

Ex no:	9- DATABASE NORMALIZATION
Date	19-03-24

Objective:

To normalize the table for a given application.

Description:

Database Normalization is a technique that helps in designing the schema of the database in an optimal manner. The normalisation process is used to reduce the redundancy in a relation or set of relations. Anomalies in insertion, deletion, and update which can be caused by relation redundancy can be avoided by normalization. The process is incremental, and higher degrees of database normalisation cannot be performed until the preceding levels are met.

There are the four types of normal forms:

Normal Form	Description
1NF	A relation is in 1NF if it contains an atomic value.
2NF	A relation will be in 2NF if it is in 1NF and all non-key attributes are fully functional dependent on the primary key.
3NF	A relation will be in 3NF if it is in 2NF and no transition dependency exists.
4NF	A relation will be in 4NF if it is in Boyce Codd normal form and has no multi-valued dependency.
5NF	A relation is in 5NF if it is in 4NF and not contains any join dependency and joining should be lossless.

First Normal Form (1NF)

- A relation will be 1NF if it contains an atomic value.
- It states that an attribute of a table cannot hold multiple values. It must hold only single-valued attribute.
- First normal form disallows the multi-valued attribute, composite attribute, and their combinations.

Instructor's name	Course code
Prof. George	(CS101, CS154)

The table is not in 1NF because of multi-valued attribute Course_code. The decomposition of the table into 1NF is shown below:

Instructor's name	Course code
Prof. George	CS101
Prof. George	CS154

Second Normal Form (2NF)

1. The table should be in the first normal form.
2. The primary key of the table should compose of exactly 1 column.

Student name	Course code
Rahul	CS152
Rajat	CS101

As per the second normal form definition, our enrollment table above isn't in the second normal form. To achieve the same (1NF to 2NF), we can rather break it into 2 tables:

Students:

Student name	Enrolment number
Rahul	1
Rajat	2

The second column is unique and it indicates the enrollment number for the student. Clearly, the enrollment number is unique. Now, we can attach each of these enrollment numbers with course codes.

Courses:

Course code	Enrolment number
CS101	2
CS154	1

Third Normal Form

In the third normal form, the following conditions are required:

- The table should be in the second normal form.
- There should not be any functional dependency.

Course code	Course venue	Instructor's name	Department
MA214	Lecture Hall 18	Prof. Ronald	Mathematics Department
ME112	Auditorium building	Prof. John	Electronics Department

Third normal incorporates functional dependency by decomposing the table into 2 separate tables:

Course code	Course venue	Instructor's ID
MA214	Lecture Hall 18	1
ME112	Auditorium building,	2

Here, the third column is the ID of the professor who's taking the course.

Instructor's ID	Instructor's Name	Department
1	Prof. Ronald	Mathematics Department
2	Prof. John	Electronics Department

Boyce-Codd Normal Form (BCNF)

Boyce-Codd Normal form is a stronger generalization of third normal form. A table is in Boyce-Codd Normal form if and only if at least one of the following conditions are met for each functional dependency $A \rightarrow B$:

- A is a superkey
- It is a trivial functional dependency.

Fourth normal form

A table is said to be in fourth normal form if there is no two or more, independent and multivalued data describing the relevant entity.

Fifth normal form

A table is in fifth Normal Form if:

- It is in fourth normal form.
- It cannot be subdivided into any smaller tables without losing some form of information.

Questions

1. Normalize the following table to 3NF

SID	CID	S_name	C_name	Grade	Faculty	F_phone
1	IS318	Adams	Database	A	Howser	60192
1	IS301	Adams	Program	B	Langley	45869
2	IS318	Jones	Database	A	Howser	60192
3	IS318	Smith	Database	B	Howser	60192
4	IS301	Baker	Program	A	Langley	45869
4	IS318	Baker	Database	B	Howser	60192

```
SQL> CREATE TABLE Norm_1048 (  
2 Sid INT,  
3 Cid VARCHAR(10),  
4 S_name VARCHAR(50),  
5 C_name VARCHAR(50),  
6 Grade CHAR(1),  
7 Faculty VARCHAR(50),  
8 F_phone INT  
9 );
```

Table created.

```
SQL> INSERT INTO Norm_1048 (Sid, Cid, S_name, C_name, Grade, Faculty,  
2 F_phone)  
3 VALUES (1, 'IS318', 'Adams', 'Database', 'A', 'Howser', 60192);  
  
1 row created.  
  
SQL> INSERT INTO Norm_1048 (Sid, Cid, S_name, C_name, Grade, Faculty,  
2 F_phone)  
3 VALUES (1, 'IS301', 'Adams', 'Program', 'B', 'Langley', 45869);  
  
1 row created.  
  
SQL> INSERT INTO Norm_1048 (Sid, Cid, S_name, C_name, Grade, Faculty,  
2 F_phone)  
3 VALUES (2, 'IS318', 'Jones', 'Database', 'A', 'Howser', 60192);  
  
1 row created.  
  
SQL> INSERT INTO Norm_1048 (Sid, Cid, S_name, C_name, Grade, Faculty,  
2 F_phone)  
3 VALUES (3, 'IS318', 'Smith', 'Database', 'B', 'Howser', 60192);  
  
1 row created.  
  
SQL> INSERT INTO Norm_1048 (Sid, Cid, S_name, C_name, Grade, Faculty,  
2 F_phone)  
3 VALUES (4, 'IS301', 'Baker', 'Program', 'A', 'Langley', 45869);  
  
1 row created.  
  
SQL> INSERT INTO Norm_1048 (Sid, Cid, S_name, C_name, Grade, Faculty,  
2 F_phone)  
3 VALUES (4, 'IS318', 'Baker', 'Database', 'B', 'Howser', 60192);  
  
1 row created.  
  
SQL> _
```

```
SQL> select * from norm_1048;
```

SID	CID	S_NAME	C_NAME
1	IS318	Adams	Database
1	IS301	Adams	Program
2	IS318	Jones	Database
3	IS318	Smith	Database
4	IS301	Baker	Program
4	IS318	Baker	Database

```
6 rows selected.
```

Std_1048(T2):

```
SQL> CREATE TABLE Std_1048 (  
2  SID INT PRIMARY KEY,  
3  S_NAME VARCHAR(255)  
4  );
```

Table created.

```
SQL>  
SQL> INSERT INTO Std_1048 (SID, S_NAME) VALUES (1, 'Adams');
```

```
1 row created.
```

```
SQL>  
SQL> INSERT INTO Std_1048 (SID, S_NAME) VALUES(2, 'Jones');
```

```
1 row created.
```

```
SQL> INSERT INTO Std_1048 (SID, S_NAME) VALUES(3, 'Smith');
```

```
1 row created.
```

```
SQL> INSERT INTO Std_1048 (SID, S_NAME) VALUES(4, 'Baker');
```

```
1 row created.
```

```
SQL> select * from std_1048;
```

SID	S_NAME
1	Adams
2	Jones
3	Smith
4	Baker

Course_1048(T3):

```
SQL> CREATE TABLE Course_1048 (  
2  CID VARCHAR(6) PRIMARY KEY,  
3  C_NAME VARCHAR(255)  
4  );
```

Table created.

```
SQL> INSERT INTO Course_1048 (CID, C_NAME) VALUES ('IS318', 'Database');
1 row created.
```

```
SQL> INSERT INTO Course_1048 (CID, C_NAME) VALUES ('IS301', 'Program');
1 row created.
```

```
SQL> select * from Course_1048;
```

CID	C_NAME
IS318	Database
IS301	Program

Common_1023(T4):

```
SQL> CREATE TABLE COM_1048 (
2  SID INT,
3  CID VARCHAR(6),
4  G CHAR(1),
5  FACULTY VARCHAR(255),
6  F_PHONE VARCHAR(10),
7  PRIMARY KEY (SID, CID),
8  FOREIGN KEY (SID) REFERENCES Std_1048(SID),
9  FOREIGN KEY (CID) REFERENCES Course_1048(CID)
10 );
```

Table created.

```
SQL> INSERT INTO COM_1048 (SID, CID, G, FACULTY, F_PHONE) VALUES (1, 'IS318', 'A', 'Howser', '60192');
1 row created.
```

```
SQL> INSERT INTO COM_1048 (SID, CID, G, FACULTY, F_PHONE) VALUES (1, 'IS301', 'B', 'Langley', '45869');
1 row created.
```

```
SQL> INSERT INTO COM_1048 (SID, CID, G, FACULTY, F_PHONE) VALUES (2, 'IS318', 'A', 'Howser', '60192');
1 row created.
```

```
SQL> INSERT INTO COM_1048 (SID, CID, G, FACULTY, F_PHONE) VALUES (3, 'IS318', 'B', 'Howser', '60192');
1 row created.
```

```
SQL> INSERT INTO COM_1048 (SID, CID, G, FACULTY, F_PHONE) VALUES (4, 'IS301', 'A', 'Langley', '45869');
1 row created.
```

```
SQL> INSERT INTO COM_1048 (SID, CID, G, FACULTY, F_PHONE) VALUES (4, 'IS318', 'B', 'Howser', '60192');
1 row created.
```

```
SQL> SELECT * FROM COM_1048;
```

	SID	CID	G	FACULTY	F_PHONE
1	IS318	A	Howser		60192
1	IS301	B	Langley		45869
2	IS318	A	Howser		60192
3	IS318	B	Howser		60192
4	IS301	A	Langley		45869
4	IS318	B	Howser		60192

6 rows selected.

Fac_1048(T5):

```
SQL> CREATE TABLE Facu_1048 (  
2  FACULTY VARCHAR(255) PRIMARY KEY,  
3  F_PHONE VARCHAR(10)  
4  );  
  
Table created.  
  
SQL> INSERT INTO Facu_1048 (FACULTY, F_PHONE) VALUES('Howser', '60192');  
  
1 row created.  
  
SQL> INSERT INTO Facu_1048 (FACULTY, F_PHONE) VALUES('Langley', '45869');  
  
1 row created.  
  
SQL> select * from facu_1048;  
  
FACULTY  
-----  
Howser  
Langley  
  
F_PHONE  
-----  
60192  
45869  
  
SQL> _
```

2. Normalize the following table to BCNF

OID	O Date	CID	C Name	C State	PID	P Desc	P Price	Qty
1006	10/24/09	2	Apex	NC	7	Table	800	1
1006	10/24/09	2	Apex	NC	5	Desk	325	1
1006	10/24/09	2	Apex	NC	4	Chair	200	5
1007	10/25/09	6	Acme	GA	11	Dresser	500	4
1007	10/25/09	6	Acme	GA	4	Chair	200	6

Q2_1048(T1):

```
SQL> CREATE TABLE Q2_1048(  
2  oid INT,  
3  o_date DATE,  
4  cid INT,  
5  c_name VARCHAR(50),  
6  c_state VARCHAR(50),  
7  pid INT,  
8  p_desc VARCHAR(50),  
9  p_price DECIMAL(10, 2),  
10 qty INT  
11 );  
  
Table created.
```

```

SQL> INSERT INTO Q2_1048 (oid, o_date, cid, c_name, c_state, pid, p_desc, p_price, qty)
  2  VALUES (1006, TO_DATE('2009-10-24', 'YYYY-MM-DD'), 2, 'Apex', 'NC', 7, 'Table', 800, 1);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048 (oid, o_date, cid, c_name, c_state, pid, p_desc, p_price, qty)
  2  VALUES (1006, TO_DATE('2009-10-24', 'YYYY-MM-DD'), 2, 'Apex', 'NC', 5, 'Desk', 325, 1);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048 (oid, o_date, cid, c_name, c_state, pid, p_desc, p_price, qty)
  2  VALUES (1006, TO_DATE('2009-10-24', 'YYYY-MM-DD'), 2, 'Apex', 'NC', 4, 'Chair', 200, 5);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048 (oid, o_date, cid, c_name, c_state, pid, p_desc, p_price, qty)
  2  VALUES (1007, TO_DATE('2009-10-25', 'YYYY-MM-DD'), 6, 'Aeme', 'GA', 11, 'Dresser', 500, 4);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048 (oid, o_date, cid, c_name, c_state, pid, p_desc, p_price, qty)
  2  VALUES (1007, TO_DATE('2009-10-25', 'YYYY-MM-DD'), 6, 'Aeme', 'GA', 4, 'Chair', 200, 6);

1 row created.

```

Q2_1048_Customer(T2)

```

SQL> CREATE TABLE Q2_1048_Customer (
  2  cid INT,
  3  c_name VARCHAR(50),
  4  c_state VARCHAR(50),
  5  PRIMARY KEY (cid)
  6  );

Table created.

```

```

SQL> INSERT INTO Q2_1048_Customer (cid, c_name, c_state)
  2  VALUES (2, 'Apex', 'NC');

1 row created.

SQL>
SQL> INSERT INTO Q2_1048_Customer (cid, c_name, c_state)
  2  VALUES (6, 'Aeme', 'GA');

1 row created.

```

Q2_1046_Product(T3)

```

SQL> CREATE TABLE Q2_1046_Product (
  2  pid INT,
  3  p_desc VARCHAR(50),
  4  p_price DECIMAL(10, 2),
  5  PRIMARY KEY (pid)
  6  );

Table created.

```

```

SQL> INSERT INTO Q2_1048_Product (pid, p_desc, p_price)
  2 VALUES (7, 'Table', 800);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048_Product (pid, p_desc, p_price)
  2 VALUES (5, 'Desk', 325);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048_Product (pid, p_desc, p_price)
  2 VALUES (4, 'Chair', 200);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048_Product (pid, p_desc, p_price)
  2 VALUES (11, 'Dresser', 500);

1 row created.

```

Q2_1048_Order(T4)

```

SQL> CREATE TABLE Q2_1048_Order (
  2 oid INT,
  3 o_date DATE,
  4 cid INT,
  5 pid INT,
  6 qty INT,
  7 PRIMARY KEY (oid, o_date, cid, pid),
  8 FOREIGN KEY (cid) REFERENCES Q2_1048_Customer (cid),
  9 FOREIGN KEY (pid) REFERENCES Q2_1048_Product (pid)
10 );

Table created.

```

```

SQL> INSERT INTO Q2_1048_Order (oid, o_date, cid, pid, qty)
  2 VALUES (1006, TO_DATE('2009-10-24', 'YYYY-MM-DD'), 2, 7, 1);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048_Order (oid, o_date, cid, pid, qty)
  2 VALUES (1006, TO_DATE('2009-10-24', 'YYYY-MM-DD'), 2, 5, 1);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048_Order (oid, o_date, cid, pid, qty)
  2 VALUES (1006, TO_DATE('2009-10-24', 'YYYY-MM-DD'), 2, 4, 5);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048_Order (oid, o_date, cid, pid, qty)
  2 VALUES (1007, TO_DATE('2009-10-25', 'YYYY-MM-DD'), 6, 11, 4);

1 row created.

SQL>
SQL> INSERT INTO Q2_1048_Order (oid, o_date, cid, pid, qty)
  2 VALUES (1007, TO_DATE('2009-10-25', 'YYYY-MM-DD'), 6, 4, 6);

1 row created.

```

Result:

To normalize the table for a given application are successfully executed.