Day-1 Notes

ACID – It is a set of properties that ensures database transactions are processed reliably.

Atomicity :- All transactions are executed if any one of transactions get failed in between then all other executed transactions are rolled back

Consistency : Data must flow in a correct form from one state to another --- let’s the negative amount is not valid in parent table due to constraint you have applied so it should also be applicable for all other table as well like in child tables . No false readability

Isolation : Transactions should not interfere with each other like booking movie ticket where only one ticket is available then both the users should not get the message ticket booked

Durability : Once its committed the data is updated and safe - It should retain the values even after power shut down or any sort of failure in an application (got hanged or crashed) even after if the bill is payment is completed the transaction must be saved completed and reflect when power is on so that bill should be provided to the customer.

1NF – Each filed should contain one value only

Student id , student sec student phoneno

1 A 34343,56565

2 B 454545

Student id , student sec student phoneno

1 A 34343

1 A 56565

2 B 454545

Select \* from student where studentid=1

Student id , student sec student phoneno

1 A 34343

1 A 56565

There is high risk belfor 1NF – Updated Anamolies

Anamolies are like

1. Splitting the string or nos and update the value of first phone number and then recombine it , while combining there could be an extra comma, space or mistypes so high chance of data which will be corrupted
2. Difficult to search
3. Redundancy and inconsistency may oocur – if data is repeated there is a confusion which number to update

2NF :

Remove the partial dependency

It applies to tables with a composite primary key

Before 2NF

There is a partial dependency If we have made primary key as student id and coursed as a primary key which is a composite primary key so some of non key column are dependent only a part of a composite primary key like student name depends on student id and course name depends on course id ( which is a partial dependency not on whole)

Student id course id student name course name marks here pk is studentid+coursed

After 2 NF

Student table

Student id student name

1 A

2 B

3 C

Student course table

Student id course id Marks

1. 101
2. 102
3. 103
4. 104

Course table

Course id course name

101 BCA

102 PD

103 BTECH

104 MCA

Every Non Key column depends on the whole pk

3NF - It should be in a 2NF and remove the transitive depencency if any

A transitive dependency means when a non-key column depends on another non key column not directly on primary key.

Before 3NF – Empid is a primary key

Empid Emmname Deptid deptname

Here department name (which is a non key column) depends on another Non-key column) i.e. Department id not on emp id

After 3NF

Empid empname deptid

Deptid deptname

So here all non key columns are directly depend only on the primary key --- there is no indirect (transitive) dependency.

BCNF -- Boy Codd Normal Form

Every determinant is a candidate key

A determinant is a column ( or a combination of columns) on the left side of a functional dependency

A functional dependency if one column can uniquely determine the value of another column(If you know the value of column A, We can easily find out the value of column B)

Before BCNF

A B

Course id Instructor Rooms

101 Nitii Hall1

102 Niti Hall2

103 Jatin Rising Hall

104 Shivani Inspiring Hall

So B is functionally dependent on A or A functionally determines B

Here the course is determinant which is on left side of a functional dependency i.e. instructor as we can determines or find out the instructor.

You can determine the instructor through a determinant i.e. course

Find out the functional dependencies

Course -🡪 Instructor

Instructor -🡪 Room

One thing which is true as Room is a functional dependency which is dependent on Instructor

Here instructor is not a candidate key ( which is not uniquely identifying tows - Because Niti appears in multiple rows That’s why it’s violating BCNF which says determinant should be a candidate key

It should not be tere BCNF if in one table the non candidate keys determine another column more than one like course . ID is a determinant for instructor and instructor is a determinant for rooms where instructor is not a proper candidate key

Course id Instructor

101 Niti

102 Niti

103 Jatin

104 Shivani

Instructor Rooms

Niti Hall 1

Jatin Rising Hall

Shivani Inspiring Hall

Course Id -- determinant which is a candidate key (that uniwquely identifying the rwos)

Instructor --- Determinant which is a candidate Key(that uniquely identifying the rows)

Activity :Normalize the below table Step by Step

OrderId Customer Name CustomerPhone ProductName Quantity Price Total Price

101 Priya 5454454 Laptop 1 40000 40000

101 Priya 4556767 Mouse 2 2500 5000

102 Jatin 45657576 Watch 1 3000 3000

Four Types of predefined System Databases:

Information\_schema

Mysql

Performance\_schema

Sys

Physical location of your databases are stored in

C:\ProgramData\MySQL\MySQL Server 8.0\Data

Inside that folder you have the table data stored in form of files

(.ibd , .sdi) depends upon storage engine

What is InnoDB and MyISAM in MySQL?

MyISAM was the default storage engine for MySQL until MySQL version 5.5

show databases;

use classicmodels;

show tables;

create database taskplanner;

use taskplanner;

show tables;

create table todotask

(

id int primary key,

title varchar(255) not null,

start\_date Date,

due\_date Date

);

create table checklist

(

id int,

task\_id int,

title varchar(255) not null,

is\_completed boolean not null default false,

primary key(id, task\_id),

foreign key (task\_id) references todotask(id) on delete cascade

);

desc todotask;

desc checklist;

drop database taskplanner;

create table states

(

statecode char(2) primary key,

sname varchar(30)

);

use taskplanner;

create table cities

(

state\_co char(2),

cname varchar(30),

foreign key (state\_co) references states(statecode)

);

drop table cities;

drop table states;

insert into states values('D0' , 'Delhi' );

insert into cities values('D0', 'New Delhi');

select \* from states;

select \* from cities;

set foreign\_key\_checks = 1;

delete from cities where state\_co='D0';

create table contacts

(

id int auto\_increment unique key,

name varchar(255) not null,

email varchar(200) not null

);

select \* from contacts;

alter table contacts add officialaddress varchar(50);

alter table contacts modify officialaddress varchar(250);

alter table contacts change officialaddress address varchar(255);

alter table contacts drop address;

alter table contacts modify email varchar(255) default 'abc@gmail.com';

select \* from contacts;

insert into contacts (id , name) values(101, 'Niti');