

STRUCTURED QUERY LANGUAGE

HANDS - ON SOLUTIONS

LAVANYA M (2211574)

CDB22DW022

PROGRAMMER ANALYST

```
SHOW DATABASES;  
CREATE DATABASE handsonsql;  
USE handsonsql;  
SHOW Tables;
```

EXERCISE 1:

```
CREATE TABLE Trainer_info(  
Trainer_id CHAR(20) PRIMARY KEY,  
Salutation CHAR(7),  
Trainer_Name CHAR(30),  
Trainer_Location CHAR(30),  
Trainer_Track CHAR(15),  
Trainer_Qualification CHAR(100),  
Trainer_Experience INTEGER,  
Trainer_Email CHAR (100),  
Trainer_Password CHAR(20));
```

```
CREATE TABLE Batch_info(  
Batch_Id CHAR(20) PRIMARY KEY,  
Batch_Owner CHAR(30),  
Batch_BU_Name CHAR(30));
```

```
CREATE TABLE MOdule_Info(  
MOdule_Id CHAR(20) PRIMARY KEY,  
Module_Name CHAR(40),  
Module_Duration INTEGER);
```

```
CREATE TABLE Associate_Info(  
Associate_Id CHAR(20) PRIMARY KEY ,
```

Salutation CHAR(7),
Associate_Name CHAR(30),
Associate_Location CHAR(30),
Associate_Track CHAR(15),
Associate_Qualification CHAR(200),
Associate_Email CHAR(100),
Associate_Password CHAR(20));

CREATE TABLE Questions(
Question_Id CHAR(20) PRIMARY KEY,
Module_Id CHAR(20),
Question_Text CHAR(200));

CREATE TABLE Associate_Status(
Associate_Id CHAR(20),
Module_Id CHAR(20),
Batch_Id CHAR(20),
Trainer_Id CHAR(20),
Start_Date CHAR(20),
End_Date CHAR(20),
FOREIGN KEY(Associate_Id) REFERENCES Associate_Info(Associate_Id),
FOREIGN KEY(Module_Id) REFERENCES Module_Info(Module_Id),
FOREIGN KEY(Batch_Id) REFERENCES Batch_Info(Batch_Id),
FOREIGN KEY(Trainer_Id) REFERENCES Trainer_Info(Trainer_Id));

CREATE TABLE Trainer_Feedback(
Trainer_Id CHAR(20),
Question_Id CHAR(20),
Batch_Id CHAR(20),
Module_Id CHAR(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 CHAR(20),
Question_Id CHAR(20),
Module_Id CHAR(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 CHAR(20) NOT NULL,
User_Password CHAR(20) NOT NULL);

EXERCISE 2:

INSERT INTO Trainer_Info (Trainer_Id,Salutation,Trainer_Name,
Trainer_Location,Trainer_Track,Trainer_Qualification,
Trainer_Experience,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' );
select * from Trainer_Info;

```

```

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','MRS.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, handson, 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.');

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_Id='F004';

EXERCISE 4:

```
SELECT * FROM Associate_Status;  
DELETE FROM Associate_Status WHERE  
Associate_id='A003' AND  
Module_Id='J2EE' AND  
Batch_Id='B004'AND  
Trainer_Id= 'F004' AND  
Start_Date= '2005-12-1' AND  
End_Date='2005-12-25';
```

EXERCISE 5:

```
SELECT * FROM Trainer_Info;  
SELECT * FROM Trainer_Info ORDER BY Trainer_Experience DESC LIMIT 5;
```

EXERCISE 6:

```
SELECT * FROM Login_Details;  
START TRANSACTION;  
INSERT INTO Login_Details VALUES  
( 'U001', 'Admin1@123'),  
( 'U002', 'Admin2@123');  
ROLLBACK;  
SELECT * FROM Login_Details;
```

EXERCISE 7:

```
CREATE USER 'Lavanya'@'Local' IDENTIFIED BY 'Password';  
GRANT CREATE ON handsonsql TO 'Lavanya'@'Local';
```

```
GRANT SELECT ON handsonsql.Login_Details TO 'Lavanya'@'Local';  
START TRANSACTION;  
SELECT * FROM Login_Details;  
INSERT INTO Login_Details VALUES  
( 'U001', 'Admin1 @123'),  
( 'U002', 'Admin2@123');  
ROLLBACK;  
REVOKE CREATE ON handsonsql FROM 'Lavanya'@'Local';  
REVOKE SELECT ON handsonsql.Login_Details FROM 'Lavanya'@'Local';
```

EXERCISE 8:

```
DROP TABLE Login_Details;  
SHOW TABLES;
```

EXERCISE 9:

```
CREATE TABLE Supplies(  
Supplier_Id INTEGER 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_Fees(  
COURSE_CODE VARCHAR(25) PRIMARY KEY,  
BASE_FEES INTEGER,
```

```
SPECIAL_FEES INTEGER,  
DISCOUNT INTEGER);
```

```
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);
```

```
CREATE TABLE Course_Fees_History(  
COURSE_CODE VARCHAR(25) PRIMARY KEY,  
BASE_FEES INTEGER,  
SPECIAL_FEES INTEGER,  
CREATED_BY VARCHAR(25),  
UPDATED_BY VARCHAR(25));
```

```
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 DISTINCT COURSE_CODE,BASE_FEES,SPECIAL_FEES FROM Course_Fees  
UNION
```

```
SELECT DISTINCT COURSE_CODE,BASE_FEES,SPECIAL_FEES FROM  
Course_Fees_History  
ORDER BY COURSE_CODE;
```

EXERCISE 11:

```
SELECT COUNT(*) FROM Course_Fees;
```

```
SELECT COUNT( DISTINCT (COURSE_CODE)) FROM Course_Fees;
```

```
/*yes, all are unique*/
```

```
SELECT COUNT( DISTINCT (BASE_FEES)) FROM Course_Fees;
```

```
/*no, some are not unique*/
```

```
SELECT COUNT( DISTINCT (SPECIAL_FEES)) FROM Course_Fees;
```

```
/*no, some are not unique*/
```

```
SELECT COUNT(*) FROM Course_Fees_History;
```

```
SELECT COUNT( DISTINCT (COURSE_CODE)) FROM Course_Fees_History;
```

```
/*yes, all are unique*/
```

```
SELECT COUNT( DISTINCT (BASE_FEES)) FROM Course_Fees_History;
```

```
/*yes, all are unique*/
```

```
SELECT COUNT( DISTINCT (SPECIAL_FEES)) FROM Course_Fees_History;
```

```
/*yes, all are unique*/
```

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 INTEGER,  
  NO_OF_PARTICIPANTS INTEGER,  
  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 VALUES
```

```
('7', NULL, 850,17),
```

```
('8', NULL, 150,12),
```

```
('9', 300, 800,14),
```

```
('10', 175, 150,15);
```

```
SELECT MAX(BASE_FEES), MIN(BASE_FEES) FROM COURSE_FEES;
```

EXERCISE 13:

```
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 = 41.234 WHERE Course_Code = '3';
```

```
UPDATE Course_Fees SET Infra_Fees = 47.493 WHERE Course_Code = '4';
```

```
UPDATE Course_Fees SET Infra_Fees = 48.051 WHERE Course_Code = '6';
```

```
UPDATE Course_Fees SET Infra_Fees = 42.481 WHERE Course_Code = '7';
```

```
UPDATE Course_Fees SET Infra_Fees = 46.021 WHERE Course_Code = '8';
```

```
UPDATE Course_Fees SET Infra_Fees = 40.373 WHERE Course_Code = '9';
```

```
UPDATE Course_Fees SET Infra_Fees = 49.761 WHERE Course_Code = '10';
```

```
SELECT ROUND(AVG(Infra_Fees),4) FROM Course_Fees;
```

EXERCISE 14:

```
SELECT Course_Name, datediff(CURRENT_DATE,Course_Start_DATE) AS  
number_of_days FROM Course_info;
```

EXERCISE 15:

```
SELECT CONCAT('<',Course_Name,'>','<',Course_Code,'>') As New_Col FROM  
Course_Info;
```

EXERCISE 16:

```
SELECT NULLIF(AVG(base_fees),0)FROM course_fees;
```

EXERCISE 17:

```
ALTER TABLE Course_Info  
ADD Message VARCHAR(50);  
INSERT INTO Course_Info(Course_code,Course_Name,Course_Type,Message) VALUES  
( 'A12','Java','CLR','"Class Room"'),  
( 'B11','DotNet','EL','"ELearning"'),  
( 'C21','Data','OF','"Offline Reading"');  
SELECT Course_Type,Message FROM Course_Info;
```

EXERCISE 18:

```
SELECT * FROM Course_Info;  
ALTER TABLE Course_Info  
ADD COLUMN Date_of_Enrollment DATE;  
SELECT Course_Start_Date, SUM(No_Of_Participants)As Tot_No_Of_Students FROM  
Course_Info
```

WHERE Date_Of_Enrollment = '2021-08-24'

GROUP BY Course_Start_Date;

EXERCISE 19:

SELECT Course_Start_Date, SUM(No_Of_Participants)As Tot_No_Of_Students FROM
Course_Info

WHERE Course_Type = 'CLR'

GROUP BY Course_Start_Date;

EXERCISE 20:

SELECT Course_Start_Date, SUM(No_Of_Participants)As Tot_No_Of_Students FROM
Course_Info

WHERE Course_Type = 'CLR'

GROUP BY Course_Start_Date

HAVING SUM(No_Of_Participants)>10;

EXERCISE 21:

SELECT * FROM Course_Info

ORDER BY Course_Duration ASC;

EXERCISE 22:

ALTER TABLE Student_INFO

ADD COLUMN Course_Code VARCHAR(10) REFERENCES Course_Info(Course_Code);

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';

EXERCISE 23:

SELECT

Course_Info.Course_Name, Course_Info.Course_Description, Course_Fees.Discount

FROM Course_Fees

INNER JOIN Course_Info

ON Course_Fees.Course_Code = Course_Info.Course_Code;

EXERCISE 24:

INSERT INTO Student_Info

VALUES

('C001','Lavanya','Muthuraman','Taramani,Chennai','A12'),

('C002','Ajith','Natraj','Agra,Delhi','B11'),

('C003','Maaya','Den','GST road,Rio','C21');

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:

INSERT INTO Course_Info(Course_code,Course_Name,No_Of_Participants,Course_Type)

VALUES

('A1','SQL',5,'SBA'),

('A2','PYTHON',3,'KBA');


```
INSERT INTO Course_Fees(Course_Code,Base_Fees,Special_Fees,Discount,Infra_Fees)
VALUES
('A1',150,140,18,43.643),
('A2',135,155,14,47.980);
```

```
INSERT INTO Student_Info(Student_Id,First_Name,Last_Name,Course_Code)
VALUES
('C004','Dhoni','John','A1'),
('C005','Ria','Ram','A2'),
('C006','Suresh','Raina','A1'),
('C007','Maaya','Rogesh','A2'),
('C008','Maroso','Kumar','A2');
```

```
SELECT 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 Student_Info.Student_Id,Student_Info.First_Name,Student_Info.Last_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;
```
