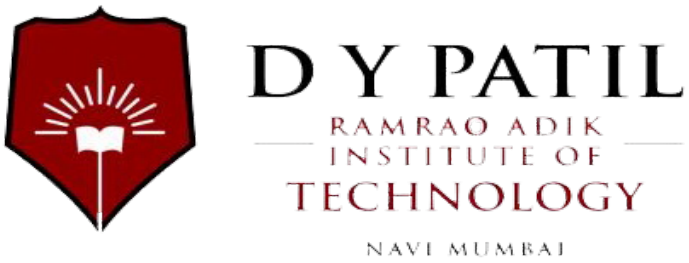

Ramrao Adik Institute of Technology,

Sector 7, Nerul, Navi Mumbai

(Affiliated to University of Mumbai)

Department of Electronics and Telecommunications

MINI PROJECT FOR THE SUBJECT DBMS

Report on

COLLEGE DATABASE MANAGEMENT SYSTEM

Performed by

RAMANAND DHOLE--17ET7006

CERTIFICATE OF APPROVAL

This is to certify that, the Mini Project for the subject DBMS entitled

COLLEGE DATABASE MANAGEMENT SYSTEM

Is bonafide work done by

RAMANAND DHOLE----17ET7006

University of Mumbai

SIGNATURE: GRADE:

TITLE: COLLEGE DATABASE MANAGEMENT SYSTEM

SOFTWARE REQUIRED: MY SQL POSTGRE
THEORY:
A database-management system (DBMS) is a collection of interrelated data and a set of programs to access those data. This is a collection of related data with an implicit meaning and hence is a database. The collection of data, usually referred to as the database, contains information relevant to an enterprise. The primary goal of a DBMS is to provide a way to store and retrieve database information that is both convenient and efficient. By data, we mean known facts that can be recorded and that have implicit meaning. For example, consider the names, telephone numbers, and addresses of the people you know.
Knowledge refers to the practical use of information. While information can be transported, stored or shared without many difficulties the same can not be said about knowledge. Knowledge necessarily involves a personal experience. Referring back to the scientific experiment, a third person reading the results will have information about it, while the person who conducted the experiment personally will have knowledge about it. Database systems are designed to manage large bodies.
Database systems are designed to manage large bodies of information. Management of data involves both defining structures for storage of information and providing mechanisms for the manipulation of information. In addition, the database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access. If data are to be shared among several users, the system must avoid possible anomalous results. Because information is so important in most organizations, computer scientists have developed a large body of concepts and techniques for managing data. These concepts and technique form the focus of this book.
1.Data isolation. Because data are scattered in various files, and files may be in different formats, writing new application programs to retrieve the appropriate data is difficult.
2. Integrity problems. The data values stored in the database must satisfy certain types of consistency constraints. For example, the balance of a bank account may never fall below a prescribed amount (say, $25). Developers enforce these constraints in the system by adding appropriate code in the various application programs. However, when new constraints are added, it is difficult to change the programs to enforce them. The problem is compounded when constraints involve several data items from different files.
3. Atomicity problems. A computer system, like any other mechanical or electrical device, is subject to failure. In many applications, it is crucial that, if a failure occurs, the data be restored to the consistent state that existed prior to the failure.
4. Concurrent-access anomalies. For the sake of overall performance of the system and faster response, many systems allow multiple users to update the data simultaneously. In such an environment, interaction of concurrent updates may result in inconsistent data.
5. Security problems. Not every user of the database system should be able to access all the data. For example, in a banking system, payroll personnel need to see only that part of the database that has information about the various bank employees. They do not need access to information about customer accounts. But, since application programs are added to the system in an ad hoc manner, enforcing such security constraints is difficult. These difficulties, among others, prompted the development of database systems. In what follows, we shall see the concepts and algorithms that enable database systems to solve the problems with file-processing systems. In most of this book, we use a bank enterprise as a running example of a typical data-processing application found in a corporation.

Advantages of DBMS:
Data independence

Efficient data access
Data integrity and security
Data administration
Reduced application development time
Disadvantages of a DBMS:
Danger of a Overkill
Complexity
Qualified Personnel
Costs
Lower Efficiency

Consider the schema for College Database:
STUDENT (USN, SName, Address, Phone, Gender)
SEMSEC (SSID, Sem, Sec)
CLASS (USN, SSID)
SUBJECT (Subcode, Title, Sem, Credits)
IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)
Write SQL queries to
1. List all the student details studying in fourth semester 'C' section.
2. Compute the total number of male and female students in each semester and in each section.
3. Create a view of Test1 marks of student USN '1BI15CS101' in all subjects.
4. Calculate the FinalIA (average of best two test marks) and update the corresponding table for all students.
5. Categorize students based on the following criterion:
If FinalIA = 17 to 20 then CAT = 'Outstanding'
If FinalIA = 12 to 16 then CAT = 'Average'
If FinalIA< 12 then CAT = 'Weak'
Give these details only for 8th semester A, B, and C section students.

ER-Diagram:

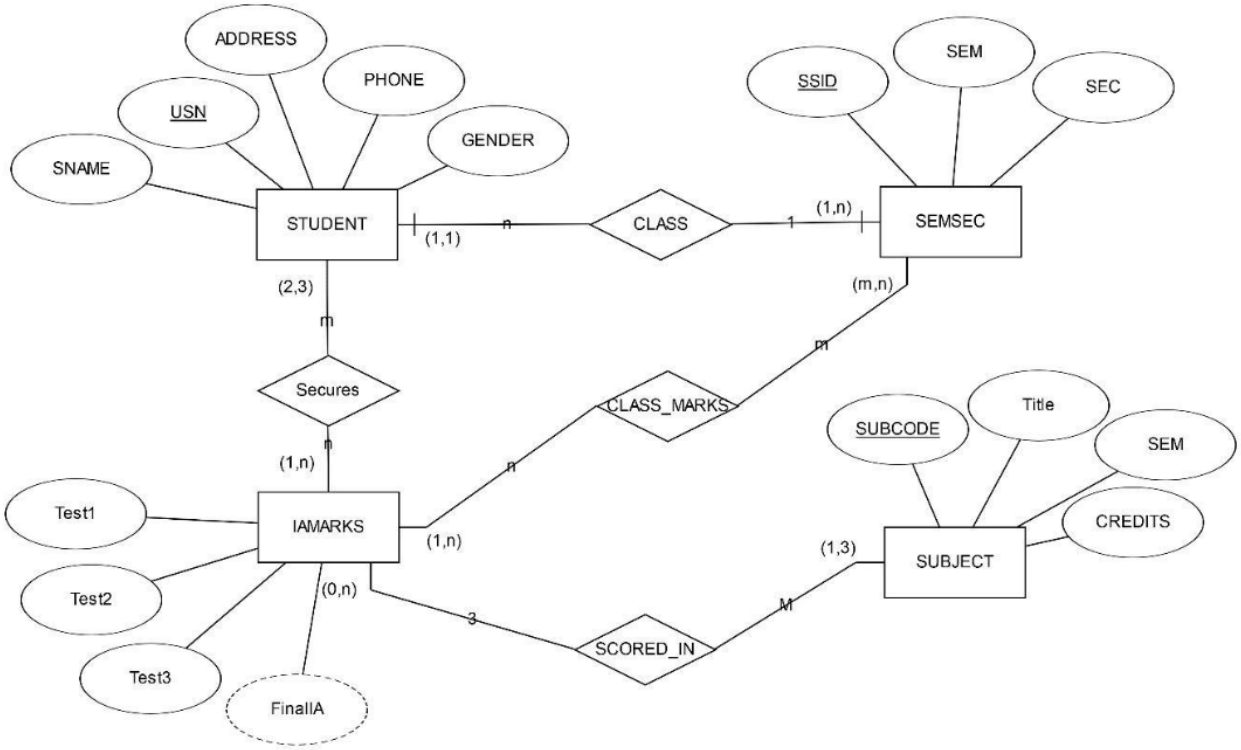

Table Creation:

STUDENT
CREATE TABLE STUDENT
(USN VARCHAR(10) PRIMARY KEY,
SNAME VARCHAR(25),
ADDRESS VARCHAR(25),
PHONE VARCHAR(10),
GENDER CHAR(1));
Table created.

SEMSEC
CREATE TABLE SEMSEC
SSID VARCHAR(5) PRIMARY KEY,
SEM NUMBER(2),
SEC CHAR(1));
Table created.

CLASS
CREATE TABLE CLASS
(USN VARCHAR(10),
SSID VARCHAR(5), PRIMARY
KEY(USN,SSID),
FOREIGN KEY(USN) REFERENCES STUDENT(USN),
FOREIGN KEY(SSID) REFERENCES SEMSEC(SSID));
Table created.

SUBJECT

CREATE TABLE SUBJECT
(SUBCODE VARCHAR(8) PRIMARY KEY,
TITLE VARCHAR(20),
SEM NUMBER(2), CREDITS
NUMBER(2));
Table created.

IAMARKS
CREATE TABLE IAMARKS
(USN VARCHAR(10),
SUBCODE VARCHAR(8),
SSID VARCHAR(5), TEST1
NUMBER(2), TEST2
NUMBER(2),

TEST3 NUMBER(2),
FINALIA NUMBER(3),
PRIMARY KEY(USN,SUBCODE,SSID),
FOREIGN KEY(USN) REFERENCES STUDENT(USN),
FOREIGN KEY(SUBCODE) REFERENCES SUBJECT(SUBCODE),
FOREIGN KEY(SSID) REFERENCES SEMSEC(SSID));
Table created.

Values for tables:

STUDENT:

INSERT INTO STUDENT VALUES
('&USN','&sname','&address','&phone','&gender');

select * from student;

USN	SNAME	ADDRESS	PHONE	G
lcg15cs001	Abhi	tumkur	9875698410	M
lcg15cs002	Amulya	Qubbi	8896557412	F
lcg16me063	Chethan	Nittur	7894759522	m
lcg14ec055	Raghavi	Sspuram	9485675521	F
lcg15ee065	Sanjay	Bangalore	9538444404	M

SEMSEC:
INSERT INTO SEMSEC VALUES ('&SSID', '&sem','&sec');
select * from semsec;

S	S ID	Sems
5A	5	A
3B	3	B
7A	7	A

2C	2	C
4B	4	B
4c	4	c

CLASS:
INSERT INTO CLASS VALUES (‘&USN’,’&SSID’);
select * from class;

USN	SSID
lcg15cs001	5A
lcg15cs002	5A
lcg16me063	3B
lcg14ec055	7A
lcg15ee065	3B
lcg15ee065	4c
lcg15cs002	4c

SUBJECT:
INSERT INTO SUBJECT VALUES ('10CS81','ACA', 8, 4);
select * from subject;

SUBCODE	TITLE	SEM	CREDITS
15cs53	Dbms	5	4
15cs33	ds	3	4
15cs34	Co	3	4
15cs158	Dba	5	2
10cs71	Oomd	7	4

IAMARKS:
INSERT INTO IAMARKS VALUES
(‘&USN’,’&SUBCODE’,’&SSID’,’&TEST1’,’&TEST2’,’&TEST3’);
select * from iamarks;

USN	SUBCODE	SSID	TEST1	TEST2	TEST3	FINALIA
lcg15cs001	15cs53	5A	18	19	15	19
lcg15cs002	15cs53	5A	15	16	14	16
lcg16me063	15cs33	3B	10	15	16	16
lcg14ec055	10cs71	7A	18	20	21	21

lcg15ee065	15cs33	3B	16	20	17	19
lcg15ee065	15cs53	4c	19	20	18	20

Queries:

1. List all the student details studying in fourth semester 'C' section.

```
select
s.usn,sname,address,phone,gender from
student s, class c, semsec ss where
sem=4 and
sec='c' and
ss.ssid=c.ssid and
c.usn=s.usn;
```

USN	SNAME	ADDRESS	PHONE	G
lcg15ee065	Sanjay	Bangalore	9538444404	M
lcg15cs002	Amulya	Gubbi	8896557412	F

2. Compute the total number of male and female students in each semester and in each section.

```
SELECT SEM,SEC,GENDER,COUNT(*)
FROM STUDENT S, SEMSEC SS,CLASS C
WHERE S.USN=C.USN AND
C.SSID=SS.SSID
GROUP BY SEM,SEC,GENDER
ORDER BY SEM;
```

SEM	S	G	COUNT
3	B	M	2
4	C	F	1
4	C	M	1
5	A	F	1
5	A	M	1
7	A	F	1

3. Create a view of Test1 marks of student USN '1BI15CS101' in all subjects.

```
CREATE VIEW TEST1 AS
SELECT SUBCODE,TEST1
FROM IAMARKS
WHERE USN='1cg15ee065';
View created.
SQL> select * from test1;
```

SUBCODE	TEST1

15cs33	16
15cs53	19

4. Calculate the FinalIA (average of best two test marks) and update the corresponding table for all students.

```
CREATE OR REPLACE PROCEDURE AVG
IS
CURSOR C_IAMARKS IS
SELECT GREATEST(TEST1,TEST2) AS A,GREATEST(TEST1,TEST3) AS B,
GREATEST(TEST3,TEST2) AS C
FROM IAMARKS
WHERE FINALIA IS NULL

FOR UPDATE;
C_A NUMBER;
C_B NUMBER;
C_C NUMBER;
C_SM NUMBER;
C_AV NUMBER;
BEGIN
OPEN C_IAMARKS;
LOOP
FETCH C_IAMARKS INTO C_A,C_B,C_C;
EXIT WHEN C_IAMARKS%NOTFOUND;
DBMS_OUTPUT.PUT_LINE(C_A||' '||C_B||' '||C_C);
IF(C_A!=C_B) THEN
C_SM:=C_A+C_B;
ELSE
C_SM:=C_A+C_C;
END IF;
C_AV:=C_SM/2;
DBMS_OUTPUT.PUT_LINE('SUM='||C_SM);
DBMS_OUTPUT.PUT_LINE('AVERAGE='||C_AV);
UPDATE IAMARKS
SET FINALIA=C_AV
WHERE CURRENT OF C_IAMARKS;
END LOOP;
CLOSE C_IAMARKS;
END AVG;
Procedure created.

SQL> BEGIN
2 AVG;
3 END;
PL/SQL procedure successfully completed.
SQL> SELECT * FROM IAMARKS;
```

USN	SUBCODE	SSID	TEST1	TEST2	TEST3	FINALIA
lcg15cs001	15cs53	5A	18	19	15	19
lcg15cs002	15cs53	5A	15	16	14	16

lcg16me063	15cs53	3B	10	15	16	16
lcg14ec055	10cs71	7A	18	20	21	21
lcg15ee065	15cs53	3B	16	20	17	19
lcg15ee065	15cs53	4c	19	20	18	20

6 rows selected.

5. Categorize students based on the following criterion:

If FinalIA = 17 to 20 then CAT = 'Outstanding' If
FinalIA = 12 to 16 then CAT = 'Average'
If FinalIA< 12 then CAT = 'Weak'
Give these details only for 8th semester A, B, and C section students.

```
SELECT S.USN,S.SNAME,S.ADDRESS,S.PHONE,S.GENDER,
CASE WHEN IA.FINALIA BETWEEN 17 AND 20 THEN 'OUTSTANDING'
WHEN IA.FINALIA BETWEEN 12 AND 16 THEN 'AVERAGE'
ELSE 'WEAK'
END AS CAT
FROM STUDENT S,SEMSEC SS,IAMARKS IA,SUBJECT SUB
WHERE S.USN=IA.USN AND
SS.SSID=IA.SSID AND
SUB.SUBCODE=IA.SUBCODE AND
SUB.SEM=7
```

USN	SNAME	ADDRESS	PHONE	G	CAT
lcg14ec055	Raghavi	Sspuram	9485675521	F	Weak

CONCLUSION: Here we have different methods to do solve the problem. Many Softwares are used to do this such as MySQL,postgre,etc but we came using MYSQL because it is a better solution to come from as such we are already studying this so we reffered MySQL as a software to do our project. In thi9s we came under many queries and solved each query and illustrated our example more specifically.