International Islamic University Chittagong

Project Report

Learning Management System

Course Code: CSE-2424

Course Title: Database Management System Lab

Submitted To –

Mohammad Aman Ullah

Associate Professor, Department of

Computer Science and Engineering, IIUC

Submitted By:

Name: Sabrina Hossain

ID: C211211

Email Id: c211211@ugrad.iiuc.ac.bd

Semester Enrolled: 4th Section: 4AF

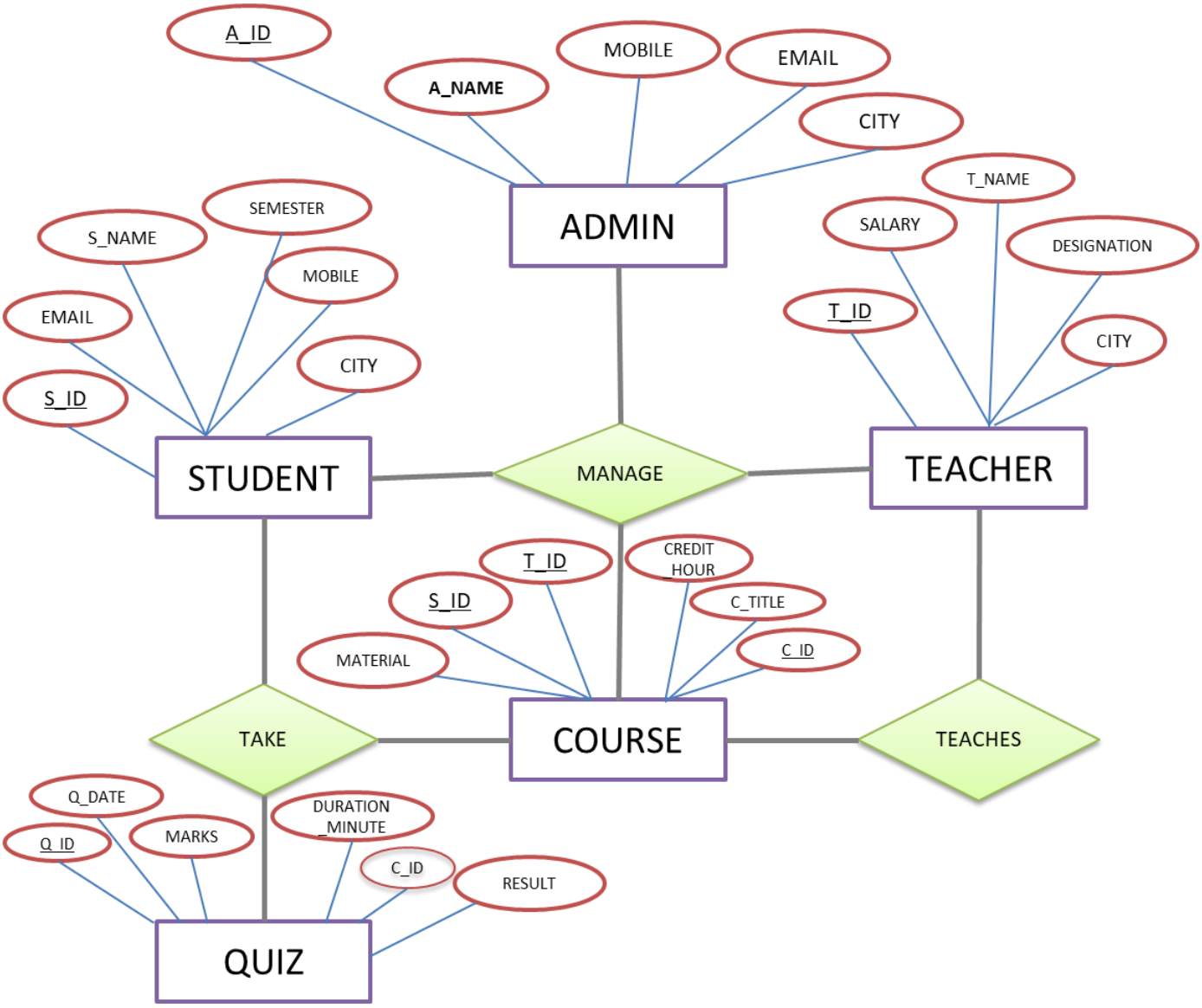
Submission Date- 27/11/2022

Abstract

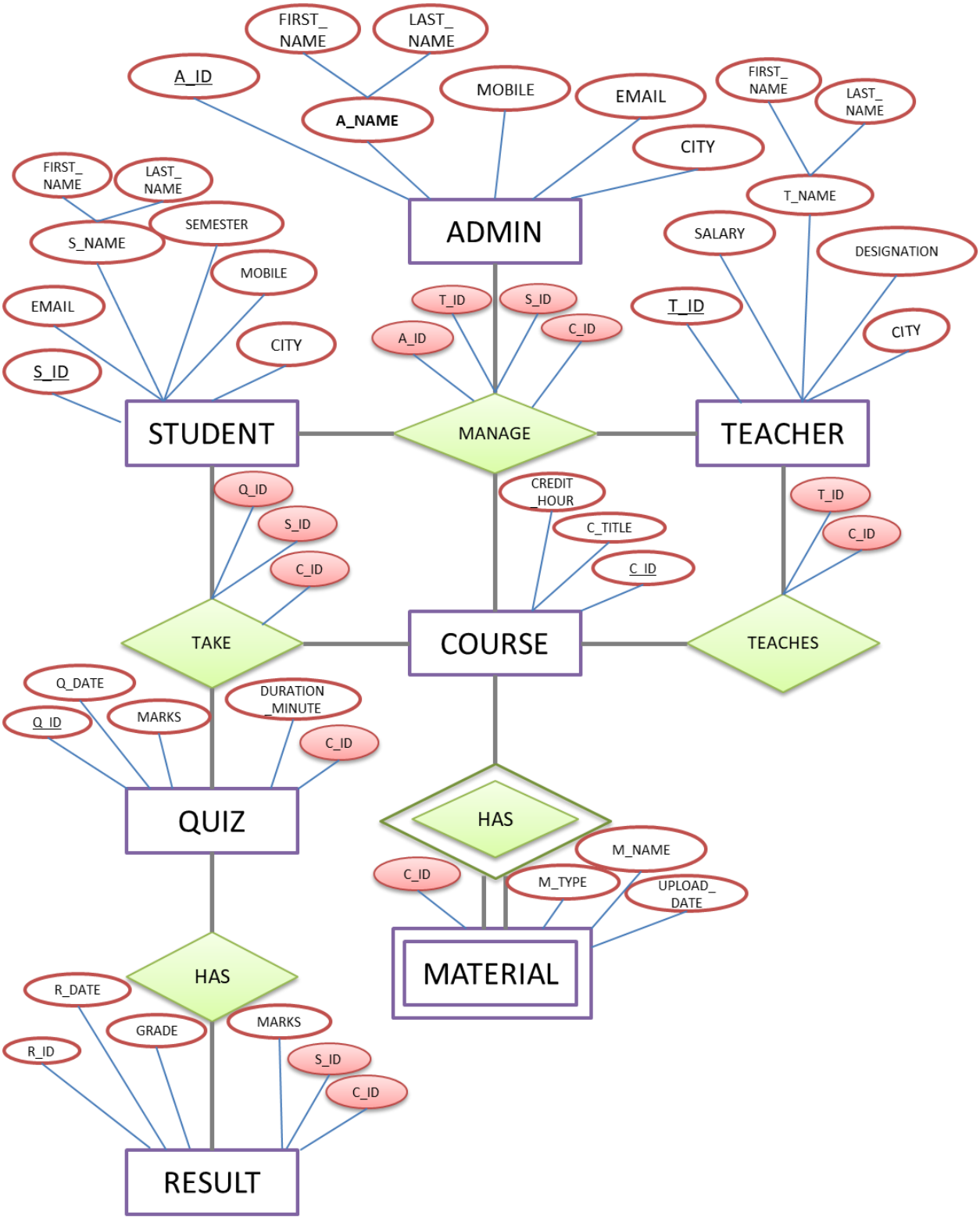
A Learning Management System (LMS) is a term used to describe software tools designed to manage user learning interventions. It is a client-server type solution, typically web-based technology used to plan, implement and assess a specific learning process. . LMS provide workspaces to facilitate information sharing and communication among students and lecturers to participate in course activities. The admin control and manage the whole system. The main features of LMS is both student and teacher can access the system, upload course regarding files, links, taking quiz, results and manage distance learning. An LMS is particularly useful for fully online courses or training programs.

Table of contents Page no

ERD-


A_ID
A_NAME
MOBILE
EMAIL
CITY
ADMIN
SEMESTER
S_NAME
MOBILE
EMAIL
CITY
S_ID
STUDENT
T_NAME
SALARY
DESIGNATION
T_ID
CITY
TEACHER
MANAGE
T_ID
CREDIT HOUR
S_ID
C_TITLE
C_ID
MATERIAL
COURSE
TAKE
TEACHES
Q_DATE
DURATION _MINUTE
Q_ID
MARKS
C_ID
RESULT
QUIZ

NORMALISED ER-DIAGRAM:



SCHEMA

ADMIN (A_ID, FIRST_NAME,LAST_NAME,MOBILE, EMAIL,ADDRESS,CITY)

STUDENT (S_ID, FIRST_NAME,LAST_NAME,SEMESTER,MOBILE,EMAIL,CITY)

TEACHER(T_ID,FIRST_NAME,LAST_NAME,MOBILE,EMAIL,DESIGNATION,SALARY,CITY)

COURSE (C_ID, C_TITLE,CREDIT_HOUR)

MATERIAL (M_NAME,M_TYPE,UPLOAD_DATE,C_ID)

QUIZ (Q_ID,Q_DATE,DURATION_MINUTE,FULL_MARKS ,C_ID)

RESULT (R_ID,R_DATE,MARKS,GRADE,C_ID, S_ID)

MANAGE (A_ID,T_ID,S_ID,C_ID)

TEACHES (T_ID,C_ID)

TAKE (S_ID,C_ID,Q_ID)

TABLE

1.ADMIN

SQL-

```
CREATE TABLE  "ADMIN"
   (    "A_ID" NUMBER(10,0),
        "FIRST_NAME" VARCHAR2(50),
        "LAST_NAME" VARCHAR2(50),
        "MOBILE" NUMBER(11,0),
        "EMAIL" VARCHAR2(100),
        "CITY" VARCHAR2(100),
         CONSTRAINT "ADMIN_PK" PRIMARY KEY ("A_ID") ENABLE
   )
/
```

2.STUDENT

SQL-

```
CREATE TABLE  "STUDENT"
   (    "S_ID" VARCHAR2(20),
        "FIRST_NAME" VARCHAR2(50),
        "LAST_NAME" VARCHAR2(50),
        "SEMESTER" NUMBER(5,0),
        "MOBILE" NUMBER(11,0),
        "EMAIL" VARCHAR2(100),
        "CITY" VARCHAR2(100),
         CONSTRAINT "STUDENT_PK" PRIMARY KEY ("S_ID") ENABLE
   )
/
```

3.TEACHER

SQL-

```
CREATE TABLE  "TEACHER"
   (    "T_ID" NUMBER(10,0),
        "FIRST_NAME" VARCHAR2(50),
        "LAST_NAME" VARCHAR2(50),
        "EMAIL" VARCHAR2(100),
        "DESIGNATION" VARCHAR2(100),
        "SALARY" NUMBER(12,0),
        "HIRE_DATE" DATE,
        "CITY" VARCHAR2(100),
         CONSTRAINT "TEACHER_PK" PRIMARY KEY ("T_ID") ENABLE
   )
```

```
/
```

4.COURSE

SQL-

```
CREATE TABLE  "COURSE"
   (    "C_ID" VARCHAR2(20),
        "C_TITLE" VARCHAR2(20),
        "CREDIT_HOUR" NUMBER(5,0),
         CONSTRAINT "COURSE_PK" PRIMARY KEY ("C_ID") ENABLE
   )
/
```

5.QUIZ

SQL-

```
CREATE TABLE  "QUIZ"
   (     "Q_ID" VARCHAR2(20),
        "Q_DATE" DATE,
        "DURATION MINUTE" NUMBER(10,0),
        "FULL_MARKS" NUMBER(10,0),
        "C_ID" VARCHAR2(20),
         CONSTRAINT "QUIZ_PK" PRIMARY KEY ("Q_ID") ENABLE,
         CONSTRAINT "QUIZ_FK" FOREIGN KEY ("C_ID")
          REFERENCES  "COURSE" ("C_ID") ENABLE
   )
/
```

6.RESULT

SQL-

```
CREATE TABLE  "RESULT"
   (    "R_ID" NUMBER(10,0),
        "R_DATE" DATE,
        "MARKS" NUMBER(10,0),
        "GRADE" VARCHAR2(10),
        "C_ID" VARCHAR2(20),
        "S_ID" VARCHAR2(20),
         CONSTRAINT "RESULT_PK" PRIMARY KEY ("R_ID") ENABLE,
         CONSTRAINT "RESULT_FK" FOREIGN KEY ("C_ID")
          REFERENCES  "COURSE" ("C_ID") ENABLE,
         CONSTRAINT "RESULT_FK2" FOREIGN KEY ("S_ID")
          REFERENCES  "STUDENT" ("S_ID") ENABLE
   )
/
```

7.MATERIAL

SQL-

```
CREATE TABLE  "MATERIAL"
   (    "M NAME" VARCHAR2(100),
        "M_TYPE" VARCHAR2(100),
        "UPLOAD_DATE" DATE,
        "C_ID" VARCHAR2(20),
         CONSTRAINT "MATERIAL_FK" FOREIGN KEY ("C_ID")
          REFERENCES  "COURSE" ("C_ID") ENABLE
   )
/
```

8.MANAGE

SQL-

```
CREATE TABLE  "MANAGE"
   (     "A_ID" NUMBER(10,0),
         "C_ID" VARCHAR2(20),
         "T_ID" NUMBER(10,0),
         "S_ID" VARCHAR2(20),
          CONSTRAINT "MANAGE_FK" FOREIGN KEY ("A_ID")
           REFERENCES  "ADMIN" ("A_ID") ENABLE,
          CONSTRAINT "MANAGE_FK2" FOREIGN KEY ("C_ID")
           REFERENCES  "COURSE" ("C_ID") ENABLE,
          CONSTRAINT "MANAGE_FK3" FOREIGN KEY ("T_ID")
           REFERENCES  "TEACHER" ("T_ID") ENABLE,
          CONSTRAINT "MANAGE_FK4" FOREIGN KEY ("S_ID")
           REFERENCES  "STUDENT" ("S_ID") ENABLE
   )
/
```

9.TAKE

SQL-

```
CREATE TABLE  "TAKE"
   (     "S_ID" VARCHAR2(20),
         "C_ID" VARCHAR2(20),
         "Q_ID" VARCHAR2(20),
          CONSTRAINT "TAKE_FK" FOREIGN KEY ("S_ID")
           REFERENCES  "STUDENT" ("S_ID") ENABLE,
          CONSTRAINT "TAKE FK2" FOREIGN KEY ("C ID")
           REFERENCES  "COURSE" ("C_ID") ENABLE,
          CONSTRAINT "TAKE_FK3" FOREIGN KEY ("Q_ID")
```

```
          REFERENCES  "QUIZ" ("Q ID") ENABLE
   )
/
```

10.TEACHES

SQL-

```
CREATE TABLE  "TEACHES"
   (    "T_ID" NUMBER(10,0),
        "C_ID" VARCHAR2(20),
         CONSTRAINT "TEACHES_FK2" FOREIGN KEY ("T_ID")
          REFERENCES  "TEACHER" ("T_ID") ENABLE,
         CONSTRAINT "TEACHES_FK3" FOREIGN KEY ("C_ID")
          REFERENCES  "COURSE" ("C_ID") ENABLE
   )
/
```

DATA:

1.ADMIN

EDIT	A_ID	FIRST_NAME	LAST_NAME	MOBILE	EMAIL	CITY
	12340	Anisul	Islam	1737478849	anisul@gmail.com	Chattogram

row(s) 1 - 1 of 1

Download

2.STUDENT

EDIT	S_ID	FIRST_NAME	LAST_NAME	SEMESTER	MOBILE	EMAIL	CITY
	A111	sima	akter	1	1353278223	sima@gmail.com	Barishal
	E212	sujit	sarker	2	46809057	sujit@gmail.com	Mymensingh
	C221	arman	hossen	2	7663768987	arman@gmail.com	Chattogram
	P201	fatema	akter	1	978985456	fatema@yahoo.com	Chattogram
	P202	kurratul	ain	1	1256788765	kain@gmail.com	Dhaka
	M201	tonmoy	mia	2	9867647645	tonmoy@gmail.com	khulna
	M202	priya	hoque	2	698599087	phq@gmail.com	khulna
	A201	biddya	sinha	3	6765327345	bidda@gmail.com	Jessore
	E215	faria	sultana	2	95976346	sultanf@gmail.com	Feni
	D12	Rahim	uddin	4	6782343473	uddinrahim@yahoo.com	Barishal
	D115	tanya	akter	4	1298765439	takter@gmail.com	Dhaka
	B112	shopia	hoque	1	3056790089	shopia@gmail.com	Mymensingh
	M203	Abdur	Rahman	2	7969234780	arrahman@yahoo.com	sylhet
	G312	karim	ahmed	3	98565673	karim1@gmail.com	Chattogram
	A203	mitu	barua	3	4576546876	mitub@gmail.com	Chattogram

row(s) 1 - 15 of 30 

Download

3.TEACHER

EDIT	T_ID	FIRST_NAME	LAST_NAME	EMAIL	DESIGNATION	SALARY	HIRE_DATE	CITY
	1128	Raisa	Alam	raisa@gmail.com	lecturer	20000	20-MAR-15	Comilla
	1171	Timu	Islam	timu@yahoo.com	Assistant professor	25000	10-FEB-11	Rajshahi
	1187	Rupali	Akter	akterupali@gmail.com	Professor	30000	20-JAN-16	Barishal
	1251	Mosaddek	Hossen	hmosaddek@gmail.com	lecturer	15000	12-JAN-19	Khulna
	1256	Sifat	Rahaman	sifat@yahoo.co,	Assistant Professor	20000	12-JAN-19	Chattogram
	1107	Lisa	Rahman	lisa1@gmail.com	Assistant professor	25000	06-MAY-15	Rangpur
	1126	Ruhul	Amin	ramin@gmail.com	Assistant professor	30000	20-JAN-16	Rangpur
	1192	Mahmud	Akber	makber@gmail.com	lecturer	20000	08-SEP-19	Chattogram
	1321	Adam	Smith	asmith@gmail.com	Professor	30000	13-JUN-18	Dhaka
	1105	Mohammad	Arif	arif@gmail.com	lecturer	15000	12-MAR-18	CHATTOGRAM
	1125	Asif	Amin	amin@yahoo.com	Assistant professor	25000	12-FEB-19	khulna
	1152	Abdul	Latif	alatif@gmail.com	Assistant professor	30000	22-JAN-14	Dhaka
	1232	Manjur	Alam	manju@yahoo.com	Assitant Professor	25000	24-MAY-15	Dhaka
	1168	Mark	Luther	mluther@gmail.com	Professor	50000	14-JUN-15	Dhaka
							row(s) 1 - 14 of 14	

Download

4.COURSE

EDIT	C_ID	C_TITLE	CREDIT_HOUR
	A101	ALGORITHM	3
	D101	DATABASE	3
	E101	ECONOMICS	1
	P101	PHYSICS	2
	M101	MATHEMATICS	3
	C102	COMPUTER	2
	G101	GEOGRAPHY	1
	A102	ACCOUNTING	2
	C101	CHEMISTRY	2
	B101	BIOLOGY	1
			row(s) 1 - 10 of 10

Download

5.MATERIAL

EDIT	M_NAME	M_TYPE	UPLOAD_DATE	C_ID
	CLASS_1	LINK	20-APR-19	A102
	LECTURE_2	VIDEO	23-FEB-20	C102
	LESSON_2	PDF	28-FEB-18	A101
	BOOK	PDF	01-JAN-16	M101
	LESSON_1	PDF	02-FEB-18	A101
			row(s) 1 - 5 of 5	

Download

6.QUIZ

EDIT	Q_ID	Q_DATE	DURATION_MINUTE	FULL_MARKS	C_ID
	C31	25-MAY-20	20	20	C102
	A52	12-MAR-18	30	20	A101
	D33	02-JAN-18	30	30	D101
	G21	22-JUN-20	10	10	G101
	A62	25-JUN-19	15	10	A102
	C23	26-APR-17	15	10	C101
	B50	21-MAR-20	20	20	B101
	E14	02-JAN-17	10	10	E101
	P41	03-MAR-18	15	10	P101
	M91	21-FEB-16	20	20	M101
				row(s) 1 - 10 of 10	

Download

7.RESULT

EDIT	R_ID	R_DATE	MARKS	GRADE	C_ID	S_ID
	8	24-JUN-19	9	A	A102	A205
	13	21-MAR-20	19	A	B101	B113
	14	02-JAN-18	25	A	D101	D111
	29	22-JUN-20	8	A	G101	G311
	2	12-MAR-18	13	B	A101	A112
	4	24-JUN-19	5	C	A102	A201
	5	24-JUN-19	8	A	A102	A202
	9	26-APR-17	9	A	C101	C221
	11	21-MAR-20	15	B	B101	B112
	17	02-JAN-17	6	B	E101	E211
	19	02-JAN-17	5	C	E101	E213
	20	02-JAN-17	10	A	E101	E214
	23	03-MAR-18	6	B	P101	P202
	24	25-MAY-20	18	A	C102	C202
	27	21-FEB-16	16	A	M101	M202

row(s) 1 - 15 of 29

Download

8.MANAGE

EDIT	A_ID	C_ID	T_ID	S_ID
	12340	E101	1152	E214
	12340	E101	1152	E215
	12340	D101	1168	D111
	12340	D101	1168	D115
	12340	C102	1256	C201
	12340	B101	1105	B112
	12340	B101	1171	B113
	12340	A101	1126	A112
	12340	A102	1187	A203
	12340	E101	1192	E212
	12340	P101	1251	P202
	12340	G101	1321	G312
	12340	A101	1126	A111
	12340	A101	1128	A113
	12340	A102	1187	A202

row(s) 1 - 15 of 30

Download

9.TAKE

EDIT	S_ID	C_ID	Q_ID
	A111	A101	A52
	A202	A102	A62
	M202	M101	M91
	G312	G101	G21
	E211	E101	E14
	E212	E101	E14
	E214	E101	E14
	A112	A101	A52
	C221	C101	C23
	C222	C101	C23
	B112	B101	B50
	B112	B101	B50
	E213	E101	E14
	D12	D101	D33
	D115	D101	D33
	row(s) 1 - 15 of 30		

Download

10.TEACHES

EDIT	T_ID	C_ID
	1125	C101
	1321	G101
	1251	P101
	1105	B101
	1107	A102
	1128	A101
	1187	A102
	1152	E101
	1171	B101
	1168	D101
	1126	A101
	1192	E101
	1256	C102
	1232	M101
row(s) 1 - 14 of 14		

Download

DML QUERIES FROM SINGLE TABLE

1. **Show the students from student table whose semester=2 .**

ANS-

```
SELECT* FROM STUDENT

WHERE SEMESTER=2;
```

OUTPUT-

S_ID	FIRST_NAME	LAST_NAME	SEMESTER	MOBILE	EMAIL	CITY
E212	sujit	sarker	2	46809057	sujit@gmail.com	Mymensingh
C221	arman	hossen	2	7663768987	arman@gmail.com	chattogram
M201	tonmoy	mia	2	9867647645	tonmoy@gmail.com	KHULNA
M202	priya	hoque	2	698599087	phq@gmail.com	khulna
E215	faria	sultana	2	95976346	sultanf@gmail.com	Feni
M203	Abdur	Rahman	2	7969234780	arrahman@yahoo.com	sylhet
E211	bhumi	sarkar	2	567845790	bhumi@yahoo.com	Mymensingh
E213	habib	uddola	2	4325635877	habib@yahoo.com	DHAKA
E214	vicky	sharma	2	324598780	vicks@gmail.com	Barishal
C222	ifaz	uddin	2	813243687	ifaz1@gmail.com	Rangpur

10 rows returned in 0.00 seconds CSV Export

2. Show the details of the teachers whose salary is less than 25000.

ANS-

```
SELECT* FROM TEACHER

Where SALARY < 25000;
```

OUTPUT-

T_ID	FIRST_NAME	LAST_NAME	EMAIL	DESIGNATION	SALARY	HIRE_DATE	CITY
1128	Raisa	Alam	raisa@gmail.com	lecturer	20000	20-MAR-15	Comilla
1251	Mosaddek	Hossen	hmosaddek@gmail.com	lecturer	15000	12-JAN-19	Khulna
1256	Sifat	Rahaman	sifat@yahoo.co,	Assistant Professor	20000	12-JAN-19	Chattogram
1192	Mahmud	Akber	makber@gmail.com	lecturer	20000	08-SEP-19	Chattogram
1105	Mohammad	Arif	arif@gmail.com	lecturer	15000	12-MAR-18	CHATTOGRAM

5 rows returned in 0.00 seconds CSV Export

3**. Show the all course record where course Id is (A102, C101, D101, E101)**

ANS-

```
SELECT* FROM COURSE

WHERE C_ID IN('A102','C101','D101','E101');
```

OUTPUT-

C_ID	C_TITLE	CREDIT_HOUR
A102	ACCOUNTING	2
C101	CHEMISTRY	2
D101	DATABASE	3
E101	ECONOMICS	1

4 rows returned in 0.00 seconds CSV Export

4. Show the record from teacher table where FIRST_NAME and DESIGNATION must have to contain at least one t character.

ANS-

```
SELECT* FROM TEACHER

WHERE FIRST_NAME like '%t%' and DESIGNATION like '%t%' ;
```

OUTPUT-

T_ID	FIRST_NAME	LAST_NAME	EMAIL	DESIGNATION	SALARY	HIRE_DATE	CITY
1256	Sifat	Rahaman	sifat@yahoo.co,	Assistant Professor	20000	12-JAN-19	Chattogram

1 rows returned in 0.00 seconds CSV Export

5. Show the FIRST_NAME and ID in which position the character 'm' is locating from Student table.

ANS-

```
SELECT S_ID, FIRST_NAME, instr(FIRST_NAME,'m')

 FROM STUDENT

WHERE instr(FIRST_NAME,'m')!=0 ;
```

OUTPUT-

S_ID	FIRST_NAME	INSTR(FIRST_NAME,'M')
A111	sima	3
C221	arman	3
P201	fatema	5
M201	tonmoy	4
D12	Rahim	5
G312	karim	5
A203	mitu	1
E211	bhumi	4
D111	yesmin	4
C202	rumi	3
B113	jamil	3
C201	Rahim	5

12 rows returned in 0.00 seconds CSV Export

6. Find the Result details of the students having 8 in MARKS and their GRADE is A.

ANS-

```
SELECT *FROM RESULT

WHERE MARKS like'8%' and GRADE LIKE 'A%';
```

OUTPUT-

R_ID	R_DATE	MARKS	GRADE	C_ID	S_ID
29	22-JUN-20	8	A	G101	G311
5	24-JUN-19	8	A	A102	A202
6	24-JUN-19	8	A	A102	A203
10	26-APR-17	8	A	C101	C222
18	02-JAN-17	8	A	E101	E212
22	03-MAR-18	8	A	P101	P201

6 rows returned in 0.00 seconds CSV Export

7.Show the ID ,Full name and email of the teachers whose experience is more than 6 years.

ANS-

```
SELECT T_ID,

FIRST_NAME,

LAST_NAME,

EMAIL FROM TEACHER

WHERE round((sysdate-HIRE_DATE)/365)>6;
```

OUTPUT-

T_ID	FIRST_NAME	LAST_NAME	EMAIL
1128	Raisa	Alam	raisa@gmail.com
1171	Timu	Islam	timu@yahoo.com
1187	Rupali	Akter	akterupali@gmail.com
1107	Lisa	Rahman	lisa1@gmail.com
1126	Ruhul	Amin	ramin@gmail.com
1152	Abdul	Latif	alatif@gmail.com
1232	Manjur	Alam	manju@yahoo.com
1168	Mark	Luther	mluther@gmail.com

8 rows returned in 0.00 seconds CSV Export

8.Show the maximum marks from RESULT.

ANS-

```
SELECT MAX(MARKS)
FROM RESULT;
```

OUTPUT-

MAX(MARKS)
27

1 rows returned in 0.00 seconds CSV Export

9. Show Q_DATE in ascending order from Quiz table.

ANS-

```
SELECT Q_ID, Q_DATE, DURATION_MINUTE, FULL_MARKS
FROM QUIZ
ORDER BY Q_DATE asc;
```

OUTPUT-

Q_ID	Q_DATE	DURATION_MINUTE	FULL_MARKS
M91	21-FEB-16	20	20
E14	02-JAN-17	10	10
C23	26-APR-17	15	10
D33	02-JAN-18	30	30
P41	03-MAR-18	15	10
A52	12-MAR-18	30	20
A62	25-JUN-19	15	10
B50	21-MAR-20	20	20
C31	25-MAY-20	20	20
G21	22-JUN-20	10	10

10 rows returned in 0.00 seconds CSV Export

10. Find the sum of the quiz time from quiz table.

ANS-

```
SELECT sum(DURATION_MINUTE)
FROM QUIZ;
```

OUTPUT-

SUM(DURATION_MINUTE)
185

1 rows returned in 0.00 seconds CSV Export

11. Find the average marks from Result table.

ANS-

SELECT AVG(MARKS) FROM RESULT;

OUTPUT-

AVG(MARKS)
12.133333333333333333333333333333333333333

1 rows returned in 0.00 seconds CSV Export

12. Combine the names in a column named as 'FULL_NAME' from the student table.

ANS-

SELECT FIRST_NAME || ' ' ||LAST_NAME AS "FULL_NAME" FROM STUDENT;

OUTPUT-

FULL_NAME
sima akter
sujit sarker
arman hossen
fatema akter
kurratul ain
tonmoy mia
priya hoque
biddya sinha
faria sultana
Rahim uddin
More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds CSV Export

DML FROM MULTIPLE TABLE

1.Show all records from course and quiz table.

ANS-

select* from course,quiz;

OUTPUT-

C_ID	C_TITLE	CREDIT_HOUR	Q_ID	Q_DATE	DURATION_MINUTE	FULL_MARKS	C_ID
A101	ALGORITHM	3	C31	25-MAY-20	20	20	C102
D101	DATABASE	3	C31	25-MAY-20	20	20	C102
E101	ECONOMICS	1	C31	25-MAY-20	20	20	C102
P101	PHYSICS	2	C31	25-MAY-20	20	20	C102
M101	MATHEMATICS	3	C31	25-MAY-20	20	20	C102
C102	COMPUTER	2	C31	25-MAY-20	20	20	C102
G101	GEOGRAPHY	1	C31	25-MAY-20	20	20	C102
A102	ACCOUNTING	2	C31	25-MAY-20	20	20	C102
C101	CHEMISTRY	2	C31	25-MAY-20	20	20	C102
B101	BIOLOGY	1	C31	25-MAY-20	20	20	C102
More than 10 rows available. Increase rows selector to view more rows.							

10 rows returned in 0.00 seconds CSV Export

2. **Show the grade in descending order from the quiz table and rename s_id column and q_id column as' student_id' and 'quiz_name'.**

   ANS-

   SELECT STUDENT.S_ID AS "STUDENT_ID", QUIZ.Q_ID AS "QUIZ_NAME",RESULT.GRADE AS "GRADE"
   FROM STUDENT ,QUIZ,RESULT
   ORDER BY GRADE DESC;

   OUTPUT-

STUDENT_ID	QUIZ_NAME	GRADE
M202	P41	C
M203	P41	C
P201	P41	C
P202	P41	C
E215	G21	C
G311	G21	C
G312	G21	C
M201	G21	C
M202	G21	C
M203	G21	C
P201	G21	C
P202	G21	C
A111	M91	C
A112	M91	C
A113	M91	C
A201	M91	C
A202	M91	C
A203	M91	C
A204	M91	C
A205	M91	C
More than 20 rows available. Increase rows selector to view more rows.		

20 rows returned in 0.00 seconds CSV Export

3. **Show all the records of the student and teacher who have their last name in common.**

   ANS-

   SELECT*
   FROM TEACHER,STUDENT
   WHERE TEACHER.LAST_NAME=STUDENT.LAST_NAME;

   OUTPUT-

T_ID	FIRST_NAME	LAST_NAME	EMAIL	DESIGNATION	SALARY	HIRE_DATE	CITY	S_ID	FIRST_NAME	LAST_NAME	SEMESTER	MOBILE	EMAIL	CITY
1107	Lisa	Rahman	lisa1@gmail.com	Assistant professor	25000	06-MAY-15	Rangpur	M203	Abdur	Rahman	2	7969234780	arrahman@yahoo.com	sylhet

1 rows returned in 0.00 seconds CSV Export

4. **Show all the data from both TAKE and RESULT table where course id is same using left outer join.**

   ANS-

   SELECT*
   FROM TAKE left outer join RESULT on(TAKE.C_ID=RESULT.C_ID);

   OUTPUT-

S_ID	C_ID	Q_ID	R_ID	R_DATE	MARKS	GRADE	C_ID	S_ID
A201	A102	A62	8	24-JUN-19	9	A	A102	A205
A205	A102	A62	8	24-JUN-19	9	A	A102	A205
A204	A102	A62	8	24-JUN-19	9	A	A102	A205
A203	A102	A62	8	24-JUN-19	9	A	A102	A205
A202	A102	A62	8	24-JUN-19	9	A	A102	A205
B113	B101	B50	13	21-MAR-20	19	A	B101	B113
B112	B101	B50	13	21-MAR-20	19	A	B101	B113
B112	B101	B50	13	21-MAR-20	19	A	B101	B113
D111	D101	D33	14	02-JAN-18	25	A	D101	D111
D115	D101	D33	14	02-JAN-18	25	A	D101	D111
More than 10 rows available. Increase rows selector to view more rows.								

10 rows returned in 0.00 seconds CSV Export

5. Show the Cartesian product of course and result table.

ANS-

SELECT *

FROM COURSE CROSS JOIN RESULT

OUTPUT-

C_ID	C_TITLE	CREDIT_HOUR	R_ID	R_DATE	MARKS	GRADE	C_ID	S_ID
A101	ALGORITHM	3	8	24-JUN-19	9	A	A102	A205
A101	ALGORITHM	3	13	21-MAR-20	19	A	B101	B113
A101	ALGORITHM	3	14	02-JAN-18	25	A	D101	D111
A101	ALGORITHM	3	29	22-JUN-20	8	A	G101	G311
A101	ALGORITHM	3	2	12-MAR-18	13	B	A101	A112
A101	ALGORITHM	3	4	24-JUN-19	5	C	A102	A201
A101	ALGORITHM	3	5	24-JUN-19	8	A	A102	A202
A101	ALGORITHM	3	9	26-APR-17	9	A	C101	C221
A101	ALGORITHM	3	11	21-MAR-20	15	B	B101	B112
A101	ALGORITHM	3	12	21-MAR-20	7	C	B101	B112
More than 10 rows available. Increase rows selector to view more rows.								

10 rows returned in 0.00 seconds CSV Export

6. Show the id and first name of the teachers and students who reside in same city.

ANS-

SELECT TEACHER.T_ID AS "TEACHER_ID", TEACHER.FIRST_NAME AS "TEACHER",

STUDENT.S_ID, STUDENT.FIRST_NAME AS "STUDENT_NAME" , STUDENT.CITY

FROM TEACHER, STUDENT

WHERE TEACHER.CITY=STUDENT.CITY;

OUTPUT-

TEACHER_ID	TEACHER	S_ID	STUDENT_NAME	CITY
1125	Asif	C202	rumi	khulna
1126	Ruhul	C222	ifaz	Rangpur
1107	Lisa	C222	ifaz	Rangpur
1128	Raisa	B111	Babu	Comilla
1168	Mark	A113	aysha	Dhaka
1232	Manjur	A113	aysha	Dhaka
1152	Abdul	A113	aysha	Dhaka
1321	Adam	A113	aysha	Dhaka

8 rows returned in 0.00 seconds CSV Export

SUB QUERIES

1.Show the teacher id and names along with their salary, but only for the one where salary is more than average.

Ans-

```
SELECT T_ID,FIRST_NAME ,LAST_NAME
FROM TEACHER
WHERE SALARY>(SELECT AVG(SALARY)FROM TEACHER);
```

OUTPUT-

T_ID	FIRST_NAME	LAST_NAME
1187	Rupali	Akter
1126	Ruhul	Amin
1321	Adam	Smith
1152	Abdul	Latif
1168	Mark	Luther

5 rows returned in 0.00 seconds CSV Export

2.Find the result details of the student whose first name is “shopia” .

ANS-

```
SELECT *
FROM RESULT
WHERE S_ID =
   (SELECT S_ID
    FROM STUDENT
    WHERE FIRST_NAME='shopia');
```

OUTPUT-

R_ID	R_DATE	MARKS	GRADE	C_ID	S_ID
11	21-MAR-20	15	B	B101	B112

1 rows returned in 0.00 seconds CSV Export

3. **Show the id and first name whose marks are greater than 16.**

ANS-

```
SELECT a.S_ID, a.FIRST_NAME, b.MARKS
FROM STUDENT a,RESULT b
WHERE a.S_ID =b.S_ID AND b.MARKS>16;
```

OUTPUT-

S_ID	FIRST_NAME	MARKS
M201	tonmoy	20
D12	Rahim	27
D115	tanya	23
D111	yesmin	25
C202	rumi	18
B113	jamil	19
A113	aysha	18

7 rows returned in 0.00 seconds CSV Export

4. **Show the teacher id and name who receive a higher salary than the teacher whose id=1107.**

ANS-

```
SELECT T_ID, FIRST_NAME,LAST_NAME
FROM TEACHER
WHERE SALARY >
( SELECT SALARY
FROM TEACHER
WHERE T_ID=1107);
```

OUTPUT-

T_ID	FIRST_NAME	LAST_NAME
1187	Rupali	Akter
1126	Ruhul	Amin
1321	Adam	Smith
1152	Abdul	Latif
1168	Mark	Luther

5 rows returned in 0.00 seconds CSV Export

5. **Find the student name whose mobile number ends with 5 and whose city name starts with 'J'.**

ANS-

```
SELECT* FROM STUDENT
WHERE CITY=(SELECT CITY FROM STUDENT WHERE CITY LIKE '%J%' AND MOBILE LIKE '%5%');
```

OUTPUT-

S_ID	FIRST_NAME	LAST_NAME	SEMESTER	MOBILE	EMAIL	CITY
A201	biddya	sinha	3	6765327345	bidda@gmail.com	Jessore

1 rows returned in 0.00 seconds CSV Export

6. **Find the details of the teachers whose salary is in the range 15000 and 20000.**

ANS-

```
SELECT * FROM TEACHER
WHERE SALARY BETWEEN 15000 and 20000;
```

OUTPUT-

T_ID	FIRST_NAME	LAST_NAME	EMAIL	DESIGNATION	SALARY	HIRE_DATE	CITY
1128	Raisa	Alam	raisa@gmail.com	lecturer	20000	20-MAR-15	Comilla
1251	Mosaddek	Hossen	hmosaddek@gmail.com	lecturer	15000	12-JAN-19	Khulna
1256	Sifat	Rahaman	sifat@yahoo.co,	Assistant Professor	20000	12-JAN-19	Chattogram
1192	Mahmud	Akber	makber@gmail.com	lecturer	20000	08-SEP-19	Chattogram
1105	Mohammad	Arif	arif@gmail.com	lecturer	15000	12-MAR-18	CHATTOGRAM

5 rows returned in 0.00 seconds CSV Export

PL/SQL QUERIES

1. **Fetch multiple data from STUDENT table using cursor.**

ANS-

```
DECLARE
std_record STUDENT%rowtype;
cursor std IS
select*
```

```
from STUDENT;
BEGIN
open std;
loop
fetch std INTO std_record;
exit when std%notfound;
dbms_output.put_line('the details of the student'||std_record.S_ID||' name:
'||std_record.FIRST_NAME ||' city: '|| std_record.CITY);
END loop;
CLOSE std;
END;
```

OUTPUT-

Results Explain Describe Saved SQL History

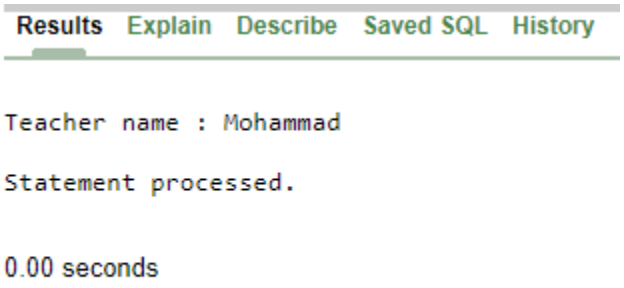
```
the details of the studentA111 name: sima city: Barishal

the details of the studentE212 name: sujit city: Mymensingh

the details of the studentC221 name: arman city: Chattogram

the details of the studentP201 name: fatema city: Chattogram

the details of the studentP202 name: kurratul city: Dhaka

the details of the studentM201 name: tonmoy city: khulna

the details of the studentM202 name: priya city: khulna

the details of the studentA201 name: biddya city: Jessore

the details of the studentE215 name: faria city: Feni

the details of the studentD12 name: Rahim city: Barishal

the details of the studentD115 name: tanya city: Dhaka

the details of the studentB112 name: shopia city: Mymensingh

the details of the studentM203 name: Abdur city: sylhet

the details of the studentG312 name: karim city: Chattogram

the details of the studentA203 name: mitu city: Chattogram

the details of the studentA204 name: zara city: khulna
the details of the studentA205 name: Joy city: Chattogram

the details of the studentE211 name: bhumi city: Mymensingh

the details of the studentE213 name: habib city: Dhaka
the details of the studentE214 name: vicky city: Barishal

the details of the studentD111 name: yesmin city: Dhaka

the details of the studentC202 name: rumi city: khulna
the details of the studentC222 name: ifaz city: Rangpur
the details of the studentB111 name: Babu city: Comilla
the details of the studentB113 name: jamil city: Chattogram

the details of the studentG311 name: rina city: khulna

the details of the studentA112 name: roton city: Rangpur

the details of the studentA113 name: aysha city: Dhaka
the details of the studentA202 name: arif city: Rangpur

the details of the studentC201 name: Rahim city: Rajshahi

Statement processed.

0.00 seconds
```

2. Show the teachers first name whose id=1105.

ANS-

```
DECLARE
teacher_rec TEACHER%rowtype;
BEGIN
 select * INTO teacher_rec
 FROM TEACHER
 WHERE T_ID=1105;
 dbms_output.put_line('Teacher name : '||teacher_rec.FIRST_NAME);
END;
```

OUTPUT-



3. Declare string variable using PL/SQL.

ANS-

```
DECLARE
  name varchar2(20);
  designation varchar2(30);
  introduction clob;
  choice char(1);
BEGIN
  name := 'Mark Luther';
 designation := 'Professor';
  introduction := ' Hello! I''m Mark Luther from Learning management system.';
  choice := 'y';
  IF choice = 'y' THEN
    dbms_output.put_line(name);
    dbms_output.put_line(designation);
    dbms_output.put_line(introduction);
  END IF;
END;
```

OUTPUT-


```
Mark Luther
Professor
 Hello! I'm Mark Luther from Learning management system.

Statement processed.
```

4. **Check if one of the value is greater between two numbers.**

ANS-

```
DECLARE
  a number:=15;
  b float:=12;
  c real;
PROCEDURE fe(x IN number,y IN number, z out number) IS

BEGIN
 IF x>y THEN
   dbms_output.put_line('x is greater then y');
   z:=x;
 ELSE
    dbms_output.put_line('y is greater then x');
    z:=y;

 END IF;
END;
BEGIN
  fe(a,b,c);
  dbms_output.put_line('The value  is: '||c);
END
```

OUTPUT-


```
x is greater then y
The value  is: 15

Statement processed.


0.00 seconds
```

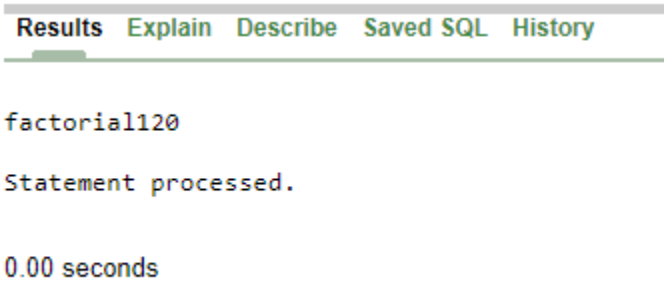
5. **Find the factorial of the given input.**

ANS-

```
declare
n number;
fac number :=1;
i number;

begin
n:=:input;
for i in 1..n
loop
fac:=fac*i;
end loop;
dbms_output.put_line('factorial ='||fac);
end;
```

INPUT GIVEN: 5

OUTPUT-



Conclusion

LMS offers a complete educational framework for students with several learning resources like online content and videos, documents and several courses. Other than the live classes and interaction, one can also find all the learning material related to their course stored on the LMS database. A learning management system database helps the institutions or the persons engaged in the same to enhance the quality of the teaching-learning process and cut many costs and other expenditures etc.

____ END ____