Columns**  : Student\_ID, Subject\_ID

**Non Prime Attribute Columns** : Marks, Exam\_name, Total\_Marks

In the above table, Total\_Marks depend on Exam\_Name which leads to **Transitive Dependency.**

|  |  |  |  |
| --- | --- | --- | --- |
| Student\_ID | Subject\_ID | Marks | Exam\_ID |
| 1 | 1 | 26 | 1 |
| 1 | 2 | 59 | 2 |
| 2 | 1 | 22 | 1 |
| 2 | 2 | 67 | 2 |

|  |  |  |
| --- | --- | --- |
| Exam\_ID | Exam\_Name | Total\_Marks |
| 1 | 1 | 30 |
| 2 | 2 | 60 |

**SQL Commands:**

DDL : Data Definition Language

1. Create
2. Drop
3. Alter - ADD, Drop,Modify
4. Truncate

DQL : Data Query language

1. Select

DML : Data Manipulation Language

1. Insert
2. Delete
3. Update

**My Train DB without Normalisation**

**Train:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TrainNo | TrainName | Source | Destination | Intermediates | NoOfSeatsAvailable | SeatNo |
| 1 | JSB | Vizag | Nellore | A,B | 100 | 1,2,3,4,5,6,etc.. |
|  |  |  |  |  |  |  |

**User:**

|  |  |  |  |
| --- | --- | --- | --- |
| UserId | UserName | Gender | Age |
| 1 | Vivek | Male | 22 |
|  |  |  |  |

**Ticket:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| PNR | **Source** | **Destination** | **TrainNo** | **Name** | **SeatNo** |
| 1 | Vizag | B | 1 | Vivek,Rahul, Ram | 10,9,8 |
|  |  |  |  |  |  |

**Train DB with Normalisation**

**Train:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TrainNo | TrainName | Source | Destination | NoOfSeatsAvailable |
| 1 | JSB | Vizag | Nellore | 100 |
|  |  |  |  |  |

**Station:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| StationNo | ArrrivalTime | DepartureTime | Trip No | TrianNo |
| 1 |  |  | 1 | 1 |
| 2 |  |  | 1 | 1 |
| 3 |  |  | 1 | 1 |
| 4 |  |  | 1 | 1 |

**Station Name:**

|  |  |
| --- | --- |
| StationNo | StationName |
| 1 | Vizag |
| 2 | A |
| 3 | B |
| 4 | Nellore |

**User:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| UserId | UserName | Gender | Age | Email | PhoneNo |
| 1 | Vivek | Male | 22 | abc@xyz.com | 999998888 |
|  |  |  |  |  |  |

**Ticket:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| PNR | **Source** | **Destination** | **TrainNo** | **NoOfPassengers** | **BookedBy(userId)** |
| 1234567 | Vizag | B | 1 | 3 | 1 |
|  |  |  |  |  |  |

**Passenger:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PassengerID | **PassengerName** | **Age** | **Gender** | **SeatNo** |
| 1 | Vivek | 22 | Male | 10 |
| 2 | Rahul | 32 | Male | 9 |
| 3 | Ram | 42 | Male | 8 |

|  |  |
| --- | --- |
| PassengerID | **PNR** |
| 1 | 1234567 |
| 2 | 1234567 |
| 3 | 1234567 |

**BCNF :**

1. It Should be in 3NF
2. For every dependency A-> B, A should be a super Key (which means A can not be a prime attribute with B being Prime Attribute)

Functional Dependency (prime attribute -> Non prime attribute)

Partial Dependency (Part of prime attribute -> Non prime attribute)

Transitive Dependency (Non prime attribute -> Non prime attribute)

Example : For a particular subject more than one teacher may present

|  |  |  |
| --- | --- | --- |
| Student\_ID | Subject | Professor |
| 1 | Java | P.Java |
| 2 | C++ | P.CPP |
| 3 | Java | P.java2 |
| 4 | C# | P.CSharp |
| 5 | Java | P.Java |

Dependencies from above table:

1. Student\_ID+Subject -> Professor
2. Professor -> subject

No Partial dependencies and transitive dependencies but from 2 dependency Professor is non prime attribute subject is an prime attribute depends on Professor

|  |  |
| --- | --- |
| Student\_ID | Prof\_ID |
| 1 | 1 |
| 2 | 2 |
| 3 | 3 |
| 4 | 4 |
| 5 | 1 |

|  |  |  |
| --- | --- | --- |
| Prof\_ID | Professor | subject |
| 1 | P.Java | Java |
| 2 | P.CPP | C++ |
| 3 | P.java2 | Java |
| 4 | P.CSharp | C# |

4CNF:

1. It should be in BCNF
2. No multi-valued dependency. (For a dependency A -> B, if for a single value of A, multiple values of B exists, then the relation will be a multi-valued dependency)

**Example:**

**Student Table:**

|  |  |  |
| --- | --- | --- |
| **Student\_ID** | **Course** | **Hobby** |
| 1 | Science | Cricket |
| 1 | Maths | Hockey |
| 2 | C# | Cricket |
| 2 | Php | Hockey |

Student with ID as 1 opt for two courses, **Science** and **Math** and have two hobbies as Cricket and Hockey. So there is a Multi-valued dependency on Student\_ID and there is no relationship between Course and Hobby

|  |  |
| --- | --- |
| **Student\_ID** | **Course** |
| 1 | Science |
| 1 | Maths |
| 2 | C# |
| 2 | Php |

|  |  |
| --- | --- |
| **Student\_ID** | **Hobby** |
| 1 | Cricket |
| 1 | Hockey |
| 2 | Cricket |
| 2 | Hockey |