Que1) Describe the purpose of normalizing data.

Ans1).

Database Normalization is a technique of organizing the data in the database. Normalization is a systematic approach of decomposing tables to eliminate data redundancy(repetition) and undesirable characteristics like Insertion, Update and Deletion Anamolies. It is a multi-step process that puts data into tabular form, removing duplicated data from the relation tables.

* Normalization is used for mainly for the below purposes:
* To free the collection of relations from undesirable insertion, update and deletion dependencies.
* To reduce the need for restructuring the collection of relations as new types of data are introduced, and thus increase the life span of application programs.
* To make the relational model more informative to users.
* To make the collection of relations neutral to the query statistics, where these statistics are liable to change as time goes by.
* Eliminating reduntant(useless) data.
* Ensuring data dependencies make sense i.e data is logically stored.

**PROBLEMS FACED WITHOUT NORMALIZATION.**

* If a table is not properly normalized and have data redundancy then it will not only eat up extra memory space but will also make it difficult to handle and update the database, without facing data loss. Insertion, Updation and Deletion Anamolies are very frequent if database is not normalized. To understand these anomalies let us take an example of a **Student** table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Rollno** | **name** | **branch** | **hod** | **office\_tel** |
| 401 | Akon | CSE | Mary | 53337 |
| 402 | Bkon | CSE | Mary | 53337 |
| 403 | Ckon | CSE | Mary | 53337 |
| 404 | Dkon | CSE | Mary | 53337 |

* In the table above, we have data of 4 Computer Sci. students. As we can see, data for the fields branch, hod(Head of Department) and office\_tel is repeated for the students who are in the same branch in the college, this is **Data Redundancy**.

There are 3 types of anomalies that arise if the data is not normalized:

#### **Insertion Anomaly**

* Suppose for a new admission, until and unless a student opts for a branch, data of the student cannot be inserted, or else we will have to set the branch information as **NULL**.
* Also, if we have to insert data of 100 students of same branch, then the branch information will be repeated for all those 100 students.
* These scenarios are nothing but **Insertion anomalies**.

#### **Updation Anomaly**

* What if Mary leaves the college? or is no longer the HOD of computer science department? In that case all the student records will have to be updated, and if by mistake we miss any record, it will lead to data inconsistency. This is Updation anomaly.

#### **Deletion Anomaly**

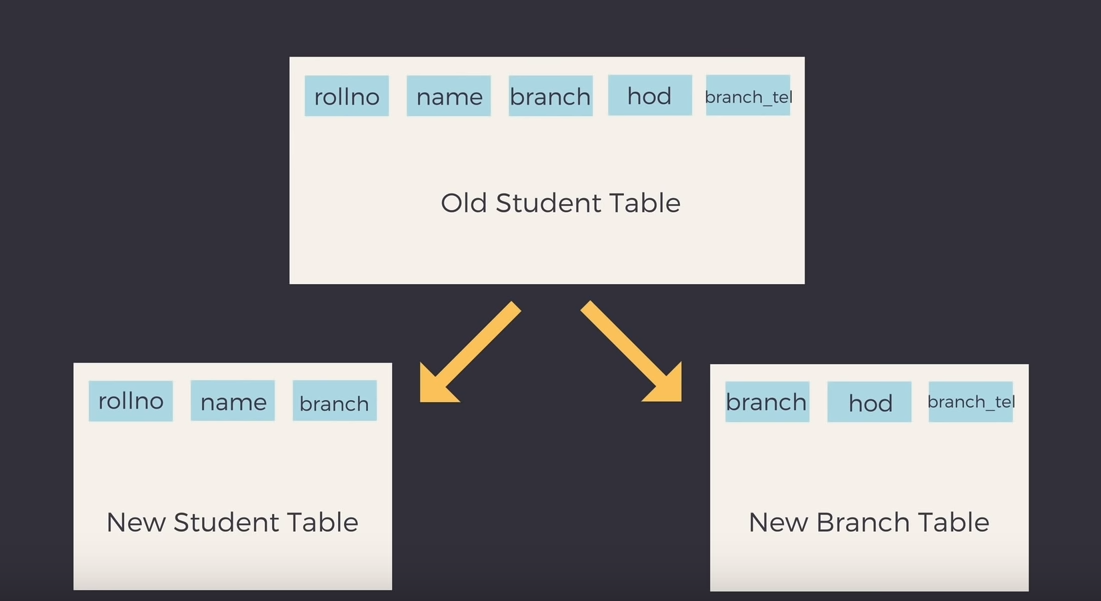
* In our **Student** table, two different informations are kept together, Student information and Branch information. Hence, at the end of the academic year, if student records are deleted, we will also lose the branch information. This is Deletion anomaly.

Que 2). How does normalization eradicate update anomalies from a relation?

Ans 2). As discussed in the forementioned problem above, What if Mary leaves the college? or is no longer the HOD of computer science department? In that case all the student records will have to be updated, and if by mistake we miss any record, it will lead to data inconsistency. This is Updation anomaly.

So now the student table will be split into two different tables.

1. STUDENT TABLE
2. BRANCH TABLE



STUDENT TABLE:

|  |  |  |
| --- | --- | --- |
| **Rollno** | **name** | **branch** |
| 401 | Akon | CSE |
| 402 | Bkon | CSE |
| 403 | Ckon | CSE |
| 404 | Dkon | CSE |

BRANCH TABLE:

|  |  |  |
| --- | --- | --- |
| **branch** | **hod** | **office\_tel** |
| CSE | Michael | 53337 |

Now these two tables will be related by the **branch** and in our case the branch name is CSE.

So like earlier we don’t have to update every row of student table over and over incase we decided to update the HOD and Office telephone number.

But we can just perform update on the BRANCH TABLE just once and replace the former hod with the new HOD and new Office telephone number.

Hence the updation anomaly has been resolved by reducing the redundancy i.e. splitting the information into 2 different tables which are related by BRANCH NAME.