

### **#EXERCISE 1:**

```
CREATE DATABASE SQL_Assignment;
```

```
SHOW DATABASES;
```

```
USE SQL_Assignment;
```

```
CREATE TABLE Trainer_Info(
```

```
Trainer_ID VARCHAR(20) PRIMARY KEY,
```

```
SALUTATION VARCHAR(7),
```

```
Trainer_Name VARCHAR (30),
```

```
Trainer_Location VARCHAR(30),
```

```
Trainer_Track VARCHAR(15),
```

```
Trainer_Qualification VARCHAR (100),
```

```
Trainer_Experience INTEGER(11),
```

```
Trainer_Email VARCHAR(100),
```

```
Trainer_Password VARCHAR(20));
```

```
SELECT * FROM Trainer_Info;
```

```
CREATE TABLE Batch_Info(
```

```
Batch_ID VARCHAR(20) PRIMARY KEY,
```

```
Batch_Owner VARCHAR (30),
```

```
Batch_BU_Name VARCHAR(30));
```

```
SELECT * FROM Batch_Info;
```

```
CREATE TABLE Module_Info(
```

```
Module_ID VARCHAR(20) PRIMARY KEY,
```

```
Module_Name VARCHAR (40),
```

```
Module_Duration INTEGER(11));
```

```
SELECT * FROM Module_Info;
```

```
CREATE TABLE Associate_Info(
```

```
Associate_ID VARCHAR(20) PRIMARY KEY,
```

```
SALUTATION VARCHAR(7),
```

```
Associate_Name VARCHAR (30),
```

```
Associate_Location VARCHAR(30),
```

```
Associate_Track VARCHAR(15),
```

```
Associate_Qualification VARCHAR (200),
```

```

Associate_Email VARCHAR(100),
Associate_Password VARCHAR(20));
SELECT * FROM Associate_Info;
CREATE TABLE Questions(
Question_Id Varchar(20) PRIMARY KEY,
Question_Text Varchar(900),
Module_Id Varchar(20),
Constraint o_fk Foreign Key(Module_Id) REFERENCES Module_info(Module_Id));
SELECT * FROM Questions;
CREATE TABLE Associate_Status(
Associate_id Varchar(20),
Module_id Varchar(20),
Start_Date date,
End_Date date,
AFeedbackGiven bool,
TFeedbackGiven bool,
PRIMARY KEY(Associate_id,Module_id),
FOREIGN KEY(associate_id) REFERENCES Associate_info(Associate_id),
FOREIGN KEY(module_id) REFERENCES Module_info(Module_id));
SELECT * FROM Associate_Status;
Alter table Associate_Status
Add batch_id Varchar(20);
Alter table Associate_Status
add Foreign key (Batch_id) REFERENCES batch_info(batch_id);
Alter table Associate_Status
Add Trainer_id Varchar(20);
Alter table Associate_Status
add Foreign key (trainer_id) REFERENCES trainer_info(trainer_id);
SELECT *from Associate_Status;
CREATE TABLE Trainer_Feedback(
Trainer_Id Varchar(20),
Question_Id Varchar(20),
Batch_Id Varchar(20),
Module_Id Varchar(20),

```

```

Trainer_Rating Integer,
FOREIGN KEY(Trainer_Id) REFERENCES Trainer_info(Trainer_Id),
FOREIGN KEY(Question_Id) REFERENCES Questions(Question_Id),
FOREIGN KEY(Batch_Id) REFERENCES Batch_info(Batch_Id),
FOREIGN KEY(Module_Id) REFERENCES Module_info(Module_Id));
CREATE TABLE Associate_Feedback(
Associate_id Varchar(20),
Question_id Varchar(20),
Module_id Varchar(20),
associate_rating integer,
FOREIGN KEY(Associate_id) REFERENCES Associate_info(Associate_id),
FOREIGN KEY(Question_Id) REFERENCES Questions(Question_Id),
FOREIGN KEY(Module_Id) REFERENCES Module_info(Module_Id));
CREATE TABLE Login_details(
User_Id Varchar(20),
user_Password Varchar(20));
SHOW TABLES;

```

## **#EXERCISE 2:**

```

INSERT INTO Trainer_info VALUES ('F001','Mr.','PANKAJ GHOSH','Pune','Java','Bachelor of
Technology',12,'Pankaj.Ghosh@alliance.com','fac1@123'),
('F002','Mr.','SANJAY RADHAKRISHNAN ','Bangalore','DotNet','Bachelor of
Technology',12,'Sanjay.Radhakrishnan@alliance.com','fac2@123'),
('F003','Mr.','VIJAY MATHUR','Chennai','Mainframe','Bachelor of
Technology',10,'Vijay.Mathur@alliance.com','fac3@123'),
('F004','Mrs.','NANDINI NAIR','Kolkata','Java','Master of Computer
Applications',9,'Nandini.Nair@alliance.com','fac4@123'),
('F005','Miss.','ANITHA PAREKH','Hyderabad','Testing','Master of Computer
Applications',6,'Anitha.Parekh@alliance.com','fac5@123'),
('F006','Mr.','MANOJ AGRAWAL','Mumbai','Mainframe','Bachelor of
Technology',9,'Manoj.Agrawal@alliance.com','fac6@123'),
('F007','Ms.','MEENA KULKARNI','Coimbatore','Testing','Bachelor of
Technology',5,'Meena.Kulkarni@alliance.com','fac7@123'),
('F009','Mr.','SAGAR MENON','Mumbai','Java','Master of Science In Information
Technology',12,'Sagar.Menon@alliance.com','fac8@123');

```

```

SELECT * FROM Trainer_Info;

INSERT INTO Batch_Info VALUES ('B001','MRS.SWATI ROY','MSP'),
('B002','MRS.ARURNA K','HEALTHCARE'),
('B003','MR.RAJESH KRISHNAN','LIFE SCIENCES'),
('B004','MR.SACHIN SHETTY','BFS'),
('B005','MR.RAMESH PATEL','COMMUNICATION'),
('B006','MRS.SUSAN CHERIAN','RETAIL & HOSPITALITY'),
('B007','MRS.SAMPADA JAIN','MSP'),
('B008','MRS.KAVITA REGE','BPO'),
('B009','MR.RAVI SEJPAL','MSP');

SELECT *FROM Batch_Info;

INSERT INTO Module_Info VALUES ('O10SQL','Oracle 10g SQL',16),
('O10PLSQL','Oracle 10g PL/ SQL',16),
('J2SE','Core Java SE 1.6',288),
('J2EE','Advanced Java EE 1.6',80),
('JAVAFX','JavaFX 2.1',80),
('DOTNT4','Net Framework 4.0',50),
('SQL2008','MS SQI Server 2008',120),
('MSBI08','MS BI Studio 2008',158),
('SHRPNT','MS Share Point',80),
('ANDRD4','Android 4.0',200),
('EM001','Instructor',0),
('EM002','Course Material',0),
('EM003','Learning Effectiveness',0),
('EM004','Environment',0),
('EM005','Job Impact',0),
('TM001','Attendees',0),
('TM002','Course Material',0),
('TM003','Environment',0);

SELECT * FROM Module_Info;

INSERT INTO Associate_Info VALUES
('A001','Miss.','GAYATHRI NARAYANAN','Gurgaon','Java','Bachelor of
Technology','Gayathri.Narayanan@hp.com','tne1@123'),

('A002','Mrs.','RADHIKA MOHAN','Kerala','Java','Bachelor of Engineering In Information
Technology','Radhika.Mohan@cognizant.com','tne2@123'),

```

('A003','Mr.','KISHORE SRINIVAS','Chennai','Java','Bachelor of Engineering In Computers','Kishore.Srinivas@ibm.com','tne3@123'),

('A004','Mr.','ANAND RANGANATHAN','Mumbai','DotNet','Master of Computer Applications','Anand.Ranganathan@finolex.com','tne4@123'),

('A005','Miss.','LEELA MENON','Kerala','Mainframe','Bachelor of Engineering In Information Technology','LeeIa.Menon@microsoft.com','tne5@123'),

('A006','Mrs.','ARTI KRISHNAN','Pune','Testing','Master of Computer Applications','Arti.Krishnan@cognizant.com','tne6@123'),

('A007','Mr.','PRABHAKAR SHUNMUGHAM','Mumbai','Java','Bachelor of Technology','Prabhakar.Shunmugham@honda.com','tne7@123');

SELECT \*FROM ASSOCIATE\_INFO;

INSERT INTO Questions(QUESTION\_ID,Module\_id,Question\_text) VALUES

('Q001','EM001','Instructor knowledgeable and able to handle all your queries'),

('Q002','EM001','All the topics in a particular course handled by the trainer without any gaps or slippages'),

('Q003','EM002','The course materials presentation, hands on, etc. referred during the training are relevant and useful.'),

('Q004','EM002','The Hands on session adequate enough to grasp the understanding of the topic.'),

('Q005','EM002','The reference materials suggested for each module are adequate.'),

('Q006','EM003','Knowledge and skills presented in this training are applicable at your work'),

('Q007','EM003','This training increases my proficiency level'),

('Q008','EM004','The physical environment e.g. classroom space, air-conditioning was conducive to learning.'),

('Q009','EM004','The software/hardware environment provided was sufficient for the purpose of the training.'),

('Q010','EM005','This training will improve your job performance.'),

('Q011','EM005','This training align with the business priorities and goals.'),

('Q012','TM001','Participants were receptive and had attitude towards learning.'),

('Q013','TM001','All participants gained the knowledge and the practical skills after this training.'),

('Q014','TM002','The course materials presentation, hands on, etc. available for the session covers the entire objectives of the course.'),

('Q015','TM002','Complexity of the course is adequate for the participant level.'),

('Q016','TM002','Case study and practical demos helpful in understanding of the topic'),

('Q017','TM003','The physical environment e.g. classroom space, air-conditioning was conducive to learning.'),

('Q018','TM003','The software/hardware environment provided was adequate for the purpose of the training.');

SELECT \*FROM questions;

INSERT INTO Associate\_Status (Associate\_id,Module\_id,Batch\_id,Trainer\_id,Start\_date,End\_date)

VALUES ('A001','O10SQL','B001','F001','2000-12-15','2000-12-25'),

('A002','O10SQL','B001','F001','2000-12-15','2000-12-25'),

('A003','O10SQL','B001','F001','2000-12-15','2000-12-25'),

```

('A001','O10PLSQL','B002','F002','2001-2-1','2001-2-12'),
('A002','O10PLSQL','B002','F002','2001-2-1','2001-2-12'),
('A003','O10PLSQL','B002','F002','2001-2-1','2001-2-12'),
('A001','J2SE','B003','F003','2002-8-20','2002-10-25'),
('A002','J2SE','B003','F003','2002-8-20','2002-10-25'),
('A001','J2EE','B004','F004','2005-12-1','2005-12-25'),
('A002','J2EE','B004','F004','2005-12-1','2005-12-25'),
('A003','J2EE','B004','F004','2005-12-1','2005-12-25'),
('A004','J2EE','B004','F004','2005-12-1','2005-12-25'),
('A005','JAVAFX','B005','F006','2005-12-4','2005-12-20'),
('A006','JAVAFX','B005','F006','2005-12-4','2005-12-20'),
('A006','SQL2008','B006','F007','2007-6-21','2007-6-28'),
('A007','SQL2008','B006','F007','2007-6-21','2007-6-28'),
('A002','MSBI08','B007','F006','2009-6-26','2009-6-29'),
('A003','MSBI08','B007','F006','2009-6-26','2009-6-29'),
('A004','MSBI08','B007','F006','2009-6-26','2009-6-29'),
('A002','ANDRD4','B008','F005','2010-6-5','2010-6-28'),
('A005','ANDRD4','B008','F005','2010-6-5','2010-6-28'),
('A003','ANDRD4','B009','F005','2011-8-1','2011-8-20'),
('A006','ANDRD4','B009','F005','2011-8-1','2011-8-20');

```

### **#EXERCISE 3:**

```

UPDATE Trainer_Info SET Trainer_Password='nn4@123' WHERE Trainer_Password='fac4@123';

SELECT * FROM TRAINER_INFO;

SELECT * FROM Associate_Status;

```

### **#EXERCISE 4:**

```

DELETE FROM Associate_Status WHERE Associate_Id = 'A003' AND Module_Id = 'J2EE' AND Batch_Id =
'B004' AND Trainer_Id = 'F004' AND Start_Date = '2005-12-01' AND End_Date = '2005-12-25';

SELECT * FROM Associate_Status;

```

### **#EXERCISE 5:**

```

SELECT Trainer_Name, Trainer_Experience FROM Trainer_Info ORDER BY Trainer_Experience DESC
LIMIT 5;

```

### **#EXERCISE 6:**

```

START TRANSACTION;

INSERT INTO Login_details
Values ('U001','Admin1@123'), ('U002','Admin2@123');

```

```
SELECT *FROM Login_Details; #two rows got inserted

ROLLBACK;

SELECT *FROM Login_Details; #after performing rollback the rowsgot deleted.
```

```
CREATE USER 'Sharmita'@'host' IDENTIFIED BY 'password';
GRANT CREATE,SELECT ON assignment1 TO 'Sharmita'@'host';
START TRANSACTION;
INSERT INTO Login_details
Values ('U001','Admin1@ 123'), ('U002','Admin2@123');
SELECT *FROM Login_Details; #two rows got inserted
ROLLBACK;
SELECT *FROM Login_Details; #after performing rollback the rowsgot deleted.
REVOKE CREATE,SELECT ON assignment1 FROM 'Sharmita'@'host'
```

#### **#EXERCISE 8:**

```
DROP TABLE Login_Details;
```

#### **#EXERCISE 9:**

```
CREATE TABLE suppliers(supplier_id integer(10) NOT NULL PRIMARY KEY,
Supplier_Name varchar(50) NOT NULL,
address varchar(50),
city varchar(50),
state varchar(25),zip_code varchar(10));
```

#### **#EXERCISE10-11**

```
CREATE TABLE COURSE (Course_Code varchar(50) PRIMARY KEY, Base_fees Integer, Special_fees
Integer, Created_By varchar(100), Updated_By varchar(100));

CREATE TABLE course_fees( COURSE_CODE varchar(50) primary key, BASE_FEES Integer,
SPECIAL_FEES Integer, DISCOUNT Integer);

CREATE TABLE course_fees_history( COURSE_CODE varchar(50) primary key, BASE_FEES Integer,
SPECIAL_FEES Integer, CREATED_BY varchar(50), Updated_By varchar(50));

INSERT INTO course_fees VALUES
('1',180,100,10),
('2',150,110,10),
('3',160,170,5),
('4',150,100,10),
```

```

('6',190,100,40);

SELECT *FROM course_fees;

INSERT INTO course_fees_history VALUES

('1',120,123,'Ram','Ramesh'),

('2',150,110,'Bala','Ram'),

('3',160,170,'Bala','Vinu'),

('4',170,235,'Ram','Ram'),

('6',190,100,'Vinod','Vinod' );

SELECT *FROM course_fees_history;

SELECT DISTINCT(COURSE_CODE) FROM COURSE_FEES UNION SELECT
DISTINCT(COURSE_CODE) FROM COURSE_FEES_HISTORY;

SELECT COUNT(*) FROM course_fees_history;

SELECT count(DISTINCT(COURSE_CODE)) FROM COURSE_FEES_HISTORY;

SELECT count(DISTINCT(base_fees)) FROM COURSE_FEES_HISTORY;

SELECT count(DISTINCT(special_fees)) FROM COURSE_FEES_HISTORY;

SELECT COUNT(*) FROM course_fees;

SELECT count( DISTINCT(COURSE_CODE)) FROM COURSE_FEES;

SELECT count( DISTINCT(base_fees)) FROM COURSE_FEES;

SELECT count( DISTINCT(special_fees)) FROM COURSE_FEES;

```

#FUNCTIONS

## #EXERCISE12

```

CREATE TABLE course_info ( COURSE_CODE varchar(10) PRIMARY KEY, COURSE_NAME
varchar(20) NOT NULL, COURSE_DESCRIPTION varchar(25), COURSE_START_DATE Date ,
COURSE_DURATION int,

```

```

NO_OF_PARTICIPANTS int, COURSE_TYPE Char(3));

```

```

CREATE TABLE student_info( STUDENT_ID varchar(10) PRIMARY KEY,
FIRST_NAME varchar(20), LAST_NAME varchar(25), ADDRESS varchar(150));

```

```

alter table student_info

```

```

add course_code varchar(10);

```

```

Alter table student_info

```

```

add foreign key (course_code) references course_info(course_code);

```

```

SELECT *FROM course_fees;

```

```

INSERT INTO course_fees VALUES ('7',NULL,120,15);

```

```

INSERT INTO course_fees VALUES ('8',NULL,200,50);

```



```

INSERT INTO course_fees VALUES ('9',300,200,50),('10',175,250,45);
SELECT * FROM course_fees;
SELECT MIN(IFNULL(base_fees,0)) AS MIN_FEES,MAX(base_fees) AS MAX_FEES FROM course_fees;

```

#### **#EXERCISE13 -14**

```

SELECT *FROM course_info;
SELECT *from course_fees;
ALTER TABLE course_fees ADD infra_fees DECIMAL(5,3);
UPDATE course_fees SET infra_fees =45.751 WHERE course_code=1;
UPDATE course_fees SET infra_fees =43.453 WHERE course_code=2;
UPDATE course_fees SET infra_fees =46.751 WHERE course_code=3;
UPDATE course_fees SET infra_fees =47.751 WHERE course_code=4;
UPDATE course_fees SET infra_fees =25.453 WHERE course_code=5;
UPDATE course_fees SET infra_fees =42.751 WHERE course_code=6;
UPDATE course_fees SET infra_fees =72.751 WHERE course_code=7;
UPDATE course_fees SET infra_fees =95.751 WHERE course_code=8;
UPDATE course_fees SET infra_fees =44.751 WHERE course_code=9;
UPDATE course_fees SET infra_fees =37.791 WHERE course_code=10;
SELECT AVG(infra_fees) FROM course_fees INNER JOIN course_info ON
course_fees.course_code=course_info.course_code;
SELECT ROUND(AVG(infra_fees),3) FROM course_fees;
SELECT COURSE_NAME,DATEDIFF(curdate(),COURSE_START_DATE) AS no_of_days FROM
course_info;

```

#### **#EXERCISE15**

```

SELECT CONCAT('<',course_name,>','<',course_code,>') FROM course_info;

```

#### **#EXERCISE16**

```

SELECT AVG(IFNULL(BASE_FEES,0)) as avg_base_fees FROM COURSE_FEES;

```

#### **#EXERCISE17**

```

ALTER TABLE course_info ADD message VARCHAR(100);
INSERT INTO Course_info(Course_Code,Course_Name,Course_Type,message) VALUES
(1,'ECONOMICS','CLR','Class Room'),(2,'DATA','EL','ELearning'),(3,'FINANCE','OF','Offline Reading');
SELECT * FROM course_info;
SELECT Course_Type, message from Course_info where Course_Type in ('CLR','EL','OF');
UPDATE Course_info SET course_start_date='2021-06-03' where course_code=1;
UPDATE Course_info SET course_start_date='2021-04-06' where course_code=2;
UPDATE Course_info SET course_start_date='2021-06-04' where course_code=3;

```

UPDATE Course\_info SET no\_of\_participants=20 where course\_code=1;

UPDATE Course\_info SET no\_of\_participants=21 where course\_code=2;

UPDATE Course\_info SET no\_of\_participants=22 where course\_code=3;

#### **#EXERCISE18**

SELECT course\_start\_date, SUM(NO\_OF\_PARTICIPANTS) as NO\_OF\_PARTICIPANTS FROM  
course\_info

GROUP BY course\_start\_date;

#### **#EXERCISE19**

SELECT course\_start\_date, SUM(NO\_OF\_PARTICIPANTS) as NO\_OF\_STUDENTS FROM course\_info

WHERE COURSE\_TYPE="CLR" GROUP BY course\_start\_date;

#### **#EXERCISE20**

SELECT course\_start\_date, SUM(NO\_OF\_PARTICIPANTS) as NO\_OF\_STUDENTS FROM course\_info

WHERE COURSE\_TYPE="CLR"

GROUP BY course\_start\_date HAVING NO\_OF\_STUDENTS>10 ;

#### **#EXERCISE21**

SELECT course\_name,datediff(curdate(),course\_start\_date)as course\_duration FROM course\_info

ORDER BY course\_duration ASC;

#### **#EXERCISE22**

SELECT  
STUDENT\_INFO.STUDENT\_ID,STUDENT\_INFO.FIRST\_NAME,STUDENT\_INFO.LAST\_NAME,  
COURSE\_INFO.COURSE\_CODE

FROM Student\_info INNER JOIN course\_info ON student\_info.course\_code=course\_info.course\_code

where course\_info.course\_code=167;

#### **#EXERCISE23**

SELECT \*from course\_fees;

SELECT \*from course\_info;

SELECT course\_info.course\_description,course\_fees.discount

FROM COURSE\_INFO INNER JOIN course\_fees ON course\_info.course\_code=course\_fees.course\_code;

#### **#EXERCISE24**

select \*from student\_info;

INSERT INTO student\_info VALUES

(2211232,'Sharmita','Saha','23 Kolkata, India',1),

(2211424,'Soumee','Ghosh','12/24 Bangalore',2),

(2211436,'Ishita','Das','Delhi',3);

```

SELECT * FROM student_info;

SELECT student_info.First_name, course_info.course_code FROM student_info
LEFT JOIN course_info ON student_info.course_code=course_info.course_code;

SELECT student_info.First_name, course_info.course_code FROM student_info
RIGHT JOIN course_info ON student_info.course_code=course_info.course_code;

```

#### **#EXERCISE25**

```

SELECT *FROM COURSE_INFO;

INSERT INTO COURSE_INFO (COURSE_CODE,COURSE_NAME) VALUES (4,
'STATISTICS'),(5,'PYTHON');

SELECT *FROM COURSE_FEES;

INSERT INTO COURSE_INFO (COURSE_CODE,COURSE_NAME) VALUES (11, 'SQL'),(12,'VBA');

SELECT *FROM STUDENT_INFO;

INSERT INTO COURSE_FEES VALUES (11,500,200,10,45.791),(12,600,300,5,42.123);

INSERT INTO student_info VALUES(2211746,'Zhan','Fernandez','Mumbai',11),(2211639,'Riddhi',
'Pal','Kolkata',12);

SELECT DISTINCT student_info.student_id FROM student_info INNER JOIN course_fees
ON student_info.course_code=course_fees.course_code WHERE
((base_fees+special_fees+infra_fees)*(1-discount/100))<1500;

```

#### **#EXERCISE26**

```

SELECT DISTINCT student_info.student_id, Student_info.FIRST_NAME FROM student_info INNER JOIN
course_fees
ON student_info.course_code=course_fees.course_code WHERE
((base_fees+special_fees+infra_fees)*(1-discount/100))<1500;

```