

SQL HANDS ON EXERCISE

#EXERCISE_1

```
CREATE database SQL_hands_on_exercise;
```

```
SHOW databases;
```

```
USE SQL_hands_on_exercise;
```

```
CREATE TABLE Trainer_info (Trainer_id VARCHAR(20) UNIQUE, Salutation VARCHAR(7),  
Trainer_Name VARCHAR(30), Trainer_Location VARCHAR(30), Trainer_Track VARCHAR(15),  
Trainer_Qualification VARCHAR(100), Trainer_Experience INT(11), Trainer_Email VARCHAR(100) NOT  
NULL, Trainer_Password VARCHAR(20) NOT NULL);
```

```
CREATE TABLE Batch_info(Batch_id VARCHAR(20) PRIMARY KEY,  
Batch_owner VARCHAR(30),Batch_BU_Name VARCHAR(30));
```

```
CREATE TABLE Module_info(Module_id VARCHAR(20) PRIMARY KEY, Module_Name VARCHAR(40),  
Module_Duration INT(11));
```

```
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) NOT NULL,  
Associate_Password VARCHAR(20) NOT NULL);
```

```
CREATE TABLE Questions(Question_id VARCHAR(20) PRIMARY KEY, Module_id VARCHAR(20)  
References Module_info(Module_id) , Question_text VARCHAR(900));
```

```
CREATE TABLE Associate_status(Associate_id VARCHAR(20) REFERENCES  
Associate_info(Associate_id), Module_id VARCHAR(20) References Module_info(Module_id),  
Start_Date VARCHAR(20),  
End_Date VARCHAR(20), AFeedbackGiven VARCHAR(20), TFeedbackGiven VARCHAR(20));
```

```
CREATE TABLE Trainer_Feedback(Trainer_id VARCHAR(20) REFERENCES Trainer_info(Trainer_id),
Question_id VARCHAR(20) REFERENCES Questions(Question_id),
Batch_id VARCHAR(20) REFERENCES Batch_info(Batch_id), Module_id VARCHAR(20) References
Module_info(Module_id), Training_rating INT(11));
```

```
CREATE TABLE Associate_Feedback(Associate_id VARCHAR(20) REFERENCES
Associate_info(Associate_id),
Question_id VARCHAR(20) REFERENCES Questions(Question_id), Module_id VARCHAR(20)
References Module_info(Module_id), Associate_rating INT(11));
```

```
CREATE TABLE Login_Details(user_id VARCHAR(20) PRIMARY KEY, User_Password VARCHAR(20) NOT
NULL);
```

#Exercise_2

```
INSERT INTO
Trainer_info(Trainer_id,Salutation,Trainer_Name,Trainer_Location,Trainer_Track,Trainer_Qualificati
on,Trainer_Experiance,Trainer_Email,Trainer_Password)

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');
```

```
INSERT INTO Batch_info(Batch_id,Batch_Owner,Batch_BU_Name)
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','COMMUNICATIONS'),
('B006','RS.SUSAN CHERIAN','RETAIL & HOSPITALITY'),
('B007','MRS.SAMPADA JAIN','MSP'),
('B008','MRS.KAVITA REGE','BPO'),
('B009','MR.RAVI SEJPAL','MSP');
```

```
INSERT INTO Module_info(Module_id,Module_Name,Module_Duration) 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 SQL 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);
```

INSERT INTO

Associate_info(Associate_id,Salutation,Associate_Name,Associate_Location,Associate_Track,Associate_Qualification,Associate_Email,Associate_Password)

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','Leela.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');

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, handson, 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.');

```
ALTER TABLE Associate_status ADD COLUMN Batch_id VARCHAR(20) REFERENCES
Batch_info(Batch_id);
```

```
ALTER TABLE Associate_status ADD COLUMN Trainer_id VARCHAR(20) REFERENCES
Trainer_info(Trainer_id);
```

```
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','2002-12-25'),
('A002','J2EE','B004','F004','2005-12-1','2002-12-25'),
('A003','J2EE','B004','F004','2005-12-1','2002-12-25'),
('A004','J2EE','B004','F004','2005-12-1','2002-12-25'),
('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

```
select * from trainer_info;  
  
SET SQL_SAFE_UPDATES=0;  
  
UPDATE trainer_info set trainer_password= 'nn4@123'  
where trainer_password= 'fac4@123';
```

#Exercise-4

```
select * from associate_status;  
  
delete from associate_status where associate_id= 'A003';
```

#Exercise-5

```
select trainer_experience from trainer_info  
  
order by trainer_experience desc  
  
limit 5;
```

#Exercise_6

```
start transaction;  
  
select * from login_details;  
  
INSERT INTO login_details(user_id,user_password)  
values('U001','Admin1@123'), ('U002','Admin2@123');  
  
select * from login_details;  
  
rollback;
```

#Exercise 7

```
CREATE USER 'kaushik@gmail.com' IDENTIFIED by 'Password';  
  
GRANT CREATE ON SQL_hands_on_exercise TO 'kaushik@gmail.com';  
  
select * from login_details;  
  
start transaction;  
  
INSERT INTO login_details(user_id,user_password)  
values('U001','Admin1@123'), ('U002','Admin2@123');  
  
rollback;  
  
REVOKE CREATE ON SQL_hands_on_exercise from 'kaushik@gmail.com';
```

```
REVOKE SELECT ON SQL_hands_on_exercise.Login_details from 'kaushik@gmail.com';
```

#Exercise_8

```
DROP TABLE LOGIN_DETAILS;
```

```
SELECT * FROM LOGIN_Details;
```

#Exercise_9

```
CREATE TABLE supplier(supplier_id INT(10) NOT NULL,supplier_name VARCHAR(50) NOT  
NULL,address VARCHAR(50),city VARCHAR(50),state VARCHAR(25),zip_code VARCHAR(10));
```

#Exercise_10

```
create table course(course_code varchar(25), Base_fees int(11), special_fees int(11), created_by  
varchar(25), updated_by varchar(25));
```

```
show tables;
```

```
create table course_fees(course_code varchar(25), Base_fees int(11), special_fees int(11), Discount  
int(11));
```

```
create table course_fees_history(course_code varchar(25), Base_fees int(11), special_fees int(11),  
created_by varchar(25), updated_by varchar(25));
```

```
INSERT INTO course_fees(course_code, Base_fees, special_fees, Discount)
```

```
values (1,180,100,10), (2,150,110,10), (3,160,170,5), (4,150,100,10), (6,190,100,40);
```

```
INSERT INTO course_fees_history(course_code, Base_fees , special_fees , created_by, updated_by)
```

```
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 course_code, Base_fees, special_fees from course_fees union select course_code, Base_fees,  
special_fees from course_fees_history;
```

```
select * from course_fees;
```


#Exercise_11

```
SELECT COUNT(*) from course_fees;
```

```
SELECT COUNT(DISTINCT(course_code)) from course_fees;
```

#since both count is equal course_code is unique in the table course_fees

```
SELECT COUNT(DISTINCT(base_fees)) from course_fees;
```

#since both count is not equal base_fees is not unique in the table course_fees

```
SELECT COUNT(DISTINCT(special_fees)) from course_fees;
```

#since both count is not equal base_fees is not unique in the table course_fees

```
SELECT COUNT(*) from course_fees_history;
```

```
SELECT COUNT(DISTINCT(course_code)) from course_fees_history;
```

#since both count is equal course_code is unique in the table course_fees_history

```
SELECT COUNT(DISTINCT(base_fees)) from course_fees_history;
```

#since both count is equal base_fees is unique in the table course_fees_history

```
SELECT COUNT(DISTINCT(special_fees)) from course_fees_history;
```

#since both count is equal base_fees is unique in the table course_fees_history

#FUNCTIONS:

#Exercise_12

```
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(11), No_of_participants int(255), 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));
```

```
INSERT INTO course_fees(course_code, Base_fees, special_fees, Discount) values (15,null,130,23), (16,null,140,16);
```

```
INSERT INTO course_fees(course_code, Base_fees, special_fees, Discount) values (17,300,150,33), (18,175,160,36);
```

```
select * from course_fees;
```

```
select MIN(IFNULL(base_fees,0)), MAX(base_fees) from course_fees;
```

#Exercise 14

```
select count(*) from course_fees;

select * from course_fees;

CREATE TABLE INFRA( sn INT PRIMARY KEY, infra_fees DECIMAL(5,3));

ALTER TABLE course_fees ADD infra_fees DECIMAL(5,3);

INSERT INTO INFRA VALUES(1,45.751), (2,43.453),(3,41.546),(4,46.248),(5,47.164), (6,44.436),
(7,48.346), (8,49.463),(9,42.431);

SELECT datediff(curdate(), course_start_date) from course_info;
```

#Exercise_13

```
select avg(INFRA.infra_fees) from course_fees INNER JOIN INFRA ON course_fees.course_code=
INFRA.sn;

select ROUND(avg(INFRA.infra_fees),2) from course_fees INNER JOIN INFRA ON
course_fees.course_code= INFRA.sn;
```

#EXERCISE_15

```
select concat('<',course_name,'>','<',course_code,'>') from course_info;
```

#Exercise_16

```
select AVG(ifnull(base_fees,0)) from course_fees;

select round(AVG(ifnull(base_fees,0)),2) from course_fees;
```

#Exercise_17

```
ALTER TABLE course_info

add column Message VARCHAR(22);

select * from course_info;

INSERT INTO course_info(course_code,course_name,course_type,message)

values(1,'SQL','CLR','Class Room'), (2,'JAVA','EL','ELearning'), (3,'PYTHON','OF','Offline Reading');

select * from course_info;

SELECT course_type,message from course_info;
```

```
select * from course_info;
```

#Exercise 18

```
update course_info set course_start_date='2021-01-01' where course_code=1;
```

```
update course_info set course_start_date='2021-02-01' where course_code=2;
```

```
update course_info set course_start_date='2021-03-01' where course_code=3;
```

```
update course_info set No_of_participants=25 where course_code=1;
```

```
update course_info set No_of_participants=30 where course_code=2;
```

```
update course_info set No_of_participants=40 where course_code=3;
```

```
select course_start_date, sum(no_of_participants) AS No_of_participants from course_info  
group by course_start_date;
```

#Exercise 19

```
select course_start_date, no_of_participants from course_info
```

```
where course_type='CLR'
```

```
group by course_start_date;
```

#Exercise 20

```
select course_start_date, no_of_participants as no_of_students from course_info
```

```
where course_type='CLR'
```

```
group by course_start_date having no_of_participants>10;
```

#Exercise 21

```
SELECT * FROM COURSE_INFO;
```

```
select course_name, datediff(curdate(),course_start_date) as course_duration from course_info  
order by course_duration ASC;
```

#Exercise 22

```
alter table student_info  
add column course_code varchar(10);  
  
select * from student_info;  
  
select student_id,first_name,last_name,student_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;
```

#Exercise 23

```
select * from course_fees;  
  
select * from course_info;  
  
select course_fees.discount, course_info.course_description from course_fees inner join  
course_info  
on course_fees.course_code=course_info.course_code;
```

#Exercise 24

```
select * from student_info;  
  
INSERT INTO student_info values(123456,'kaushik','kar','36 kolkata',1),  
(234567,'shrija','dey','123 kolkata',2),  
(345678,'surojit','sen','29 bangalore',3);  
  
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;
```

#Exercise_25

```
select * from course_info;
```

```
INSERT INTO COURSE_INFO (COURSE_CODE,COURSE_NAME) VALUES (4, 'c  
programming'),(5,'tableau'),(6, 'Machine learnuing'),(7,'VBA');
```

```
SELECT *FROM COURSE_FEES;
```

```
SELECT *FROM STUDENT_INFO;
```

```
INSERT INTO COURSE_FEES VALUES (6,500,200,10,45.791),(7,600,300,5,42.123);
```

```
INSERT INTO student_info VALUES(2211746,'rishabh','sen','Mumbai',6),(2211639,'avik',  
'das','Kolkata',7);
```

```
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;
```

#exercise_26

```
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;
```