**Group-6**

|  |
| --- |
| **Members:-** |
| **1.Hamza Ahmad (Leader)( FA18-BCS-084)** |
| **2.Hanzala Shahid (FA18-BCS-014)** |
| **3.Usama Fareed (FA18-BCS-026)** |
| **4.Sharjeel Khan Niazi (FA18-BCS-019)** |
| **5.Hamza Aslam (SP17-BCS-050)** |
| **6.Abdullah Noor Niazi (FA18-BCS-004)** |
| **7.Usman Jadoon (FA18-BCS-100)** |

Contents

[Lab 6 1](#_Toc59726613)

[Common solution 1](#_Toc59726614)

[Individual Solutions:- 5](#_Toc59726615)

[Hanzala Shahid 5](#_Toc59726616)

[Hamza Bin Ahmed 10](#_Toc59726617)

[Muhammad Sharjeel khan 14](#_Toc59726618)

[Hamza Aslam 21](#_Toc59726619)

[Abdullah Noor Niazi 31](#_Toc59726620)

[Usman Jadoon 40](#_Toc59726621)

[Usama Fareed 45](#_Toc59726622)

[Lab 7 51](#_Toc59726623)

[Common Solution 51](#_Toc59726624)

[Individual Solutions:- 52](#_Toc59726625)

[Hanzala Shahid 52](#_Toc59726626)

[Hamza Bin Ahmed 53](#_Toc59726627)

[Muhammad Sharjeel khan 54](#_Toc59726628)

[Hamza Aslam 54](#_Toc59726629)

[Abdullah Noor Niazi 56](#_Toc59726630)

[Usman Jadoon 57](#_Toc59726631)

[Usama Fareed 58](#_Toc59726632)

[Lab 8 59](#_Toc59726633)

[Common Solution 59](#_Toc59726634)

[Individual Solutions:- 61](#_Toc59726635)

[Hanzala Shahid 61](#_Toc59726636)

[Hamza Bin Ahmed 62](#_Toc59726637)

[Muhammad Sharjeel khan 64](#_Toc59726638)

[Hamza Aslam 65](#_Toc59726639)

[Abdullah Noor Niazi 67](#_Toc59726640)

[Usman Jadoon 69](#_Toc59726641)

[Usama Fareed 70](#_Toc59726642)

[Lab 9 72](#_Toc59726643)

[Common Solution 72](#_Toc59726644)

[Individual Solutions:- 73](#_Toc59726645)

[Hanzala Shahid 73](#_Toc59726646)

[Hamza Bin Ahmed 74](#_Toc59726647)

[Muhammad Sharjeel khan 74](#_Toc59726648)

[Hamza Aslam 75](#_Toc59726649)

[Abdullah Noor Niazi 76](#_Toc59726650)

[Usman Jadoon 77](#_Toc59726651)

[Usama Fareed 78](#_Toc59726652)

[Lab 10 78](#_Toc59726653)

[Common Solution 78](#_Toc59726654)

[Individual Solutions:- 84](#_Toc59726655)

[Hanzala Shahid 84](#_Toc59726656)

[Hamza Bin Ahmed 90](#_Toc59726657)

[Muhammad Sharjeel khan 96](#_Toc59726658)

[Hamza Aslam 102](#_Toc59726659)

[Abdullah Noor Niazi 108](#_Toc59726660)

[Usman Jadoon 115](#_Toc59726661)

[Usama Fareed 121](#_Toc59726662)

[Lab 11 127](#_Toc59726663)

[Common Solution 127](#_Toc59726664)

[Individual Solutions:- 127](#_Toc59726665)

[Hanzala Shahid 127](#_Toc59726666)

[Hamza Bin Ahmed 128](#_Toc59726667)

[Muhammad Sharjeel khan 128](#_Toc59726668)

[Hamza Aslam 128](#_Toc59726669)

[Abdullah Noor Niazi 129](#_Toc59726670)

[Usman Jadoon 129](#_Toc59726671)

[Usama Fareed 130](#_Toc59726672)

[Lab 12 130](#_Toc59726673)

[Common Solution 130](#_Toc59726674)

[Individual Solutions:- 131](#_Toc59726675)

[Hanzala Shahid 131](#_Toc59726676)

[Hamza Bin Ahmed 132](#_Toc59726677)

[Muhammad Sharjeel khan 132](#_Toc59726678)

[Hamza Aslam 133](#_Toc59726679)

[Abdullah Noor Niazi 134](#_Toc59726680)

[Usman Jadoon 135](#_Toc59726681)

[Usama Fareed 136](#_Toc59726682)

[Lab 13 137](#_Toc59726683)

[Common Solution 137](#_Toc59726684)

[Individual Solutions:- 137](#_Toc59726685)

[Hanzala Shahid 137](#_Toc59726686)

[Hamza Bin Ahmed 137](#_Toc59726687)

[Muhammad Sharjeel khan 138](#_Toc59726688)

[Hamza Aslam 138](#_Toc59726689)

[Abdullah Noor Niazi 139](#_Toc59726690)

[Usman Jadoon 139](#_Toc59726691)

[Usama Fareed 139](#_Toc59726692)

[Lab 14 140](#_Toc59726693)

[Common Solution 140](#_Toc59726694)

[Individual Solutions:- 140](#_Toc59726695)

[Hanzala Shahid 140](#_Toc59726696)

[Hamza Bin Ahmed 141](#_Toc59726697)

[Muhammad Sharjeel khan 141](#_Toc59726698)

[Hamza Aslam 142](#_Toc59726699)

[Abdullah Noor Niazi 142](#_Toc59726700)

[Usman Jadoon 143](#_Toc59726701)

[Usama Fareed 144](#_Toc59726702)

[Lab 15 144](#_Toc59726703)

[Common Solution 144](#_Toc59726704)

[Individual Solutions:- 145](#_Toc59726705)

[Hanzala Shahid 145](#_Toc59726706)

[Hamza Bin Ahmed 146](#_Toc59726707)

[Muhammad Sharjeel khan 147](#_Toc59726708)

[Hamza Aslam 148](#_Toc59726709)

[Abdullah Noor Niazi 150](#_Toc59726710)

[Usman Jadoon 151](#_Toc59726711)

[Usama Fareed 152](#_Toc59726712)

[Lab 16 153](#_Toc59726713)

[Common Solution 153](#_Toc59726714)

[Individual Solutions:- 155](#_Toc59726715)

[Hanzala Shahid 155](#_Toc59726716)

[Hamza Bin Ahmed 157](#_Toc59726717)

[Muhammad Sharjeel khan 158](#_Toc59726718)

[Hamza Aslam 159](#_Toc59726719)

[Abdullah Noor Niazi 160](#_Toc59726720)

[Usman Jadoon 162](#_Toc59726721)

[Usama Fareed 164](#_Toc59726722)

[Lab 20 166](#_Toc59726723)

[Common Solution 166](#_Toc59726724)

[Individual Solutions:- 167](#_Toc59726725)

[Hanzala Shahid 168](#_Toc59726726)

[Hamza Bin Ahmed 169](#_Toc59726727)

[Muhammad Sharjeel khan 170](#_Toc59726728)

[Hamza Aslam 171](#_Toc59726729)

[Abdullah Noor Niazi 173](#_Toc59726730)

[Usman Jadoon 174](#_Toc59726731)

[Usama Fareed 175](#_Toc59726732)

# Lab 6

## Common solution

**Qno1:-**

**Answer:-**

create database DreamHome;

USE DreamHome;

create table Branch

(

branchNo varchar(20) NOT NULL Primary Key,

street varchar(100) NOT NULL,

city varchar(50) NOT NULL,

postcode varchar(20) NOT NULL

);

create table Staff

(

staffNo varchar(20) NOT NULL PRIMARY KEY,

fName varchar(50) NOT NULL,

lName varchar(50) NOT NULL,

position varchar(50) NOT NULL,

sex varchar(1) NOT NULL, DOB DateTime NOT NULL,

salary DECIMAL NOT NULL,

branchNo varchar(20) NOT NULL References Branch(branchNo)

);

create table Client

(

clientNo varchar(20) NOT NULL PRIMARY KEY,

fName varchar(50) NOT NULL,

lName varchar(50) NOT NULL,

telNo varchar(20) NOT NULL,

prefType varchar(50) NOT NULL,

maxRent DECIMAL NOT NULL

);

create table PrivateOwner

(

ownerNo varchar(20) NOT NULL PRIMARY KEY,

fName varchar(50) NOT NULL,

lName varchar(50) NOT NULL,

address varchar(50) NOT NULL,

telNo varchar(20) NOT NULL

);

create table PropertyForRent

(

propertyNo varchar(20) NOT NULL PRIMARY KEY,

street varchar(100) NOT NULL,

city varchar(50) NOT NULL,

postcode varchar(20) NOT NULL,

type varchar(10) NOT NULL,

rooms int NOT NULL,

rent DECIMAL NOT NULL,

ownerNo varchar(20) References PrivateOwner(ownerNo),

staffNo varchar(20) NOT NULL References Staff(staffNo),

branchNo varchar(20) NOT NULL References Branch(branchNo)

);

create table Viewing

(

clientNo varchar(20) NOT NULL References Client(clientNo),

propertyNo varchar(20) NOT NULL References PropertyForRent(propertyNo),

viewDate DateTime NOT NULL,

comment varchar(200) NOT NULL

);

create table Registration

(

clientNo varchar(20) NOT NULL References Client(clientNo),

branchNo varchar(20) NOT NULL References Branch(branchNo),

staffNo varchar(20) NOT NULL References Staff(staffNo),

dateJoined DateTime NOT NULL

);

INSERT into Branch

(

branchNo,

street,

city,

postcode

)

VALUES

(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),

(N'B002',N'H#78 Supply', N'ABT', N'53000'),

(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),

(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff

(

staffNo,

fName,

lName,

position,

sex,

DOB,

salary,

branchNo

)

VALUES

(

N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS DateTime),

CAST(9000 AS Decimal(18, 0)),N'B002'

),

(N'SG14', N'David', N'Ford', N'Supervisor', N'M',

CAST(0x0000531200000000 AS DateTime), CAST(18000 AS Decimal(18,0)),

N'B003'),

(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',

CAST(0x000056D400000000 AS DateTime), CAST(12000 AS Decimal(18,0)),

N'B003'),

(N'SG5',N'Susan', N'Brand', N'Manager', N'F',

CAST(0x0000C85800000000 AS DateTime), CAST(24000 AS Decimal(18,0)),

N'B003'),

(N'SL21', N'John', N'White', N'Manager', N'M',

CAST(0x0000CFF200000000 AS DateTime), CAST(30000 AS Decimal(18,0)),

N'B004'),

(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',

CAST(0x00005D6000000000 AS DateTime), CAST(9000 AS Decimal(18, 0)),

N'B002');

insert into Client

(

clientNo,

fName,

lName,

telNo,

prefType,

maxRent

)

values

(

'B1001','Mahad','Ali','030078601','yes',1000.0

),

(

'B1002','Sharjeel','Khan','030054621','yes',2000.0

),

(

'B1003','Hanzala','Shahid','030456601','no',1500.0

),

(

'B1004','Hamza','Aslam','0306446641','yes',8800.0

),

(

'B1005','Hamza','Ahmad','0354654401','noo',800.0

),

(

'B1006','Usama','Fareed','030074541','yes',4000.0

);

insert into PrivateOwner

(

ownerNo,

fName,

lName,

[address],

telNo

)

values

(

'B1','Azid','Ali','F18-4A','0354654264'

),

(

'B2','Mahad','Ali','F17-4A','0345154264'

),

(

'B3','Sharjeel','Khan','F14-7A','0352354264'

),

(

'B4','Hanzala','Shahid','F88-4A','0359354264'

),

(

'B5','Hamza','Aslam','F11-3A','0351694264'

),

(

'B6','Hamza','Ahmad','F19-5A','0354654264'

);

insert into PropertyForRent

(

propertyNo,street,city,postcode,[type],rooms,rent,ownerNo,staffNo,branchNo

)

values

(

'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'

),

(

'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'

),

(

'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'

);

insert into Viewing

(

clientNo,

propertyNo,

viewDate,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

insert into Viewing

(

clientNo,

propertyNo,

viewDate,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

**End of qno1**

**Qno:-2**

**Answer:-**

select \* from Branch update Branch set city='ABT' where city='ISB';

## Individual Solutions:-

### Hanzala Shahid

**FA18-BCS-014**

**Lab 6**

**Qno1:-**

**Answer:-**

create database DreamHome;

USE DreamHome;

create table Branch

(

branchNo varchar(20) NOT NULL Primary Key,

street varchar(100) NOT NULL,

city varchar(50) NOT NULL,

postcode varchar(20) NOT NULL

);

create table Staff

(

staffNo varchar(20) NOT NULL PRIMARY KEY,

fName varchar(50) NOT NULL,

lName varchar(50) NOT NULL,

position varchar(50) NOT NULL,

sex varchar(1) NOT NULL, DOB DateTime NOT NULL,

salary DECIMAL NOT NULL,

branchNo varchar(20) NOT NULL References Branch(branchNo)

);

create table Client

(

clientNo varchar(20) NOT NULL PRIMARY KEY,

fName varchar(50) NOT NULL,

lName varchar(50) NOT NULL,

telNo varchar(20) NOT NULL,

prefType varchar(50) NOT NULL,

maxRent DECIMAL NOT NULL

);

create table PrivateOwner

(

ownerNo varchar(20) NOT NULL PRIMARY KEY,

fName varchar(50) NOT NULL,

lName varchar(50) NOT NULL,

address varchar(50) NOT NULL,

telNo varchar(20) NOT NULL

);

create table PropertyForRent

(

propertyNo varchar(20) NOT NULL PRIMARY KEY,

street varchar(100) NOT NULL,

city varchar(50) NOT NULL,

postcode varchar(20) NOT NULL,

type varchar(10) NOT NULL,

rooms int NOT NULL,

rent DECIMAL NOT NULL,

ownerNo varchar(20) References PrivateOwner(ownerNo),

staffNo varchar(20) NOT NULL References Staff(staffNo),

branchNo varchar(20) NOT NULL References Branch(branchNo)

);

create table Viewing

(

clientNo varchar(20) NOT NULL References Client(clientNo),

propertyNo varchar(20) NOT NULL References PropertyForRent(propertyNo),

viewDate DateTime NOT NULL,

comment varchar(200) NOT NULL

);

create table Registration

(

clientNo varchar(20) NOT NULL References Client(clientNo),

branchNo varchar(20) NOT NULL References Branch(branchNo),

staffNo varchar(20) NOT NULL References Staff(staffNo),

dateJoined DateTime NOT NULL

);

INSERT into Branch

(

branchNo,

street,

city,

postcode

)

VALUES

(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),

(N'B002',N'H#78 Supply', N'ABT', N'53000'),

(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),

(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff

(

staffNo,

fName,

lName,

position,

sex,

DOB,

salary,

branchNo

)

VALUES

(

N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS DateTime),

CAST(9000 AS Decimal(18, 0)),N'B002'

),

(N'SG14', N'David', N'Ford', N'Supervisor', N'M',

CAST(0x0000531200000000 AS DateTime), CAST(18000 AS Decimal(18,0)),

N'B003'),

(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',

CAST(0x000056D400000000 AS DateTime), CAST(12000 AS Decimal(18,0)),

N'B003'),

(N'SG5',N'Susan', N'Brand', N'Manager', N'F',

CAST(0x0000C85800000000 AS DateTime), CAST(24000 AS Decimal(18,0)),

N'B003'),

(N'SL21', N'John', N'White', N'Manager', N'M',

CAST(0x0000CFF200000000 AS DateTime), CAST(30000 AS Decimal(18,0)),

N'B004'),

(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',

CAST(0x00005D6000000000 AS DateTime), CAST(9000 AS Decimal(18, 0)),

N'B002');

insert into Client

(

clientNo,

fName,

lName,

telNo,

prefType,

maxRent

)

values

(

'B1001','Mahad','Ali','030078601','yes',1000.0

),

(

'B1002','Sharjeel','Khan','030054621','yes',2000.0

),

(

'B1003','Hanzala','Shahid','030456601','no',1500.0

),

(

'B1004','Hamza','Aslam','0306446641','yes',8800.0

),

(

'B1005','Hamza','Ahmad','0354654401','noo',800.0

),

(

'B1006','Usama','Fareed','030074541','yes',4000.0

);

insert into PrivateOwner

(

ownerNo,

fName,

lName,

[address],

telNo

)

values

(

'B1','Azid','Ali','F18-4A','0354654264'

),

(

'B2','Mahad','Ali','F17-4A','0345154264'

),

(

'B3','Sharjeel','Khan','F14-7A','0352354264'

),

(

'B4','Hanzala','Shahid','F88-4A','0359354264'

),

(

'B5','Hamza','Aslam','F11-3A','0351694264'

),

(

'B6','Hamza','Ahmad','F19-5A','0354654264'

);

insert into PropertyForRent

(

propertyNo,street,city,postcode,[type],rooms,rent,ownerNo,staffNo,branchNo

)

values

(

'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'

),

(

'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'

),

(

'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'

);

insert into Viewing

(

clientNo,

propertyNo,

viewDate,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

insert into Viewing

(

clientNo,

propertyNo,

viewDate,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

**End of qno1**

**Qno:-2**

**Answer:-**

select \* from Branch update Branch set city='ABT' where city='ISB';

### Hamza Bin Ahmed

**FA18-BCS-084**

**Qno1:-**

**Answer:-**

create database DreamHome;

USE DreamHome;

create table Branch

(

branchNo varchar(20) NOT NULL Primary Key,

street varchar(100) NOT NULL,

city varchar(50) NOT NULL,

postcode varchar(20) NOT NULL

);

create table Staff

(

staffNo varchar(20) NOT NULL PRIMARY KEY,

fName varchar(50) NOT NULL,

lName varchar(50) NOT NULL,

position varchar(50) NOT NULL,

sex varchar(1) NOT NULL, DOB DateTime NOT NULL,

salary DECIMAL NOT NULL,

branchNo varchar(20) NOT NULL References Branch(branchNo)

);

create table Client

(

clientNo varchar(20) NOT NULL PRIMARY KEY,

fName varchar(50) NOT NULL,

lName varchar(50) NOT NULL,

telNo varchar(20) NOT NULL,

prefType varchar(50) NOT NULL,

maxRent DECIMAL NOT NULL

);

create table PrivateOwner

(

ownerNo varchar(20) NOT NULL PRIMARY KEY,

fName varchar(50) NOT NULL,

lName varchar(50) NOT NULL,

address varchar(50) NOT NULL,

telNo varchar(20) NOT NULL

);

create table PropertyForRent

(

propertyNo varchar(20) NOT NULL PRIMARY KEY,

street varchar(100) NOT NULL,

city varchar(50) NOT NULL,

postcode varchar(20) NOT NULL,

type varchar(10) NOT NULL,

rooms int NOT NULL,

rent DECIMAL NOT NULL,

ownerNo varchar(20) References PrivateOwner(ownerNo),

staffNo varchar(20) NOT NULL References Staff(staffNo),

branchNo varchar(20) NOT NULL References Branch(branchNo)

);

create table Viewing

(

clientNo varchar(20) NOT NULL References Client(clientNo),

propertyNo varchar(20) NOT NULL References PropertyForRent(propertyNo),

viewDate DateTime NOT NULL,

comment varchar(200) NOT NULL

);

create table Registration

(

clientNo varchar(20) NOT NULL References Client(clientNo),

branchNo varchar(20) NOT NULL References Branch(branchNo),

staffNo varchar(20) NOT NULL References Staff(staffNo),

dateJoined DateTime NOT NULL

);

INSERT into Branch

(

branchNo,

street,

city,

postcode

)

VALUES

(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),

(N'B002',N'H#78 Supply', N'ABT', N'53000'),

(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),

(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff

(

staffNo,

fName,

lName,

position,

sex,

DOB,

salary,

branchNo

)

VALUES

(

N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS DateTime),

CAST(9000 AS Decimal(18, 0)),N'B002'

),

(N'SG14', N'David', N'Ford', N'Supervisor', N'M',

CAST(0x0000531200000000 AS DateTime), CAST(18000 AS Decimal(18,0)),

N'B003'),

(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',

CAST(0x000056D400000000 AS DateTime), CAST(12000 AS Decimal(18,0)),

N'B003'),

(N'SG5',N'Susan', N'Brand', N'Manager', N'F',

CAST(0x0000C85800000000 AS DateTime), CAST(24000 AS Decimal(18,0)),

N'B003'),

(N'SL21', N'John', N'White', N'Manager', N'M',

CAST(0x0000CFF200000000 AS DateTime), CAST(30000 AS Decimal(18,0)),

N'B004'),

(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',

CAST(0x00005D6000000000 AS DateTime), CAST(9000 AS Decimal(18, 0)),

N'B002');

insert into Client

(

clientNo,

fName,

lName,

telNo,

prefType,

maxRent

)

values

(

'B1001','Mahad','Ali','030078601','yes',1000.0

),

(

'B1002','Sharjeel','Khan','030054621','yes',2000.0

),

(

'B1003','Hanzala','Shahid','030456601','no',1500.0

),

(

'B1004','Hamza','Aslam','0306446641','yes',8800.0

),

(

'B1005','Hamza','Ahmad','0354654401','noo',800.0

),

(

'B1006','Usama','Fareed','030074541','yes',4000.0

);

insert into PrivateOwner

(

ownerNo,

fName,

lName,

[address],

telNo

)

values

(

'B1','Azid','Ali','F18-4A','0354654264'

),

(

'B2','Mahad','Ali','F17-4A','0345154264'

),

(

'B3','Sharjeel','Khan','F14-7A','0352354264'

),

(

'B4','Hanzala','Shahid','F88-4A','0359354264'

),

(

'B5','Hamza','Aslam','F11-3A','0351694264'

),

(

'B6','Hamza','Ahmad','F19-5A','0354654264'

);

insert into PropertyForRent

(

propertyNo,street,city,postcode,[type],rooms,rent,ownerNo,staffNo,branchNo

)

values

(

'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'

),

(

'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'

),

(

'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'

);

insert into Viewing

(

clientNo,

propertyNo,

viewDate,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

insert into Viewing

(

clientNo,

propertyNo,

viewDate,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

**End of qno1**

**Qno:-2**

**Answer:-**

select \* from Branch update Branch set city='ABT' where city='ISB';

### Muhammad Sharjeel khan

**FA18-BCS-019**

**Qno1: -**

**Answer: -**

create database DreamHome;

USE DreamHome;

create table Branch

(

Branch\_No varchar(18) NOT NULL Primary Key,

Street\_No varchar(90) NOT NULL,

City\_Name varchar(50) NOT NULL,

postcode int(20) NOT NULL

);

create table Staff

(

Staff\_No varchar(20) NOT NULL PRIMARY KEY,

First\_N varchar(50) NOT NULL,

Last\_N varchar(50) NOT NULL,

position varchar(50) NOT NULL,

sex varchar(1) NOT NULL, DOB Date-Time NOT NULL,

salary DECIMAL NOT NULL,

branch\_No varchar(20) NOT NULL References Branch(branch\_No)

);

create table Client

(

Client\_No varchar(20) NOT NULL PRIMARY KEY,

First\_N varchar(50) NOT NULL,

Last\_N varchar(50) NOT NULL,

Tel\_No int(20) NOT NULL,

prefType varchar(50) NOT NULL,

maxRent DECIMAL NOT NULL

);

create table PrivateOwner

(

Owner\_No varchar(20) NOT NULL PRIMARY KEY,

First\_N varchar(50) NOT NULL,

Last\_N varchar(50) NOT NULL,

address varchar(50) NOT NULL,

tel\_No int(20) NOT NULL

);

create table PropertyForRent

(

Property\_No varchar(20) NOT NULL PRIMARY KEY,

Street\_NO varchar(100) NOT NULL,

City\_Name varchar(50) NOT NULL,

postcode int(20) NOT NULL,

type varchar(10) NOT NULL,

rooms int NOT NULL,

rent DECIMAL NOT NULL,

owner\_No varchar(20) References PrivateOwner(owner\_No),

staff\_No varchar(20) NOT NULL References Staff(staff\_No),

branch\_No varchar(20) NOT NULL References Branch(branch\_No)

);

create table Viewing

(

Client\_No varchar(20) NOT NULL References Client(client\_No),

Property\_No varchar(20) NOT NULL References PropertyForRent(property\_No),

View-Date DateTime NOT NULL,

comment varchar(200) NOT NULL

);

create table Registration

(

Client\_No varchar(20) NOT NULL References Client(client\_No),

Branch\_No varchar(20) NOT NULL References Branch(branch\_No),

Staff\_No varchar(20) NOT NULL References Staff(staff\_No),

Date-Joined Date-Time NOT NULL

);

INSERT into Branch

(

Branch\_No,

Street\_no,

City\_Name,

postcode

)

VALUES

(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),

(N'B002',N'H#78 Supply', N'ABT', N'53000'),

(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),

(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff

(

Staff\_No,

First\_N,

Last\_N,

position,

sex,

DOB,

salary,

branch\_No

)

VALUES

(

N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS Date-Time),

CAST(9000 AS Decimal(18, 0)),N'B002'

),

(N'SG14', N'David', N'Ford', N'Supervisor', N'M',

CAST(0x0000531200000000 AS Date-Time), CAST(18000 AS Decimal(18,0)),

N'B003'),

(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',

CAST(0x000056D400000000 AS Date-Time), CAST(12000 AS Decimal(18,0)),

N'B003'),

(N'SG5',N'Susan', N'Brand', N'Manager', N'F',

CAST(0x0000C85800000000 AS Date-Time), CAST(24000 AS Decimal(18,0)),

N'B003'),

(N'SL21', N'John', N'White', N'Manager', N'M',

CAST(0x0000CFF200000000 AS Date-Time), CAST(30000 AS Decimal(18,0)),

N'B004'),

(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',

CAST(0x00005D6000000000 AS Date-Time), CAST(9000 AS Decimal(18, 0)),

N'B002');

insert into Client

(

Client\_No,

First\_N,

Last\_N,

Tel\_No,

prefType,

maxRent

)

values

(

'B1001','Mahad','Ali','030078601','yes',1000.0

),

(

'B1002','Sharjeel','Khan','030054621','yes',2000.0

),

(

'B1003','Hanzala','Shahid','030456601','no',1500.0

),

(

'B1004','Hamza','Aslam','0306446641','yes',8800.0

),

(

'B1005','Hamza','Ahmad','0354654401','noo',800.0

),

(

'B1006','Usama','Fareed','030074541','yes',4000.0

),

(

'B1007','Abdullah','Niazi','03359613366','yes',2500.0

);

insert into PrivateOwner

(

Owner\_No,

First\_n,

Last\_n,

[address],

Tel\_No

)

values

(

'B1','Azid','Ali','F18-4A','0354654264'

),

(

'B2','Mahad','Ali','F17-4A','0345154264'

),

(

'B3','Sharjeel','Khan','F14-7A','0352354264'

),

(

'B4','Hanzala','Shahid','F88-4A','0359354264'

),

(

'B5','Hamza','Aslam','F11-3A','0351694264'

),

(

'B6','Hamza','Ahmad','F19-5A','0354654264'

)’

(

'B7','Abdullah','Niazi','F18-5A','03359613366'

);

insert into PropertyForRent

(

Property\_No,street\_NO,city\_Name,postcode,[type],rooms,rent,owner\_No,staff\_No,branch\_No

)

values

(

'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'

),

(

'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'

),

(

'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'

);

insert into Viewing

(

Client\_No,

Property\_No,

View-Date,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

insert into Viewing

(

Client\_No,

Property\_No,

View-Date,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

**End of qno1**

**Qno:-2**

**Answer:-**

select \* from Branch update Branch set city='ABT' where city='ISB';

### Hamza Aslam

**FA18-BCS-050**

**Qno1:-**

**Answer:-**

create database HomeSweetHome;

USE HomeSweetHome;

create table Branch

(

branchNumber varchar(25) NOT NULL Primary Key,

streetNumber varchar(110) NOT NULL,

cityName varchar(60) NOT NULL,

postcodeNumber int NOT NULL

);

create table Staff

(

staffNumber int NOT NULL PRIMARY KEY,

firstName varchar(50) NOT NULL,

lastName varchar(50) NOT NULL,

position varchar(50) NOT NULL,

sex char NOT NULL, DOB DateTime NOT NULL,

salary DECIMAL NOT NULL,

branchNumber varchar(25) NOT NULL References Branch(branchNumber)

);

create table Client

(

clientNumber varchar(20) NOT NULL PRIMARY KEY,

firstName varchar(50) NOT NULL,

lastName varchar(50) NOT NULL,

phoneNo int NOT NULL,

prefType varchar(50) NOT NULL,

maxRent DECIMAL NOT NULL

);

create table PrivateOwner

(

ownerNumber varchar(20) NOT NULL PRIMARY KEY,

firstName varchar(50) NOT NULL,

lastName varchar(50) NOT NULL,

address varchar(50) NOT NULL,

phoneNo int NOT NULL

);

create table PropertyForRent

(

propertyNumber varchar(20) NOT NULL PRIMARY KEY,

streetNumber varchar(100) NOT NULL,

cityName varchar(50) NOT NULL,

postcodeNumber int NOT NULL,

type varchar(10) NOT NULL,

rooms int NOT NULL,

rent DECIMAL NOT NULL,

ownerNumber varchar(20) References PrivateOwner(ownerNumber),

staffNumber varchar(20) NOT NULL References Staff(staffNumber),

branchNumber varchar(20) NOT NULL References Branch(branchNumber)

);

create table Viewing

(

clientNumber varchar(20) NOT NULL References Client(clientNumber),

propertyNumber varchar(20) NOT NULL References PropertyForRent(propertyNumber),

viewDate DateTime NOT NULL,

comment varchar(200) NOT NULL

);

create table Registration

(

clientNumber varchar(20) NOT NULL References Client(clientNumber),

branchNumber varchar(20) NOT NULL References Branch(branchNumber),

staffNumber varchar(20) NOT NULL References Staff(staffNumber),

dateJoined DateTime NOT NULL

);

INSERT into Branch

(

branchNumber,

streetNumber,

cityNumber,

postcodeNumber

)

VALUES

(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),

(N'B002',N'H#78 Supply', N'ABT', N'53000'),

(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),

(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff

(

staffNumber,

firstName,

lastName,

position,

sex,

DOB,

salary,

branchNumber

)

VALUES

(

N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS DateTime),

CAST(9000 AS Decimal(18, 0)),N'B002'

),

(N'SG14', N'David', N'Ford', N'Supervisor', N'M',

CAST(0x0000531200000000 AS DateTime), CAST(18000 AS Decimal(18,0)),

N'B003'),

(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',

CAST(0x000056D400000000 AS DateTime), CAST(12000 AS Decimal(18,0)),

N'B003'),

(N'SG5',N'Susan', N'Brand', N'Manager', N'F',

CAST(0x0000C85800000000 AS DateTime), CAST(24000 AS Decimal(18,0)),

N'B003'),

(N'SL21', N'John', N'White', N'Manager', N'M',

CAST(0x0000CFF200000000 AS DateTime), CAST(30000 AS Decimal(18,0)),

N'B004'),

(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',

CAST(0x00005D6000000000 AS DateTime), CAST(9000 AS Decimal(18, 0)),

N'B002');

insert into Client

(

clientNumber,

firstName,

lastName,

PhoneNumber,

prefType,

maxRent

)

values

(

'B1001','Mahad','Ali','030078601','yes',1000.0

),

(

'B1002','Sharjeel','Khan','030054621','yes',2000.0

),

(

'B1003','Hanzala','Shahid','030456601','no',1500.0

),

(

'B1004','Hamza','Aslam','0306446641','yes',8800.0

),

(

'B1005','Hamza','Ahmad','0354654401','noo',800.0

),

(

'B1006','Usama','Fareed','030074541','yes',4000.0

),

(

'B1007','Abdullah','Niazi','03359613366','yes',2500.0

);

insert into PrivateOwner

(

ownerNumber,

firstName,

lastName,

[address],

PhoneNumber

)

values

(

'B1','Azid','Ali','F18-4A','0354654264'

),

(

'B2','Mahad','Ali','F17-4A','0345154264'

),

(

'B3','Sharjeel','Khan','F14-7A','0352354264'

),

(

'B4','Hanzala','Shahid','F88-4A','0359354264'

),

(

'B5','Hamza','Aslam','F11-3A','0351694264'

),

(

'B6','Hamza','Ahmad','F19-5A','0354654264'

)

(

'B7','Abdullah','Niazi','F18-5A','03359613366'

);

insert into PropertyForRent

(

propertyNumber,streetNumber,cityName,postcodeNumber,[type],rooms,rent,ownerNumnber,staffNumber,branchNumber

)

values

(

'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'

),

(

'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'

),

(

'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'

);

insert into Viewing

(

clientNumber,

propertyNumber,

viewDate,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

insert into Viewing

(

clientNumber,

propertyNumber,

viewDate,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

**Qno:-2**

**Answer:-**

select \* from Branch update Branch set city='ABT' where city='ISB';

### Abdullah Noor Niazi

**FA18-BCS-004**

**Qno1:-**

**Answer:-**

create database HomeSweetHome;

USE HomeSweetHome;

create table Branch

(

branchNumber varchar(25) NOT NULL Primary Key,

streetNumber varchar(110) NOT NULL,

cityName varchar(60) NOT NULL,

postcodeNumber int NOT NULL

);

create table Staff

(

staffNumber int NOT NULL PRIMARY KEY,

firstName varchar(50) NOT NULL,

lastName varchar(50) NOT NULL,

position varchar(50) NOT NULL,

sex char NOT NULL, DOB DateTime NOT NULL,

salary DECIMAL NOT NULL,

branchNumber varchar(25) NOT NULL References Branch(branchNumber)

);

create table Client

(

clientNumber varchar(20) NOT NULL PRIMARY KEY,

firstName varchar(50) NOT NULL,

lastName varchar(50) NOT NULL,

phoneNo int NOT NULL,

prefType varchar(50) NOT NULL,

maxRent DECIMAL NOT NULL

);

create table PrivateOwner

(

ownerNumber varchar(20) NOT NULL PRIMARY KEY,

firstName varchar(50) NOT NULL,

lastName varchar(50) NOT NULL,

address varchar(50) NOT NULL,

phoneNo int NOT NULL

);

create table PropertyForRent

(

propertyNumber varchar(20) NOT NULL PRIMARY KEY,

streetNumber varchar(100) NOT NULL,

cityName varchar(50) NOT NULL,

postcodeNumber int NOT NULL,

type varchar(10) NOT NULL,

rooms int NOT NULL,

rent DECIMAL NOT NULL,

ownerNumber varchar(20) References PrivateOwner(ownerNumber),

staffNumber varchar(20) NOT NULL References Staff(staffNumber),

branchNumber varchar(20) NOT NULL References Branch(branchNumber)

);

create table Viewing

(

clientNumber varchar(20) NOT NULL References Client(clientNumber),

propertyNumber varchar(20) NOT NULL References PropertyForRent(propertyNumber),

viewDate DateTime NOT NULL,

comment varchar(200) NOT NULL

);

create table Registration

(

clientNumber varchar(20) NOT NULL References Client(clientNumber),

branchNumber varchar(20) NOT NULL References Branch(branchNumber),

staffNumber varchar(20) NOT NULL References Staff(staffNumber),

dateJoined DateTime NOT NULL

);

INSERT into Branch

(

branchNumber,

streetNumber,

cityNumber,

postcodeNumber

)

VALUES

(N'B001',N'H#7 I-10/2', N'ISB', N'52000'),

(N'B002',N'H#78 Supply', N'ABT', N'53000'),

(N'B003',N'H#79 I-10/2', N'ISB', N'52000'),

(N'B004',N'H#78 Mandian', N'ABT', N'53000');

insert into Staff

(

staffNumber,

firstName,

lastName,

position,

sex,

DOB,

salary,

branchNumber

)

VALUES

(

N'SA9', N'Mary', N'Howe', N'Assistant', N'F',CAST(0x0000641000000000 AS DateTime),

CAST(9000 AS Decimal(18, 0)),N'B002'

),

(N'SG14', N'David', N'Ford', N'Supervisor', N'M',

CAST(0x0000531200000000 AS DateTime), CAST(18000 AS Decimal(18,0)),

N'B003'),

(N'SG37', N'Ann', N'Beech', N'Assistant', N'F',

CAST(0x000056D400000000 AS DateTime), CAST(12000 AS Decimal(18,0)),

N'B003'),

(N'SG5',N'Susan', N'Brand', N'Manager', N'F',

CAST(0x0000C85800000000 AS DateTime), CAST(24000 AS Decimal(18,0)),

N'B003'),

(N'SL21', N'John', N'White', N'Manager', N'M',

CAST(0x0000CFF200000000 AS DateTime), CAST(30000 AS Decimal(18,0)),

N'B004'),

(N'SL41', N'Julie', N'Lee', N'Assistant', N'F',

CAST(0x00005D6000000000 AS DateTime), CAST(9000 AS Decimal(18, 0)),

N'B002');

insert into Client

(

clientNumber,

firstName,

lastName,

PhoneNumber,

prefType,

maxRent

)

values

(

'B1001','Mahad','Ali','030078601','yes',1000.0

),

(

'B1002','Sharjeel','Khan','030054621','yes',2000.0

),

(

'B1003','Hanzala','Shahid','030456601','no',1500.0

),

(

'B1004','Hamza','Aslam','0306446641','yes',8800.0

),

(

'B1005','Hamza','Ahmad','0354654401','noo',800.0

),

(

'B1006','Usama','Fareed','030074541','yes',4000.0

),

(

'B1007','Abdullah','Niazi','03359613366','yes',2500.0

);

insert into PrivateOwner

(

ownerNumber,

firstName,

lastName,

[address],

PhoneNumber

)

values

(

'B1','Azid','Ali','F18-4A','0354654264'

),

(

'B2','Mahad','Ali','F17-4A','0345154264'

),

(

'B3','Sharjeel','Khan','F14-7A','0352354264'

),

(

'B4','Hanzala','Shahid','F88-4A','0359354264'

),

(

'B5','Hamza','Aslam','F11-3A','0351694264'

),

(

'B6','Hamza','Ahmad','F19-5A','0354654264'

)

(

'B7','Abdullah','Niazi','F18-5A','03359613366'

);

insert into PropertyForRent

(

propertyNumber,streetNumber,cityName,postcodeNumber,[type],rooms,rent,ownerNumnber,staffNumber,branchNumber

)

values

(

'BF2','H2-h2','ABT','22010','large',8,'30000','B2','SG14','B002'

),

(

'BF3','H3-h3','ISB','62010','medium',6,'20000','B3','SG37','B003'

),

(

'BF4','H4-h4','ISB','62010','small',4,'10000','B4','SG5','B004'

);

insert into Viewing

(

clientNumber,

propertyNumber,

viewDate,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

insert into Viewing

(

clientNumber,

propertyNumber,

viewDate,

comment

)

values

(

'B1002','BF2','2020-2-10','No, i dont wanna say anything'

),

(

'B1003','BF3','2020-3-10','No, i dont wanna say anything'

),

(

'B1004','BF4','2020-4-10','No, i dont wanna say anything'

);

**Qno:-2**

**Answer:-**

select \* from Branch update Branch set city='ABT' where city='ISB';

### Usman Jadoon

**FA18-BCS-100**

Create database DreamHome;

USE DreamHome;

Create table Branch

(

branch No varchar(20)NOTNULL PrimaryKey,

street varchar(100)NOTNULL,

city varchar(20)NOTNULL,

postcode varchar(20)NOTNULL

);

Create table

(

stafF No varchar(20)NOTNULL PRIMARYKEY,

fName varchar(15)NOTNULL,

lName varchar(15)NOTNULL,

position varchar(15)NOTNULL,

sex varchar(15)NOTNULL,DOB DateTimeNOTNULL,

salary DECIMALNOTNULL,

branchNovarchar(15)NOTNULLReferences Branch(branchNo)

);

createtable Client

(

client No varchar(20)NOtnull PRIMARYKEY,

fName varchar(50)NOTNULL,

lName varchar(50)NOTNULL,

telNo varchar(20)NOTNULL,

PREFType varchar(50)NOTNULL,

maxRent DECMALNOTNULL,

);

CreatetablePrivateOwner

(

ownerNo varchar(20)NOtnullPRIMARYKY,

fName varchar(15)NOTNULL,

lName varchar(15)NOTNULL,

telNo varchar(15)NOTNULL,

);

CreatetablePropertyForRent

(

property No varchar(20)NOtnullPRIMARYKY,

street varchar(50)NOTNULL,

city varchar(15)NOTNULL,

postcode (15)NOTNULL,

rooms intNOTNULL,

rent DECIMALNOTNULL,

ownerNo varchar(20) References PrivateOwner(ownerNo);

staffNo varchar(20) References PrivateOwner(staffNo);

branchNo varchar(20) References PrivateOwner(branchNo);

);

createtable Viewing

(

clientNo varchar(20)NOTNULL References client(clientNo),

propertyNo varchar(15)NOTNULL References client(clientNo),

viewDateTimeNOTNULL,

COMMENT varchar(150)NOTNULL

);

createtable Registration

(

clientNovarchar(50)NOTNULL References client(clientNo),

branchNovarchar(50)NOTNULL References Branch(branchNo),

staffNovarchar(50)NOTNULL References Staff(staffNo),

dateJoinedDateTimeNOTNULL

);

INSERTinto Branch

(

branchNo,

street,

city,

postcode

)

VALUES

(N 'B005'N'H#7 I-12/2',N'KAR' '50000'),

(N 'B006'N'H#75 supply',N'ABT' '54000'),

(N 'B007'N'H#79 I-14/2',N'ISB' '55000'),

(N 'B008'N'H#78 Mandian',N'LHR' '55000'),

InserT into Client

(

ClientNo,

fName,

lName,

telNo,

prefType,

maxRent

)

values

(

'81001','Ali','0312789653','yes',1000.0

),

(

'81001','Nabeel','0312789753','yes',1500.0

),

(

'81001','usman','0312789153','no',1800.0

),

(

'81001','Ahmed','0312789253','yes',1300.0

),

(

'81001','Akhtar','0312789553','yes',14400.0

),

(

'81001','Sohail','0312789553','no',1300.0

),

insert into private owner

{

ownnerNo,

fNAME,

lName,

[adress],

telno,

values

(

'B1','Ali','Jamal',F18-4a',0354654264

),

(

'B2','Ajmal','Akmal',F17-4a','0354654264,

),

(

'B3','Khatak','Kamran',F16-4a',03541654264

),

(

'B4','Adnan','Amir',F19-4a',03554654264

),

(

'B5','Inam','Akhtar',F14-4a',03554654264

),

(

'b6','adnan','Umair',F13-4a',03584654264

),

insertinto viewing

(

Clientno,

PropertyNo,

viewData

comment

)

values

(

'B1001',BF2','2020-3-10','HELLO HOW ARE YOU'

),

(

'B1002',BF3','2020-3-10','HELLO HOW ARE YOU'

),

(

'B1003',BF4','2020-5-10','HELLO HOW ARE YOU'

),

Question 2

select\*from Branch update Branch set city='ABT'where city ='ISB';

### Usama Fareed

**FA18-BCS-026**

Create database DreamHome;

USE DreamHome;

Create table Branch

(

branch No varchar(20)NOTNULL PrimaryKey,

street varchar(100)NOTNULL,

city varchar(20)NOTNULL,

postcode varchar(20)NOTNULL

);

CreatetABLE

(

stafF No varchar(20)NOTNULL PRIMARYKEY,

fName varchar(15)NOTNULL,

lName varchar(15)NOTNULL,

position varchar(15)NOTNULL,

sex varchar(15)NOTNULL,DOB DateTimeNOTNULL,

salary DECIMALNOTNULL,

branchNovarchar(15)NOTNULLReferences Branch(branchNo)

);

createtable Client

(

client No varchar(20)NOtnullPRIMARYKY,

fName varchar(50)NOTNULL,

lName varchar(50)NOTNULL,

telNo varchar(20)NOTNULL,

PREFType varchar(50)NOTNULL,

maxRent DECMALNOTNULL,

);

CreatetablePrivateOwner

(

ownerNo varchar(20)NOtnullPRIMARYKY,

fName varchar(15)NOTNULL,

lName varchar(15)NOTNULL,

telNo varchar(15)NOTNULL,

);

CreatetablePropertyForRent

(

property No varchar(20)NOtnullPRIMARYKY,

street varchar(50)NOTNULL,

city varchar(15)NOTNULL,

postcode (15)NOTNULL,

rooms intNOTNULL,

rent DECIMALNOTNULL,

ownerNo varchar(20) References PrivateOwner(ownerNo);

staffNo varchar(20) References PrivateOwner(staffNo);

branchNo varchar(20) References PrivateOwner(branchNo);

);

createtable Viewing

(

clientNo varchar(20)NOTNULL References client(clientNo),

propertyNo varchar(15)NOTNULL References client(clientNo),

viewDateTimeNOTNULL,

COMMENT varchar(150)NOTNULL

);

createtable Registration

(

clientNovarchar(50)NOTNULL References client(clientNo),

branchNovarchar(50)NOTNULL References Branch(branchNo),

staffNovarchar(50)NOTNULL References Staff(staffNo),

dateJoinedDateTimeNOTNULL

);

INSERTinto Branch

(

branchNo,

street,

city,

postcode

)

VALUES

(N 'B005'N'H#7 I-12/2',N'KAR' '50000'),

(N 'B006'N'H#75 supply',N'ABT' '54000'),

(N 'B007'N'H#79 I-14/2',N'ISB' '55000'),

(N 'B008'N'H#78 Mandian',N'LHR' '55000'),

InserT into Client

(

ClientNo,

fName,

lName,

telNo,

prefType,

maxRent

)

values

(

'81001','Ali','0312789653','yes',1000.0

),

(

'81001','Nabeel','0312789753','yes',1500.0

),

(

'81001','usman','0312789153','no',1800.0

),

(

'81001','Ahmed','0312789253','yes',1300.0

),

(

'81001','Akhtar','0312789553','yes',14400.0

),

(

'81001','Sohail','0312789553','no',1300.0

),

insert into private owner

{

ownnerNo,

fNAME,

lName,

[adress],

telno,

values

(

'B1','Ali','Jamal',F18-4a',0354654264

),

(

'B2','Ajmal','Akmal',F17-4a','0354654264,

),

(

'B3','Khatak','Kamran',F16-4a',03541654264

),

(

'B4','Adnan','Amir',F19-4a',03554654264

),

(

'B5','Inam','Akhtar',F14-4a',03554654264

),

(

'b6','adnan','Umair',F13-4a',03584654264

),

insertinto viewing

(

Clientno,

PropertyNo,

viewData

comment

)

values

(

'B1001',BF2','2020-3-10','HELLO HOW ARE YOU'

),

(

'B1002',BF3','2020-3-10','HELLO HOW ARE YOU'

),

(

'B1003',BF4','2020-5-10','HELLO HOW ARE YOU'

),

Question 2

select\*from Branch update Branch set city='ABT'where city ='ISB';

# Lab 7

Common Solution**:-**

**Qno1:-**

**Answer**

select distinct(postcode) from Branch

**Qno2:-**

**Answer**

select distinct(fName) from Staff

**Qno3:-**

**Answer**

select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],

position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,

branchNo as [Section No] from Staff

**Qno4:-**

**Answer**

select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],

telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from Client;

**Qno5:-**

**Answer**

select \* from Staff where salary>10000

**Qno6:-**

**Answer**

select \* from Staff where position='Manager' or position='Supervisor'

## Individual Solutions:-

**Lab 7**

### Hanzala Shahid

**FA18-BCS-014**

**Qno1:-**

**Answer**

select distinct(postcode) from Branch

**Qno2:-**

**Answer**

select distinct(fName) from Staff

**Qno3:-**

**Answer**

select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],

position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,

branchNo as [Section No] from Staff

**Qno4:-**

**Answer**

select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],

telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from Client;

**Qno5:-**

**Answer**

select \* from Staff where salary>10000

**Qno6:-**

**Answer**

select \* from Staff where position='Manager' or position='Supervisor'

### Hamza Bin Ahmed

**FA18-BCS-084**

**Qno1:-**

**Answer**

select distinct(postcode) from Branch

**Qno2:-**

**Answer**

select distinct(fName) from Staff

**Qno3:-**

**Answer**

select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],

position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,

branchNo as [Section No] from Staff

**Qno4:-**

**Answer**

select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],

telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from Client;

**Qno5:-**

**Answer**

select \* from Staff where salary>10000

**Qno6:-**

**Answer**

select \* from Staff where position='Manager' or position='Supervisor'

### Muhammad Sharjeel khan

**FA18-BCS-019**

select distinct(postcode) from Branch

select distinct(fName) from Staff

select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],

position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,

branchNo as [Section No] from Staff

select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],

telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from Client;

select \* from Staff where salary>10000

select \* from Staff where position='Manager' or position='Supervisor'

### Hamza Aslam

**FA18-BCS-050**

**Qno1:-**

**Answer**

select distinct(postcodeNumber) from Branch

**Qno2:-**

**Answer**

select distinct(firstName) from Staff

**Qno3:-**

**Answer**

select staffNumber as [Cadre No], firstName as [Baptism Name], lastName as [Sur name],

position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,

branchNumber as [Section No] from Staff

**Qno4:-**

**Answer**

select clientNumber as [Buyer No], firstName as [Baptism Name], lastName as [Sur name],

PhoneNumber as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from Client;

**Qno5:-**

**Answer**

select \* from Staff where salary>10000

**Qno6:-**

**Answer**

select \* from Staff where position='Manager' or position='Supervisor'

### Abdullah Noor Niazi

**FA18-BCS-004**

**Qno1:-**

**Answer**

select distinct(postcodeNumber) from Branch

**Qno2:-**

**Answer**

select distinct(firstName) from Staff

**Qno3:-**

**Answer**

select staffNumber as [Cadre No], firstName as [Baptism Name], lastName as [Sur name],

position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,

branchNumber as [Section No] from Staff

**Qno4:-**

**Answer**

select clientNumber as [Buyer No], firstName as [Baptism Name], lastName as [Sur name],

PhoneNumber as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from Client;

**Qno5:-**

**Answer**

select \* from Staff where salary>10000

**Qno6:-**

**Answer**

select \* from Staff where position='Manager' or position='Supervisor'

### Usman Jadoon

**FA18-BCS-100**

**Qno1:-**

**Answer**

select distinct(postcode) from Branch

**Qno2:-**

**Answer**

select distinct(fName) from Staff

**Qno3:-**

**Answer**

select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],

position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,

branchNo as [Section No] from Staff

**Qno4:-**

**Answer**

select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],

telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from Client;

**Qno5:-**

**Answer**

select \* from Staff where salary>10000

**Qno6:-**

**Answer**

select \* from Staff where position='Manager' or position='Supervisor'

### Usama Fareed

**FA18-BCS-026**

**Qno1:-**

**Answer**

select distinct(postcode) from Branch

**Qno2:-**

**Answer**

select distinct(fName) from Staff

**Qno3:-**

**Answer**

select staffNo as [Cadre No], fName as [Baptism Name], lName as [Sur name],

position as [Locale], sex as [Gender],DOB as [Birtday] ,salary as Income,

branchNo as [Section No] from Staff

**Qno4:-**

**Answer**

select clientNo as [Buyer No], fName as [Baptism Name], lName as [Sur name],

telNo as [Fax Number],prefType as [Proclivity Type],maxRent as [Supreme Cost] from Client;

**Qno5:-**

**Answer**

select \* from Staff where salary>10000

**Qno6:-**

**Answer**

select \* from Staff where position='Manager' or position='Supervisor'

# Lab 8

Common Solution**:-**

**Qno1:-**

**Answer**

select staffNo,fName,lName,salary from staff order by salary desc

**Qno2:-**

**Answer**

select propertyNo,type,rooms,rent from PropertyForRent

order by type

select propertyNo,type,rooms,rent

from PropertyForRent

order by type,rent desc

**Qno3:-**

**Answer**

select count(\*) as myCount

from PropertyForRent

where rent<=500

**Qno4:-**

**Answer**

select count(Distinct propertyNo) As myCount from Viewing

WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';

**Qno5:-**

**Answer**

select count(staffNo) as myCount,sum(salary) as mySalary from staff

where

position='Manager'

**Qno6:-**

**Answer**

select MIN(salary) as myMin,

MAX(salary) as myMax,

AVG(salary) as myAVG from Staff

**Qno7:-**

**Answer**

SELECT staffNo, fName, lName, position, salary

FROM Staff

WHERE (SELECT AVG(salary) FROM Staff) < salary;

**Qno8:-**

**Answer**

select \*from Staff where salary> any(select salary from Staff where branchNo='B003')

**Qno9:-**

**Answer**

select \*from Staff where salary> all(select salary from Staff where branchNo='B003')

## Individual Solutions:-

**Lab 8**

### Hanzala Shahid

**FA18-BCS-014**

**Qno1:-**

**Answer**

select staffNo,fName,lName,salary from staff order by salary desc

**Qno2:-**

**Answer**

select propertyNo,type,rooms,rent from PropertyForRent

order by type

select propertyNo,type,rooms,rent

from PropertyForRent

order by type,rent desc

**Qno3:-**

**Answer**

select count(\*) as myCount

from PropertyForRent

where rent<=500

**Qno4:-**

**Answer**

select count(Distinct propertyNo) As myCount from Viewing

WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';

**Qno5:-**

**Answer**

select count(staffNo) as myCount,sum(salary) as mySalary from staff

where

position='Manager'

**Qno6:-**

**Answer**

select MIN(salary) as myMin,

MAX(salary) as myMax,

AVG(salary) as myAVG from Staff

**Qno7:-**

**Answer**

SELECT staffNo, fName, lName, position, salary

FROM Staff

WHERE (SELECT AVG(salary) FROM Staff) < salary;

**Qno8:-**

**Answer**

select \*from Staff where salary> any(select salary from Staff where branchNo='B003')

**Qno9:-**

**Answer**

select \*from Staff where salary> all(select salary from Staff where branchNo='B003')

### Hamza Bin Ahmed

**FA18-BCS-084**

**Qno1:-**

**Answer**

select staffNo,fName,lName,salary from staff order by salary desc

**Qno2:-**

**Answer**

select propertyNo,type,rooms,rent from PropertyForRent

order by type

select propertyNo,type,rooms,rent

from PropertyForRent

order by type,rent desc

**Qno3:-**

**Answer**

select count(\*) as myCount

from PropertyForRent

where rent<=500

**Qno4:-**

**Answer**

select count(Distinct propertyNo) As myCount from Viewing

WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';

**Qno5:-**

**Answer**

select count(staffNo) as myCount,sum(salary) as mySalary from staff

where

position='Manager'

**Qno6:-**

**Answer**

select MIN(salary) as myMin,

MAX(salary) as myMax,

AVG(salary) as myAVG from Staff

**Qno7:-**

**Answer**

SELECT staffNo, fName, lName, position, salary

FROM Staff

WHERE (SELECT AVG(salary) FROM Staff) < salary;

**Qno8:-**

**Answer**

select \*from Staff where salary> any(select salary from Staff where branchNo='B003')

**Qno9:-**

**Answer**

select \*from Staff where salary> all(select salary from Staff where branchNo='B003')

### Muhammad Sharjeel khan

**FA18-BCS-019**

select staffNo,fName,lName,salary from staff order by salary desc

select propertyNo,type,rooms,rent from PropertyForRent

order by type

select propertyNo,type,rooms,rent

from PropertyForRent

order by type,rent desc

select count(\*) as myCount

from PropertyForRent

where rent<=500

select count(Distinct propertyNo) As myCount from Viewing

WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';

select count(staffNo) as myCount,sum(salary) as mySalary from staff

where

position='Manager'

select MIN(salary) as myMin,

MAX(salary) as myMax,

AVG(salary) as myAVG from Staff

SELECT staffNo, fName, lName, position, salary

FROM Staff

WHERE (SELECT AVG(salary) FROM Staff) < salary;

select \*from Staff where salary> any(select salary from Staff where branchNo='B003')

select \*from Staff where salary> all(select salary from Staff where branchNo='B003')

### Hamza Aslam

**FA18-BCS-050**

**Qno1:-**

**Answer**

select staffNumber,firstName,lastName,salary from staff order by salary desc

**Qno2:-**

**Answer**

select propertyNumber,type,rooms,rent from PropertyForRent

order by type

select propertyNumber,type,rooms,rent

from PropertyForRent

order by type,rent desc

**Qno3:-**

**Answer**

select count(\*) as myCount

from PropertyForRent

where rent<=500

**Qno4:-**

**Answer**

select count(Distinct propertyNumber) As myCount from Viewing

WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';

**Qno5:-**

**Answer**

select count(staffNumber) as myCount,sum(salary) as mySalary from staff

where

position='Manager'

**Qno6:-**

**Answer**

select MIN(salary) as myMin,

MAX(salary) as myMax,

AVG(salary) as myAVG from Staff

**Qno7:-**

**Answer**

SELECT staffNumber, firstName, lastName, position, salary

FROM Staff

WHERE (SELECT AVG(salary) FROM Staff) < salary;

**Qno8:-**

**Answer**

select \*from Staff where salary> any(select salary from Staff where branchNumber='B003')

**Qno9:-**

**Answer**

select \*from Staff where salary> all(select salary from Staff where branchNumber='B003')

### Abdullah Noor Niazi

**FA18-BCS-004**

**Qno1:-**

**Answer**

select staffNumber,firstName,lastName,salary from staff order by salary desc

**Qno2:-**

**Answer**

select propertyNumber,type,rooms,rent from PropertyForRent

order by type

select propertyNumber,type,rooms,rent

from PropertyForRent

order by type,rent desc

**Qno3:-**

**Answer**

select count(\*) as myCount

from PropertyForRent

where rent<=500

**Qno4:-**

**Answer**

select count(Distinct propertyNumber) As myCount from Viewing

WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';

**Qno5:-**

**Answer**

select count(staffNumber) as myCount,sum(salary) as mySalary from staff

where

position='Manager'

**Qno6:-**

**Answer**

select MIN(salary) as myMin,

MAX(salary) as myMax,

AVG(salary) as myAVG from Staff

**Qno7:-**

**Answer**

SELECT staffNumber, firstName, lastName, position, salary

FROM Staff

WHERE (SELECT AVG(salary) FROM Staff) < salary;

**Qno8:-**

**Answer**

select \*from Staff where salary> any(select salary from Staff where branchNumber='B003')

**Qno9:-**

**Answer**

select \*from Staff where salary> all(select salary from Staff where branchNumber='B003')

### Usman Jadoon

**FA18-BCS-100**

**Qno1:-**

**Answer**

select staffNo,fName,lName,salary from staff order by salary desc

**Qno2:-**

**Answer**

select propertyNo,type,rooms,rent from PropertyForRent

order by type

select propertyNo,type,rooms,rent

from PropertyForRent

order by type,rent desc

**Qno3:-**

**Answer**

select count(\*) as myCount

from PropertyForRent

where rent<=500

**Qno4:-**

**Answer**

select count(Distinct propertyNo) As myCount from Viewing

WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';

**Qno5:-**

**Answer**

select count(staffNo) as myCount,sum(salary) as mySalary from staff

where

position='Manager'

**Qno6:-**

**Answer**

select MIN(salary) as myMin,

MAX(salary) as myMax,

AVG(salary) as myAVG from Staff

**Qno7:-**

**Answer**

SELECT staffNo, fName, lName, position, salary

FROM Staff

WHERE (SELECT AVG(salary) FROM Staff) < salary;

**Qno8:-**

**Answer**

select \*from Staff where salary> any(select salary from Staff where branchNo='B003')

**Qno9:-**

**Answer**

select \*from Staff where salary> all(select salary from Staff where branchNo='B003')

### Usama Fareed

**FA18-BCS-026**

**Qno1:-**

**Answer**

select staffNo,fName,lName,salary from staff order by salary desc

**Qno2:-**

**Answer**

select propertyNo,type,rooms,rent from PropertyForRent

order by type

select propertyNo,type,rooms,rent

from PropertyForRent

order by type,rent desc

**Qno3:-**

**Answer**

select count(\*) as myCount

from PropertyForRent

where rent<=500

**Qno4:-**

**Answer**

select count(Distinct propertyNo) As myCount from Viewing

WHERE viewDate BETWEEN '1-May-04' AND '31-May-04';

**Qno5:-**

**Answer**

select count(staffNo) as myCount,sum(salary) as mySalary from staff

where

position='Manager'

**Qno6:-**

**Answer**

select MIN(salary) as myMin,

MAX(salary) as myMax,

AVG(salary) as myAVG from Staff

**Qno7:-**

**Answer**

SELECT staffNo, fName, lName, position, salary

FROM Staff

WHERE (SELECT AVG(salary) FROM Staff) < salary;

**Qno8:-**

**Answer**

select \*from Staff where salary> any(select salary from Staff where branchNo='B003')

**Qno9:-**

**Answer**

select \*from Staff where salary> all(select salary from Staff where branchNo='B003')

# Lab 9

Common Solution**:-**

**Qno1:**

list all tables in the employees database

Answer:

USE EMPLOYEE;

show TABLES;

**Qno2:**

Write a query to find the names (first\_name, last\_name) and the salaries of the employees who have a higher salary than the employee whose last\_name=’Bull’.

Answer:-

Select FIRST\_NAME, LAST\_NAME, SALARY

FROM employees

WHERE SALARY>(SELECT salary FROM employees WHERE last\_name=’Bull’);

**Qno3:**

**Answer:-**

Select first\_name , last\_name

FROM employees

WHERE department\_id

IN(SELECT department\_id FROM departments WHERE department\_name=’IT’);

## Individual Solutions:-

**Lab 9**

### Hanzala Shahid

**FA18-BCS-014**

2. Write the following queries.

1.Write a query to find the names (first\_name, last\_name) and the salaries of the employees who have a higher salary than the employee who’s last\_name=’Bull’.

Answer#

SELECT FIRST\_NAME, LAST\_NAME, SALARY

FROM employees

WHERE SALARY >

(SELECT salary FROM employees WHERE last\_name = 'Bull');

2• Write a query to find the names (first\_name, last\_name) of all employeeswho works in the IT department.

Answer#

SELECT first\_name, last\_name

FROM employees

WHERE department\_id

IN (SELECT department\_id FROM departments WHERE department\_name='IT');

### Hamza Bin Ahmed

**FA18-BCS-084**

2. Write the following queries.

1.Write a query to find the names (first\_name, last\_name) and the salaries of the employees who have a higher salary than the employee who’s last\_name=’Bull’.

Answer#

SELECT FIRST\_NAME, LAST\_NAME, SALARY

FROM employees

WHERE SALARY >

(SELECT salary FROM employees WHERE last\_name = 'Bull');

2• Write a query to find the names (first\_name, last\_name) of all employeeswho works in the IT department.

Answer#

SELECT first\_name, last\_name

FROM employees

WHERE department\_id

IN (SELECT department\_id FROM departments WHERE department\_name='IT');

### Muhammad Sharjeel khan

**FA18-BCS-019**

2. Write the following queries.

1.Write a query to find the names (first\_name, last\_name) and the salaries of the employees who have a higher salary than the employee who’s last\_name=’Bull’.

Answer#

SELECT FIRST\_NAME, LAST\_NAME, SALARY

FROM employees

WHERE SALARY >

(SELECT salary FROM employees WHERE last\_name = 'Bull');

2• Write a query to find the names (first\_name, last\_name) of all employeeswho works in the IT department.

Answer#

SELECT first\_name, last\_name

FROM employees

WHERE department\_id

IN (SELECT department\_id FROM departments WHERE department\_name='IT');

### Hamza Aslam

**FA18-BCS-050**

2. Write the following queries.

1.Write a query to find the names (first\_name, last\_name) and the salaries of the employees who have a higher salary than the employee who’s last\_name=’Bull’.

Answer#

SELECT FIRST\_NAME, LAST\_NAME, SALARY

FROM employees

WHERE SALARY >

(SELECT salary FROM employees WHERE last\_name = 'Bull');

2• Write a query to find the names (first\_name, last\_name) of all employeeswho works in the IT department.

Answer#

SELECT first\_name, last\_name

FROM employees

WHERE department\_id

IN (SELECT department\_id FROM departments WHERE department\_name='IT');

### Abdullah Noor Niazi

**FA18-BCS-004**

Qno1:

list all tables in the employees database

Answer:

USE EMPLOYEE;

show TABLES;

Qno2:

Write a query to find the names (first\_name, last\_name) and the salaries of the employees who have a higher salary than the employee whose last\_name=’Bull’.

Answer:-

Select FIRST\_NAME, LAST\_NAME, SALARY

FROM employees

WHERE SALARY>(SELECT salary FROM employees WHERE last\_name=’Bull’);

Qno3:

Answer:-

Select first\_name , last\_name

FROM employees

WHERE department\_id

IN(SELECT department\_id FROM departments WHERE department\_name=’IT’);

### Usman Jadoon

**FA18-BCS-100**

2. Write the following queries.

1.Write a query to find the names (first\_name, last\_name) and the salaries of the employees who have a higher salary than the employee who’s last\_name=’Bull’.

Answer#

SELECT FIRST\_NAME, LAST\_NAME, SALARY

FROM employees

WHERE SALARY >

(SELECT salary FROM employees WHERE last\_name = 'Bull');

2• Write a query to find the names (first\_name, last\_name) of all employeeswho works in the IT department.

Answer#

SELECT first\_name, last\_name

FROM employees

WHERE department\_id

IN (SELECT department\_id FROM departments WHERE department\_name='IT');

### Usama Fareed

**FA18-BCS-026**

2. Write the following queries.

1.Write a query to find the names (first\_name, last\_name) and the salaries of the employees who have a higher salary than the employee who’s last\_name=’Bull’.

Answer#

SELECT FIRST\_NAME, LAST\_NAME, SALARY

FROM employees

WHERE SALARY >

(SELECT salary FROM employees WHERE last\_name = 'Bull');

2• Write a query to find the names (first\_name, last\_name) of all employeeswho works in the IT department.

Answer#

SELECT first\_name, last\_name

FROM employees

WHERE department\_id

IN (SELECT department\_id FROM departments WHERE department\_name='IT');

# Lab 10

Common Solution**:-**

**Qno1:-**

Write a query to find the names (first\_name, last\_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

SELECT first\_name, last\_name FROM employees

WHERE manager\_id in(select employee\_id FROM employees WHERE department\_id

IN(SELECT department\_id FROM departments WHERE location\_id IN(select location\_id from locations

Where country\_id=’US’)));

**Qno2:-**

Write a query to find the names (first\_name, last\_name) of the employees who are managers.

Answer:-

SELECT first\_name, last\_name

FROM employees

WHERE (employee\_id IN(SELECT manager\_id FROM employees));

**Qno3:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is greater than the average salary

Answer:-

SELECT first\_name, last\_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);

**Qno4:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

SELECT first\_name,last\_name,salary FROM employees WHERE employees.salary=(SELECT min\_salary FROM jobs WHERE employees.job\_id=jobs.job\_id);

**Qno5:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

SELECT first\_name,last\_name,salary

FROM employees WHERE department\_id IN (SELECT department\_id FROM departments WHERE department\_name LIKE ‘IT**%**’)AND salary>(SELECT avg(salary) From employees);

**Qno6:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than Mr. Bell

Answer:-

SELECT \* FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department\_id);

**Qno7:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

SELECT \* FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);

**Qno8:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

SELECT first\_name,last\_name from employees whose(SELECT AVG(salary) from departments)

**Qno9:-**

Write a query to find the names (first\_name, last\_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB\_ID = ’SH\_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

**SELECT first\_name,last\_name, job\_id, salary**

**FROM employees**

**WHERE salary >**

**ALL (SELECT salary FROM employees WHERE job\_id = 'SH\_CLERK') ORDER BY salary;**

**Qno10:-**

.Write a query to find the names (first\_name, last\_name) of the employees who are not supervisors.

**Answer:-**

**SELECT b.first\_name,b.last\_name**

**FROM employees b**

**WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager\_id = b.employee\_id);**

**Qno11:-**

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

**SELECT employee\_id, first\_name, last\_name,**

**(SELECT department\_name FROM departments d**

**WHERE e.department\_id = d.department\_id) department**

**FROM employees e ORDER BY department;**

**Qno12:-**

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

**SELECT employee\_id, first\_name**

**FROM employees AS A**

**WHERE salary >**

**(SELECT AVG(salary) FROM employees WHERE department\_id = A.department\_id);**

**Qno13:-**

Write a query to fetch even numbered records from employees table

Answer:-

**SET @i = 0;**

**SELECT i, employee\_id**

**FROM (SELECT @i := @i + 1 AS i, employee\_id FROM employees)**

**a WHERE MOD(a.i, 2) = 0;**

**Qno14:-**

Write a query to find the 5th maximum salary in the employees table.

Answer:-

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 5 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary >= e1.salary);**

**Qno15:-**

Write a query to find the 4th minimum salary in the employees table

**Answer:-**

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 4 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary <= e1.salary);**

**Qno16:-**

Write a query to select last 10 records from a table.

Answer:-

**SELECT \* FROM (**

**SELECT \* FROM employees ORDER BY employee\_id DESC LIMIT 10) sub**

**ORDER BY employee\_id ASC;**

**Qno17:-**

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

**SELECT \* FROM departments**

**WHERE department\_id**

**NOT IN (select department\_id FROM employees);**

**Qno18:-**

Write a query to get 3 maximum salaries.

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary >= a.salary)**

**ORDER BY a.salary DESC;**

**Qno19:-**

Write a query to get 3 minimum salaries

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary <= a.salary)**

**ORDER BY a.salary DESC;**

**Qno20:-**

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

**SELECT \***

**FROM employees emp1**

**WHERE (1) = (**

**SELECT COUNT(DISTINCT(emp2.salary))**

**FROM employees emp2**

**WHERE e**mp2.salary > emp1**.salary);**

## Individual Solutions:-

**Lab 10**

### Hanzala Shahid

**FA18-BCS-014**

**Qno1:-**

Write a query to find the names (first\_name, last\_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

SELECT first\_name, last\_name FROM employees

WHERE manager\_id in(select employee\_id FROM employees WHERE department\_id

IN(SELECT department\_id FROM departments WHERE location\_id IN(select location\_id from locations

Where country\_id=’US’)));

**Qno2:-**

Write a query to find the names (first\_name, last\_name) of the employees who are managers.

Answer:-

SELECT first\_name, last\_name

FROM employees

WHERE (employee\_id IN(SELECT manager\_id FROM employees));

**Qno3:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is greater than the average salary

Answer:-

SELECT first\_name, last\_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);

**Qno4:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

SELECT first\_name,last\_name,salary FROM employees WHERE employees.salary=(SELECT min\_salary FROM jobs WHERE employees.job\_id=jobs.job\_id);

**Qno5:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

SELECT first\_name,last\_name,salary

FROM employees WHERE department\_id IN (SELECT department\_id FROM departments WHERE department\_name LIKE ‘IT**%**’)AND salary>(SELECT avg(salary) From employees);

**Qno6:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than Mr. Bell

Answer:-

SELECT \* FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department\_id);

**Qno7:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

SELECT \* FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);

**Qno8:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

SELECT first\_name,last\_name from employees whose(SELECT AVG(salary) from departments)

**Qno9:-**

Write a query to find the names (first\_name, last\_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB\_ID = ’SH\_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

**SELECT first\_name,last\_name, job\_id, salary**

**FROM employees**

**WHERE salary >**

**ALL (SELECT salary FROM employees WHERE job\_id = 'SH\_CLERK') ORDER BY salary;**

**Qno10:-**

.Write a query to find the names (first\_name, last\_name) of the employees who are not supervisors.

**Answer:-**

**SELECT b.first\_name,b.last\_name**

**FROM employees b**

**WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager\_id = b.employee\_id);**

**Qno11:-**

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

**SELECT employee\_id, first\_name, last\_name,**

**(SELECT department\_name FROM departments d**

**WHERE e.department\_id = d.department\_id) department**

**FROM employees e ORDER BY department;**

**Qno12:-**

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

**SELECT employee\_id, first\_name**

**FROM employees AS A**

**WHERE salary >**

**(SELECT AVG(salary) FROM employees WHERE department\_id = A.department\_id);**

**Qno13:-**

Write a query to fetch even numbered records from employees table

Answer:-

**SET @i = 0;**

**SELECT i, employee\_id**

**FROM (SELECT @i := @i + 1 AS i, employee\_id FROM employees)**

**a WHERE MOD(a.i, 2) = 0;**

**Qno14:-**

Write a query to find the 5th maximum salary in the employees table.

Answer:-

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 5 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary >= e1.salary);**

**Qno15:-**

Write a query to find the 4th minimum salary in the employees table

**Answer:-**

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 4 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary <= e1.salary);**

**Qno16:-**

Write a query to select last 10 records from a table.

Answer:-

**SELECT \* FROM (**

**SELECT \* FROM employees ORDER BY employee\_id DESC LIMIT 10) sub**

**ORDER BY employee\_id ASC;**

**Qno17:-**

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

**SELECT \* FROM departments**

**WHERE department\_id**

**NOT IN (select department\_id FROM employees);**

**Qno18:-**

Write a query to get 3 maximum salaries.

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary >= a.salary)**

**ORDER BY a.salary DESC;**

**Qno19:-**

Write a query to get 3 minimum salaries

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary <= a.salary)**

**ORDER BY a.salary DESC;**

**Qno20:-**

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

**SELECT \***

**FROM employees emp1**

**WHERE (1) = (**

**SELECT COUNT(DISTINCT(emp2.salary))**

**FROM employees emp2**

**WHERE e**mp2.salary > emp1**.salary);**

### Hamza Bin Ahmed

**FA18-BCS-084**

**Qno1:-**

Write a query to find the names (first\_name, last\_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

SELECT first\_name, last\_name FROM employees

WHERE manager\_id in(select employee\_id FROM employees WHERE department\_id

IN(SELECT department\_id FROM departments WHERE location\_id IN(select location\_id from locations

Where country\_id=’US’)));

**Qno2:-**

Write a query to find the names (first\_name, last\_name) of the employees who are managers.

Answer:-

SELECT first\_name, last\_name

FROM employees

WHERE (employee\_id IN(SELECT manager\_id FROM employees));

**Qno3:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is greater than the average salary

Answer:-

SELECT first\_name, last\_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);

**Qno4:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

SELECT first\_name,last\_name,salary FROM employees WHERE employees.salary=(SELECT min\_salary FROM jobs WHERE employees.job\_id=jobs.job\_id);

**Qno5:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

SELECT first\_name,last\_name,salary

FROM employees WHERE department\_id IN (SELECT department\_id FROM departments WHERE department\_name LIKE ‘IT**%**’)AND salary>(SELECT avg(salary) From employees);

**Qno6:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than Mr. Bell

Answer:-

SELECT \* FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department\_id);

**Qno7:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

SELECT \* FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);

**Qno8:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

SELECT first\_name,last\_name from employees whose(SELECT AVG(salary) from departments)

**Qno9:-**

Write a query to find the names (first\_name, last\_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB\_ID = ’SH\_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

**SELECT first\_name,last\_name, job\_id, salary**

**FROM employees**

**WHERE salary >**

**ALL (SELECT salary FROM employees WHERE job\_id = 'SH\_CLERK') ORDER BY salary;**

**Qno10:-**

.Write a query to find the names (first\_name, last\_name) of the employees who are not supervisors.

**Answer:-**

**SELECT b.first\_name,b.last\_name**

**FROM employees b**

**WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager\_id = b.employee\_id);**

**Qno11:-**

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

**SELECT employee\_id, first\_name, last\_name,**

**(SELECT department\_name FROM departments d**

**WHERE e.department\_id = d.department\_id) department**

**FROM employees e ORDER BY department;**

**Qno12:-**

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

**SELECT employee\_id, first\_name**

**FROM employees AS A**

**WHERE salary >**

**(SELECT AVG(salary) FROM employees WHERE department\_id = A.department\_id);**

**Qno13:-**

Write a query to fetch even numbered records from employees table

Answer:-

**SET @i = 0;**

**SELECT i, employee\_id**

**FROM (SELECT @i := @i + 1 AS i, employee\_id FROM employees)**

**a WHERE MOD(a.i, 2) = 0;**

**Qno14:-**

Write a query to find the 5th maximum salary in the employees table.

Answer:-

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 5 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary >= e1.salary);**

**Qno15:-**

Write a query to find the 4th minimum salary in the employees table

**Answer:-**

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 4 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary <= e1.salary);**

**Qno16:-**

Write a query to select last 10 records from a table.

Answer:-

**SELECT \* FROM (**

**SELECT \* FROM employees ORDER BY employee\_id DESC LIMIT 10) sub**

**ORDER BY employee\_id ASC;**

**Qno17:-**

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

**SELECT \* FROM departments**

**WHERE department\_id**

**NOT IN (select department\_id FROM employees);**

**Qno18:-**

Write a query to get 3 maximum salaries.

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary >= a.salary)**

**ORDER BY a.salary DESC;**

**Qno19:-**

Write a query to get 3 minimum salaries

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary <= a.salary)**

**ORDER BY a.salary DESC;**

**Qno20:-**

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

**SELECT \***

**FROM employees emp1**

**WHERE (1) = (**

**SELECT COUNT(DISTINCT(emp2.salary))**

**FROM employees emp2**

**WHERE e**mp2.salary > emp1**.salary);**

### Muhammad Sharjeel khan

**FA18-BCS-019**

**Qno1:-**

Write a query to find the names (first\_name, last\_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

SELECT first\_name, last\_name FROM employees

WHERE manager\_id in(select employee\_id FROM employees WHERE department\_id

IN(SELECT department\_id FROM departments WHERE location\_id IN(select location\_id from locations

Where country\_id=’US’)));

**Qno2:-**

Write a query to find the names (first\_name, last\_name) of the employees who are managers.

Answer:-

SELECT first\_name, last\_name

FROM employees

WHERE (employee\_id IN(SELECT manager\_id FROM employees));

**Qno3:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is greater than the average salary

Answer:-

SELECT first\_name, last\_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);

**Qno4:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

SELECT first\_name,last\_name,salary FROM employees WHERE employees.salary=(SELECT min\_salary FROM jobs WHERE employees.job\_id=jobs.job\_id);

**Qno5:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

SELECT first\_name,last\_name,salary

FROM employees WHERE department\_id IN (SELECT department\_id FROM departments WHERE department\_name LIKE ‘IT**%**’)AND salary>(SELECT avg(salary) From employees);

**Qno6:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than Mr. Bell

Answer:-

SELECT \* FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department\_id);

**Qno7:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

SELECT \* FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);

**Qno8:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

SELECT first\_name,last\_name from employees whose(SELECT AVG(salary) from departments)

**Qno9:-**

Write a query to find the names (first\_name, last\_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB\_ID = ’SH\_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

**SELECT first\_name,last\_name, job\_id, salary**

**FROM employees**

**WHERE salary >**

**ALL (SELECT salary FROM employees WHERE job\_id = 'SH\_CLERK') ORDER BY salary;**

**Qno10:-**

.Write a query to find the names (first\_name, last\_name) of the employees who are not supervisors.

**Answer:-**

**SELECT b.first\_name,b.last\_name**

**FROM employees b**

**WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager\_id = b.employee\_id);**

**Qno11:-**

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

**SELECT employee\_id, first\_name, last\_name,**

**(SELECT department\_name FROM departments d**

**WHERE e.department\_id = d.department\_id) department**

**FROM employees e ORDER BY department;**

**Qno12:-**

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

**SELECT employee\_id, first\_name**

**FROM employees AS A**

**WHERE salary >**

**(SELECT AVG(salary) FROM employees WHERE department\_id = A.department\_id);**

**Qno13:-**

Write a query to fetch even numbered records from employees table

Answer:-

**SET @i = 0;**

**SELECT i, employee\_id**

**FROM (SELECT @i := @i + 1 AS i, employee\_id FROM employees)**

**a WHERE MOD(a.i, 2) = 0;**

**Qno14:-**

Write a query to find the 5th maximum salary in the employees table.

Answer:-

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 5 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary >= e1.salary);**

**Qno15:-**

Write a query to find the 4th minimum salary in the employees table

**Answer:-**

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 4 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary <= e1.salary);**

**Qno16:-**

Write a query to select last 10 records from a table.

Answer:-

**SELECT \* FROM (**

**SELECT \* FROM employees ORDER BY employee\_id DESC LIMIT 10) sub**

**ORDER BY employee\_id ASC;**

**Qno17:-**

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

**SELECT \* FROM departments**

**WHERE department\_id**

**NOT IN (select department\_id FROM employees);**

**Qno18:-**

Write a query to get 3 maximum salaries.

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary >= a.salary)**

**ORDER BY a.salary DESC;**

**Qno19:-**

Write a query to get 3 minimum salaries

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary <= a.salary)**

**ORDER BY a.salary DESC;**

**Qno20:-**

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

**SELECT \***

**FROM employees emp1**

**WHERE (1) = (**

**SELECT COUNT(DISTINCT(emp2.salary))**

**FROM employees emp2**

**WHERE e**mp2.salary > emp1**.salary);**

### Hamza Aslam

**FA18-BCS-050**

**Qno1:-**

Write a query to find the names (first\_name, last\_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

SELECT first\_name, last\_name FROM employees

WHERE manager\_id in(select employee\_id FROM employees WHERE department\_id

IN(SELECT department\_id FROM departments WHERE location\_id IN(select location\_id from locations

Where country\_id=’US’)));

**Qno2:-**

Write a query to find the names (first\_name, last\_name) of the employees who are managers.

Answer:-

SELECT first\_name, last\_name

FROM employees

WHERE (employee\_id IN(SELECT manager\_id FROM employees));

**Qno3:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is greater than the average salary

Answer:-

SELECT first\_name, last\_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);

**Qno4:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

SELECT first\_name,last\_name,salary FROM employees WHERE employees.salary=(SELECT min\_salary FROM jobs WHERE employees.job\_id=jobs.job\_id);

**Qno5:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

SELECT first\_name,last\_name,salary

FROM employees WHERE department\_id IN (SELECT department\_id FROM departments WHERE department\_name LIKE ‘IT**%**’)AND salary>(SELECT avg(salary) From employees);

**Qno6:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than Mr. Bell

Answer:-

SELECT \* FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department\_id);

**Qno7:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

SELECT \* FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);

**Qno8:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

SELECT first\_name,last\_name from employees whose(SELECT AVG(salary) from departments)

**Qno9:-**

Write a query to find the names (first\_name, last\_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB\_ID = ’SH\_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

**SELECT first\_name,last\_name, job\_id, salary**

**FROM employees**

**WHERE salary >**

**ALL (SELECT salary FROM employees WHERE job\_id = 'SH\_CLERK') ORDER BY salary;**

**Qno10:-**

.Write a query to find the names (first\_name, last\_name) of the employees who are not supervisors.

**Answer:-**

**SELECT b.first\_name,b.last\_name**

**FROM employees b**

**WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager\_id = b.employee\_id);**

**Qno11:-**

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

**SELECT employee\_id, first\_name, last\_name,**

**(SELECT department\_name FROM departments d**

**WHERE e.department\_id = d.department\_id) department**

**FROM employees e ORDER BY department;**

**Qno12:-**

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

**SELECT employee\_id, first\_name**

**FROM employees AS A**

**WHERE salary >**

**(SELECT AVG(salary) FROM employees WHERE department\_id = A.department\_id);**

**Qno13:-**

Write a query to fetch even numbered records from employees table

Answer:-

**SET @i = 0;**

**SELECT i, employee\_id**

**FROM (SELECT @i := @i + 1 AS i, employee\_id FROM employees)**

**a WHERE MOD(a.i, 2) = 0;**

**Qno14:-**

Write a query to find the 5th maximum salary in the employees table.

Answer:-

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 5 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary >= e1.salary);**

**Qno15:-**

Write a query to find the 4th minimum salary in the employees table

**Answer:-**

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 4 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary <= e1.salary);**

**Qno16:-**

Write a query to select last 10 records from a table.

Answer:-

**SELECT \* FROM (**

**SELECT \* FROM employees ORDER BY employee\_id DESC LIMIT 10) sub**

**ORDER BY employee\_id ASC;**

**Qno17:-**

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

**SELECT \* FROM departments**

**WHERE department\_id**

**NOT IN (select department\_id FROM employees);**

**Qno18:-**

Write a query to get 3 maximum salaries.

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary >= a.salary)**

**ORDER BY a.salary DESC;**

**Qno19:-**

Write a query to get 3 minimum salaries

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary <= a.salary)**

**ORDER BY a.salary DESC;**

**Qno20:-**

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

**SELECT \***

**FROM employees emp1**

**WHERE (1) = (**

**SELECT COUNT(DISTINCT(emp2.salary))**

**FROM employees emp2**

**WHERE e**mp2.salary > emp1**.salary);**

### Abdullah Noor Niazi

**FA18-BCS-004**

Qno1:-

Write a query to find the names (first\_name, last\_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

SELECT first\_name, last\_name FROM employees

WHERE manager\_id in(select employee\_id FROM employees WHERE department\_id

IN(SELECT department\_id FROM departments WHERE location\_id IN(select location\_id from locations

Where country\_id=’US’)));

Qno2:-

Write a query to find the names (first\_name, last\_name) of the employees who are managers.

Answer:-

SELECT first\_name, last\_name

FROM employees

WHERE (employee\_id IN(SELECT manager\_id FROM employees));

Qno3:-

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is greater than the average salary

Answer:-

SELECT first\_name, last\_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);

Qno4:-

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

SELECT first\_name,last\_name,salary FROM employees WHERE employees.salary=(SELECT min\_salary FROM jobs WHERE employees.job\_id=jobs.job\_id);

Qno5:-

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

SELECT first\_name,last\_name,salary

FROM employees WHERE department\_id IN (SELECT department\_id FROM departments WHERE department\_name LIKE ‘IT%’)AND salary>(SELECT avg(salary) From employees);

Qno6:-

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than Mr. Bell

Answer:-

SELECT \* FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department\_id);

Qno7:-

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

SELECT \* FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);

Qno8:-

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

SELECT first\_name,last\_name from employees whose(SELECT AVG(salary) from departments)

Qno9:-

Write a query to find the names (first\_name, last\_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB\_ID = ’SH\_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

SELECT first\_name,last\_name, job\_id, salary

FROM employees

WHERE salary >

ALL (SELECT salary FROM employees WHERE job\_id = 'SH\_CLERK') ORDER BY salary;

Qno10:-

.Write a query to find the names (first\_name, last\_name) of the employees who are not supervisors.

Answer:-

SELECT b.first\_name,b.last\_name

FROM employees b

WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager\_id = b.employee\_id);

Qno11:-

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

SELECT employee\_id, first\_name, last\_name,

(SELECT department\_name FROM departments d

WHERE e.department\_id = d.department\_id) department

FROM employees e ORDER BY department;

Qno12:-

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

SELECT employee\_id, first\_name

FROM employees AS A

WHERE salary >

(SELECT AVG(salary) FROM employees WHERE department\_id = A.department\_id);

Qno13:-

Write a query to fetch even numbered records from employees table

Answer:-

SET @i = 0;

SELECT i, employee\_id

FROM (SELECT @i := @i + 1 AS i, employee\_id FROM employees)

a WHERE MOD(a.i, 2) = 0;

Qno14:-

Write a query to find the 5th maximum salary in the employees table.

Answer:-

SELECT DISTINCT salary

FROM employees e1

WHERE 5 = (SELECT COUNT(DISTINCT salary)

FROM employees e2

WHERE e2.salary >= e1.salary);

Qno15:-

Write a query to find the 4th minimum salary in the employees table

Answer:-

SELECT DISTINCT salary

FROM employees e1

WHERE 4 = (SELECT COUNT(DISTINCT salary)

FROM employees e2

WHERE e2.salary <= e1.salary);

Qno16:-

Write a query to select last 10 records from a table.

Answer:-

SELECT \* FROM (

SELECT \* FROM employees ORDER BY employee\_id DESC LIMIT 10) sub

ORDER BY employee\_id ASC;

Qno17:-

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

SELECT \* FROM departments

WHERE department\_id

NOT IN (select department\_id FROM employees);

Qno18:-

Write a query to get 3 maximum salaries.

Answer:-

SELECT DISTINCT salary

FROM employees a

WHERE 3 >= (SELECT COUNT(DISTINCT salary)

FROM employees b

WHERE b.salary >= a.salary)

ORDER BY a.salary DESC;

Qno19:-

Write a query to get 3 minimum salaries

Answer:-

SELECT DISTINCT salary

FROM employees a

WHERE 3 >= (SELECT COUNT(DISTINCT salary)

FROM employees b

WHERE b.salary <= a.salary)

ORDER BY a.salary DESC;

Qno20:-

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

SELECT \*

FROM employees emp1

WHERE (1) = (

SELECT COUNT(DISTINCT(emp2.salary))

FROM employees emp2

WHERE emp2.salary > emp1.salary);

### Usman Jadoon

**FA18-BCS-100**

**Qno1:-**

Write a query to find the names (first\_name, last\_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

SELECT first\_name, last\_name FROM employees

WHERE manager\_id in(select employee\_id FROM employees WHERE department\_id

IN(SELECT department\_id FROM departments WHERE location\_id IN(select location\_id from locations

Where country\_id=’US’)));

**Qno2:-**

Write a query to find the names (first\_name, last\_name) of the employees who are managers.

Answer:-

SELECT first\_name, last\_name

FROM employees

WHERE (employee\_id IN(SELECT manager\_id FROM employees));

**Qno3:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is greater than the average salary

Answer:-

SELECT first\_name, last\_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);

**Qno4:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

SELECT first\_name,last\_name,salary FROM employees WHERE employees.salary=(SELECT min\_salary FROM jobs WHERE employees.job\_id=jobs.job\_id);

**Qno5:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

SELECT first\_name,last\_name,salary

FROM employees WHERE department\_id IN (SELECT department\_id FROM departments WHERE department\_name LIKE ‘IT**%**’)AND salary>(SELECT avg(salary) From employees);

**Qno6:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than Mr. Bell

Answer:-

SELECT \* FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department\_id);

**Qno7:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

SELECT \* FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);

**Qno8:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

SELECT first\_name,last\_name from employees whose(SELECT AVG(salary) from departments)

**Qno9:-**

Write a query to find the names (first\_name, last\_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB\_ID = ’SH\_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

**SELECT first\_name,last\_name, job\_id, salary**

**FROM employees**

**WHERE salary >**

**ALL (SELECT salary FROM employees WHERE job\_id = 'SH\_CLERK') ORDER BY salary;**

**Qno10:-**

.Write a query to find the names (first\_name, last\_name) of the employees who are not supervisors.

**Answer:-**

**SELECT b.first\_name,b.last\_name**

**FROM employees b**

**WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager\_id = b.employee\_id);**

**Qno11:-**

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

**SELECT employee\_id, first\_name, last\_name,**

**(SELECT department\_name FROM departments d**

**WHERE e.department\_id = d.department\_id) department**

**FROM employees e ORDER BY department;**

**Qno12:-**

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

**SELECT employee\_id, first\_name**

**FROM employees AS A**

**WHERE salary >**

**(SELECT AVG(salary) FROM employees WHERE department\_id = A.department\_id);**

**Qno13:-**

Write a query to fetch even numbered records from employees table

Answer:-

**SET @i = 0;**

**SELECT i, employee\_id**

**FROM (SELECT @i := @i + 1 AS i, employee\_id FROM employees)**

**a WHERE MOD(a.i, 2) = 0;**

**Qno14:-**

Write a query to find the 5th maximum salary in the employees table.

Answer:-

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 5 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary >= e1.salary);**

**Qno15:-**

Write a query to find the 4th minimum salary in the employees table

**Answer:-**

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 4 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary <= e1.salary);**

**Qno16:-**

Write a query to select last 10 records from a table.

Answer:-

**SELECT \* FROM (**

**SELECT \* FROM employees ORDER BY employee\_id DESC LIMIT 10) sub**

**ORDER BY employee\_id ASC;**

**Qno17:-**

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

**SELECT \* FROM departments**

**WHERE department\_id**

**NOT IN (select department\_id FROM employees);**

**Qno18:-**

Write a query to get 3 maximum salaries.

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary >= a.salary)**

**ORDER BY a.salary DESC;**

**Qno19:-**

Write a query to get 3 minimum salaries

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary <= a.salary)**

**ORDER BY a.salary DESC;**

**Qno20:-**

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

**SELECT \***

**FROM employees emp1**

**WHERE (1) = (**

**SELECT COUNT(DISTINCT(emp2.salary))**

**FROM employees emp2**

**WHERE e**mp2.salary > emp1**.salary);**

### Usama Fareed

**FA18-BCS-026**

**Qno1:-**

Write a query to find the names (first\_name, last\_name) of the employees who have a manager and work for a department based in the United States.

Answer:-

SELECT first\_name, last\_name FROM employees

WHERE manager\_id in(select employee\_id FROM employees WHERE department\_id

IN(SELECT department\_id FROM departments WHERE location\_id IN(select location\_id from locations

Where country\_id=’US’)));

**Qno2:-**

Write a query to find the names (first\_name, last\_name) of the employees who are managers.

Answer:-

SELECT first\_name, last\_name

FROM employees

WHERE (employee\_id IN(SELECT manager\_id FROM employees));

**Qno3:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is greater than the average salary

Answer:-

SELECT first\_name, last\_name, salary FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);

**Qno4:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary is equal to the minimum salary for their job grade.

Answer:-

SELECT first\_name,last\_name,salary FROM employees WHERE employees.salary=(SELECT min\_salary FROM jobs WHERE employees.job\_id=jobs.job\_id);

**Qno5:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than the average salary and who works in any of the IT departments.

Answer:-

SELECT first\_name,last\_name,salary

FROM employees WHERE department\_id IN (SELECT department\_id FROM departments WHERE department\_name LIKE ‘IT**%**’)AND salary>(SELECT avg(salary) From employees);

**Qno6:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn more than Mr. Bell

Answer:-

SELECT \* FROM employees WHERE salary > ALL(SELECT AVG(salary) FROM employees GROUP BY

Department\_id);

**Qno7:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees who earn the same salary as the minimum salary for all departments

Answer:-

SELECT \* FROM employees

WHERE salary=(SELECT MIN(salary) FROM employees);

**Qno8:-**

Write a query to find the names (first\_name, last\_name), the salary of the employees whose salary greater than the average salary of all departments

Answer:-

SELECT first\_name,last\_name from employees whose(SELECT AVG(salary) from departments)

**Qno9:-**

Write a query to find the names (first\_name, last\_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB\_ID = ’SH\_CLERK’). Sort the results of the salary of the lowest to highest

Answer:-

**SELECT first\_name,last\_name, job\_id, salary**

**FROM employees**

**WHERE salary >**

**ALL (SELECT salary FROM employees WHERE job\_id = 'SH\_CLERK') ORDER BY salary;**

**Qno10:-**

.Write a query to find the names (first\_name, last\_name) of the employees who are not supervisors.

**Answer:-**

**SELECT b.first\_name,b.last\_name**

**FROM employees b**

**WHERE NOT EXISTS (SELECT 'X' FROM employees a WHERE a.manager\_id = b.employee\_id);**

**Qno11:-**

Write a query to display the employee ID, first name, last names, and department names of all employees.

Answer:-

**SELECT employee\_id, first\_name, last\_name,**

**(SELECT department\_name FROM departments d**

**WHERE e.department\_id = d.department\_id) department**

**FROM employees e ORDER BY department;**

**Qno12:-**

Write a query to display the employee ID, first name, last names, salary of all employees whose salary is above average for their departments

Answer:-

**SELECT employee\_id, first\_name**

**FROM employees AS A**

**WHERE salary >**

**(SELECT AVG(salary) FROM employees WHERE department\_id = A.department\_id);**

**Qno13:-**

Write a query to fetch even numbered records from employees table

Answer:-

**SET @i = 0;**

**SELECT i, employee\_id**

**FROM (SELECT @i := @i + 1 AS i, employee\_id FROM employees)**

**a WHERE MOD(a.i, 2) = 0;**

**Qno14:-**

Write a query to find the 5th maximum salary in the employees table.

Answer:-

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 5 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary >= e1.salary);**

**Qno15:-**

Write a query to find the 4th minimum salary in the employees table

**Answer:-**

**SELECT DISTINCT salary**

**FROM employees e1**

**WHERE 4 = (SELECT COUNT(DISTINCT salary)**

**FROM employees e2**

**WHERE e2.salary <= e1.salary);**

**Qno16:-**

Write a query to select last 10 records from a table.

Answer:-

**SELECT \* FROM (**

**SELECT \* FROM employees ORDER BY employee\_id DESC LIMIT 10) sub**

**ORDER BY employee\_id ASC;**

**Qno17:-**

Write a query to list department number, name for all the departments in which there are no employees in the department

Answer:-

**SELECT \* FROM departments**

**WHERE department\_id**

**NOT IN (select department\_id FROM employees);**

**Qno18:-**

Write a query to get 3 maximum salaries.

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary >= a.salary)**

**ORDER BY a.salary DESC;**

**Qno19:-**

Write a query to get 3 minimum salaries

Answer:-

**SELECT DISTINCT salary**

**FROM employees a**

**WHERE 3 >= (SELECT COUNT(DISTINCT salary)**

**FROM employees b**

**WHERE b.salary <= a.salary)**

**ORDER BY a.salary DESC;**

**Qno20:-**

Write a query to get nth max salaries of employees.Further practice with nested queries

Answer:-

**SELECT \***

**FROM employees emp1**

**WHERE (1) = (**

**SELECT COUNT(DISTINCT(emp2.salary))**

**FROM employees emp2**

**WHERE e**mp2.salary > emp1**.salary);**

# Lab 11

Common Solution**:-**

**Qno1:-**

Create a table tow columns for name and family\_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

SELECT customer\_id, CONCAT(first\_name,second\_name,last\_name)AS All\_names from customer

# Individual Solutions:-

**Lab 11**

### Hanzala Shahid

**FA18-BCS-014**

**Qno1:-**

Create a table tow columns for name and family\_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

SELECT customer\_id, CONCAT(first\_name,second\_name,last\_name)AS All\_names from customer

### Hamza Bin Ahmed

**FA18-BCS-084**

**Qno1:-**

Create a table tow columns for name and family\_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

SELECT customer\_id, CONCAT(first\_name,second\_name,last\_name)AS All\_names from customer

### Muhammad Sharjeel khan

**FA18-BCS-019**

**Qno1:-**

Create a table tow columns for name and family\_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

SELECT customer\_id, CONCAT(first\_name,second\_name,last\_name)AS All\_names from customer

### Hamza Aslam

**FA18-BCS-050**

**Qno1:-**

Create a table tow columns for name and family\_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

SELECT customer\_id, CONCAT(first\_name,second\_name,last\_name)AS All\_names from customer

### Abdullah Noor Niazi

**FA18-BCS-004**

**Qno1:-**

Create a table tow columns for name and family\_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

SELECT customer\_id, CONCAT(first\_name,second\_name,last\_name)AS All\_names from customer

### Usman Jadoon

**FA18-BCS-100**

**Qno1:-**

Create a table tow columns for name and family\_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

SELECT customer\_id, CONCAT(first\_name,second\_name,last\_name)AS All\_names from customer

### Usama Fareed

**FA18-BCS-026**

**Qno1:-**

Create a table tow columns for name and family\_name respectively. Insert the names your three friends in lower case case caracters. Write a query to create columns aliased fullname by using the INITCAT() and CONCAT() functions.

Answer:-

SELECT customer\_id, CONCAT(first\_name,second\_name,last\_name)AS All\_names from customer

# Lab 12

Common Solution**:-**

**Qno1:-**

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

**Qno2:-**

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

**Qno3:-**

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

**Qno4:-**

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

## Individual Solutions:-

**Lab 12**

### Hanzala Shahid

**FA18-BCS-014**

**Qno1:-**

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

**Qno2:-**

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

**Qno3:-**

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

**Qno4:-**

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

### Hamza Bin Ahmed

**FA18-BCS-084**

**Qno1:-**

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

**Qno2:-**

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

**Qno3:-**

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

**Qno4:-**

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

### Muhammad Sharjeel khan

**FA18-BCS-019**

**Qno1:-**

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

**Qno2:-**

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

**Qno3:-**

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

**Qno4:-**

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

### Hamza Aslam

**FA18-BCS-050**

**Qno1:-**

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

**Qno2:-**

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

**Qno3:-**

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

**Qno4:-**

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

### Abdullah Noor Niazi

**FA18-BCS-004**

**Qno1:-**

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

**Qno2:-**

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

**Qno3:-**

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

**Qno4:-**

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

### Usman Jadoon

**FA18-BCS-100**

**Qno1:-**

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

**Qno2:-**

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

**Qno3:-**

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

**Qno4:-**

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

### Usama Fareed

**FA18-BCS-026**

**Qno1:-**

Print countrycode and sum of percentage from countrylangauge, apply groupby on countrycode.

Answer:-

SELECT countrycode ,SUM(PER(countrylanguage) from country language Group By(countrycode)

**Qno2:-**

Find sum of any integer column from country table.

Answer:-

Select SUM(population) from country table;

**Qno3:-**

Count number of records in country table

Answer:-

SELECT COUNT(records) from countrytable;

**Qno4:-**

Count Distinct (languages) from countrylanguage

Answer:-

SELECT DISTINCT(languages) from countrylanguage;

# Lab 13

Common Solution**:-**

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven’t include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

## Individual Solutions:-

**Lab 13**

### Hanzala Shahid

**FA18-BCS-014**

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven’t include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

### Hamza Bin Ahmed

**FA18-BCS-084**

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven’t include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

### Muhammad Sharjeel khan

**FA18-BCS-019**

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven’t include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

### Hamza Aslam

**FA18-BCS-050**

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven’t include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

### Abdullah Noor Niazi

**FA18-BCS-004**

Qno1:-

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven’t include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

### Usman Jadoon

**FA18-BCS-100**

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven’t include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

### Usama Fareed

**FA18-BCS-026**

Select customers name, number , phone from customers table, select checknumber from payments table. Display it for all customers. [either they have made payment or they haven’t include all customers].

Answer:-

SELECT customer,phone from customertable;

SELECT checknumber from payment;

# Lab 14

Common Solution**:-**

**Qno1:-**

Update customer with any a particular order number (you can select any order number).

UPDATE *customer*  
SET *order\_no*=*5*;

**Qno2:-**

Applying union print data of orders and order details table.

**Answer:-**

SELECT *data.customerdata from customers*  
UNION ALL  
SELECT *order.orderdata* FROM *orders*;

## Individual Solutions:-

**Lab 14**

### Hanzala Shahid

**FA18-BCS-014**

**Qno1:-**

Update customer with any a particular order number (you can select any order number).

UPDATE *customer*  
SET *order\_no*=*14*;

**Qno2:-**

Applying union print data of orders and order details table.

**Answer:-**

SELECT *data.customerdata from customers*  
UNION ALL  
SELECT *order.orderdata* FROM *orders*;

### Hamza Bin Ahmed

**FA18-BCS-084**

**Qno1:-**

Update customer with any a particular order number (you can select any order number).

UPDATE *customer*  
SET *order\_no*=*3*;

**Qno2:-**

Applying union print data of orders and order details table.

**Answer:-**

SELECT *data.customerdata from customers*  
UNION ALL  
SELECT *order.orderdata* FROM *orders*;

### Muhammad Sharjeel khan

**FA18-BCS-019**

**Qno1:-**

Update customer with any a particular order number (you can select any order number).

UPDATE *customer*  
SET *order\_no*=*9*;

**Qno2:-**

Applying union print data of orders and order details table.

**Answer:-**

SELECT *data.customerdata from customers*  
UNION ALL  
SELECT *order.orderdata* FROM *orders*;

### Hamza Aslam

**FA18-BCS-050**

**Qno1:-**

Update customer with any a particular order number (you can select any order number).

UPDATE *customer*  
SET *order\_no*=*7*;

**Qno2:-**

Applying union print data of orders and order details table.

**Answer:-**

SELECT *data.customerdata from customers*  
UNION ALL  
SELECT *order.orderdata* FROM *orders*;

### Abdullah Noor Niazi

**FA18-BCS-004**

**Qno1:-**

Update customer with any a particular order number (you can select any order number).

UPDATE *customer*  
SET *order\_no*=*2*;

**Qno2:-**

Applying union print data of orders and order details table.

**Answer:-**

SELECT *data.customerdata from customers*  
UNION ALL  
SELECT *order.orderdata* FROM *orders*;

### Usman Jadoon

**FA18-BCS-100**

**Qno1:-**

Update customer with any a particular order number (you can select any order number).

UPDATE *customer*  
SET *order\_no*=*17*;

**Qno2:-**

Applying union print data of orders and order details table.

**Answer:-**

SELECT *data.customerdata from customers*  
UNION ALL  
SELECT *order.orderdata* FROM *orders*;

### Usama Fareed

**FA18-BCS-026**

**Qno1:-**

Update customer with any a particular order number (you can select any order number).

UPDATE *customer*  
SET *order\_no*=*13*;

**Qno2:-**

Applying union print data of orders and order details table.

**Answer:-**

SELECT *data.customerdata from customers*  
UNION ALL  
SELECT *order.orderdata* FROM *orders*;

# Lab 15

Common Solution**:-**

**Qno1:-**

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

**Qno2:-**

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

DELETE FROM city

WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);

**Qno3:-**

Try to update values for null column COMM column

**Answer:-**

**UPDATE[city]**

**SET [COMM]=0**

**WHERE [COMM] is null;**

**Qno4:-**

Try to update it for a specific employee whose salary is less than 1000

Answer:-

UPDATE employee WHERE employee<1000;

**Qno5:-**

Answer:-

INSERT INTO *city*  
VALUES (*“Abbottabad”*,*“Haripur”*,*“Mansehra”*);

## Individual Solutions:-

**Lab 15**

### Hanzala Shahid

**FA18-BCS-014**

**Qno1:-**

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

**Qno2:-**

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

DELETE FROM city

WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);

**Qno3:-**

Try to update values for null column COMM column

**Answer:-**

**UPDATE[city]**

**SET [COMM]=0**

**WHERE [COMM] is null;**

**Qno4:-**

Try to update it for a specific employee whose salary is less than 1000

Answer:-

UPDATE employee WHERE employee<1000;

**Qno5:-**

Answer:-

INSERT INTO *city*  
VALUES (*“Abbottabad”*,*“Haripur”*,*“Mansehra”*);

### Hamza Bin Ahmed

**FA18-BCS-084**

**Qno1:-**

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

**Qno2:-**

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

DELETE FROM city

WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);

**Qno3:-**

Try to update values for null column COMM column

**Answer:-**

**UPDATE[city]**

**SET [COMM]=0**

**WHERE [COMM] is null;**

**Qno4:-**

Try to update it for a specific employee whose salary is less than 1000

Answer:-

UPDATE employee WHERE employee<1000;

**Qno5:-**

Answer:-

INSERT INTO *city*  
VALUES (*“Lahore”*,*“Bahria”*,*“Karachi”*);

### Muhammad Sharjeel khan

**FA18-BCS-019**

**Qno1:-**

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

**Qno2:-**

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

DELETE FROM city

WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);

**Qno3:-**

Try to update values for null column COMM column

**Answer:-**

**UPDATE[city]**

**SET [COMM]=0**

**WHERE [COMM] is null;**

**Qno4:-**

Try to update it for a specific employee whose salary is less than 1000

Answer:-

UPDATE employee WHERE employee<1000;

**Qno5:-**

Answer:-

INSERT INTO city  
VALUES (“Rawalpindi”, “Sialkot”, “Mirpur”);

### Hamza Aslam

**FA18-BCS-050**

**Qno1:-**

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

**Qno2:-**

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

DELETE FROM city

WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);

**Qno3:-**

Try to update values for null column COMM column

**Answer:-**

**UPDATE[city]**

**SET [COMM]=0**

**WHERE [COMM] is null;**

**Qno4:-**

Try to update it for a specific employee whose salary is less than 1000

Answer:-

UPDATE employee WHERE employee<1000;

**Qno5:-**

Answer:-

INSERT INTO *city*  
VALUES (*“Mian Channu”*,*“Chakwal”*,*“Abbottabad”*);

### Abdullah Noor Niazi

**FA18-BCS-004**

Qno1:-

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

Qno2:-

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

DELETE FROM city

WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);

Qno3:-

Try to update values for null column COMM column

Answer:-

UPDATE[city]

SET [COMM]=0

WHERE [COMM] is null;

Qno4:-

Try to update it for a specific employee whose salary is less than 1000

Answer:-

UPDATE employee WHERE employee<1000;

Qno5:-

Try to insert values in customers table

Answer:-

INSERT INTO city  
VALUES (“Rawalpindi”, “Sialkot”, “Mirpur”);

### Usman Jadoon

**FA18-BCS-100**

**Qno1:-**

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

**Qno2:-**

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

DELETE FROM city

WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);

**Qno3:-**

Try to update values for null column COMM column

**Answer:-**

**UPDATE[city]**

**SET [COMM]=0**

**WHERE [COMM] is null;**

**Qno4:-**

Try to update it for a specific employee whose salary is less than 1000

Answer:-

UPDATE employee WHERE employee<1000;

**Qno5:-**

Answer:-

INSERT INTO city  
VALUES (“Gujrat”, “Gujranwala”, “lahore”);

### Usama Fareed

**FA18-BCS-026**

**Qno1:-**

Apply update on any country name.

Answer:-

UPDATE country SET column1=”England” WHERE column=”Islamabad”;

**Qno2:-**

Delete Islamabad city by applying delete query on city table with it’s ID

Answer:-

DELETE FROM city

WHERE country.id = ANY (SELECT id FROM city WHERE id = 2);

**Qno3:-**

Try to update values for null column COMM column

**Answer:-**

**UPDATE[city]**

**SET [COMM]=0**

**WHERE [COMM] is null;**

**Qno4:-**

Try to update it for a specific employee whose salary is less than 1000

Answer:-

UPDATE employee WHERE employee<1000;

**Qno5:-**

Answer:-

INSERT INTO city  
VALUES (“Karachi”, “Lahore”, “pindi”);

# Lab 16

Common Solution**:-**

**Qno1:-**

The first column is called supplier\_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL);

**Qno2:-**

The second column is called supplier\_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar2(50) NOT NULL);

**Qno3:-**

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL);

**Qno4:-**

Define the supplier\_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier\_id NOT NULL AUTO\_INCREMENT,supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier\_id);

**Qno5:-**

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

**Qno6:-**

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

**Qno7:-**

The second column supplierId as foreignkey in item table

Answer:-

CREATE TABLE item (  
    itemname varchar(255),  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

**Qno8:-**

The third column should be itemprice In INT

Answer:-

CREATE TABLE item (  
    itemname varchar(255),

Itemprice int NOT NULL,  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

## Individual Solutions:-

**Lab 16**

### Hanzala Shahid

**FA18-BCS-014**

**Qno1:-**

The first column is called supplier\_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL);

**Qno2:-**

The second column is called supplier\_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar2(50) NOT NULL);

**Qno3:-**

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL);

**Qno4:-**

Define the supplier\_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier\_id NOT NULL AUTO\_INCREMENT,supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier\_id);

**Qno5:-**

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

**Qno6:-**

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

**Qno7:-**

The second column supplierId as foreignkey in item table

Answer:-

CREATE TABLE item (  
    itemname varchar(255),  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

**Qno8:-**

The third column should be itemprice In INT

Answer:-

CREATE TABLE item (  
    itemname varchar(255),

Itemprice int NOT NULL,  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

### Hamza Bin Ahmed

**FA18-BCS-084**

**Qno1:-**

The first column is called supplier\_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL);

**Qno2:-**

The second column is called supplier\_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar2(50) NOT NULL);

**Qno3:-**

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL);

**Qno4:-**

Define the supplier\_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier\_id NOT NULL AUTO\_INCREMENT,supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier\_id);

**Qno5:-**

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

**Qno6:-**

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

**Qno7:-**

The second column supplierId as foreignkey in item table

Answer:-

CREATE TABLE item (  
    itemname varchar(255),  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

**Qno8:-**

The third column should be itemprice In INT

Answer:-

CREATE TABLE item (  
    itemname varchar(255),

Itemprice int NOT NULL,  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

### Muhammad Sharjeel khan

**FA18-BCS-019**

### Hamza Aslam

**FA18-BCS-050**

**Qno1:-**

The first column is called supplier\_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL);

**Qno2:-**

The second column is called supplier\_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar2(50) NOT NULL);

**Qno3:-**

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL);

**Qno4:-**

Define the supplier\_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier\_id NOT NULL AUTO\_INCREMENT,supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier\_id);

**Qno5:-**

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

**Qno6:-**

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

**Qno7:-**

The second column supplierId as foreignkey in item table

Answer:-

CREATE TABLE item (  
    itemname varchar(255),  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

**Qno8:-**

The third column should be itemprice In INT

Answer:-

CREATE TABLE item (  
    itemname varchar(255),

Itemprice int NOT NULL,  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

### Abdullah Noor Niazi

**FA18-BCS-004**

**Qno1:-**

The first column is called supplier\_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL);

**Qno2:-**

The second column is called supplier\_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar2(50) NOT NULL);

**Qno3:-**

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL);

**Qno4:-**

Define the supplier\_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier\_id NOT NULL AUTO\_INCREMENT,supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier\_id);

**Qno5:-**

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

**Qno6:-**

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

**Qno7:-**

The second column supplierId as foreignkey in item table

Answer:-

CREATE TABLE item (  
    itemname varchar(255),  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

**Qno8:-**

The third column should be itemprice In INT

Answer:-

CREATE TABLE item (  
    itemname varchar(255),

Itemprice int NOT NULL,  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

### Usman Jadoon

**FA18-BCS-100**

**Qno1:-**

The first column is called supplier\_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL);

**Qno2:-**

The second column is called supplier\_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar2(50) NOT NULL);

**Qno3:-**

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL);

**Qno4:-**

Define the supplier\_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier\_id NOT NULL AUTO\_INCREMENT,supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier\_id);

**Qno5:-**

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

**Qno6:-**

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

**Qno7:-**

The second column supplierId as foreignkey in item table

Answer:-

CREATE TABLE item (  
    itemname varchar(255),  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

**Qno8:-**

The third column should be itemprice In INT

Answer:-

CREATE TABLE item (  
    itemname varchar(255),

Itemprice int NOT NULL,  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

### Usama Fareed

**FA18-BCS-026**

**Qno1:-**

The first column is called supplier\_id which is created as a number data type (maximum 10 digits in length) and cannot contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL);

**Qno2:-**

The second column is called supplier\_name which is a varchar2 datatype (50 maximum characters in length) and also can not contain null values

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar2(50) NOT NULL);

**Qno3:-**

The third column is called address which is a varchar2 data type but can contain null values.

Answer:-

CREATE TABLE suppliers(supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL);

**Qno4:-**

Define the supplier\_id as the primary key

Answer:-

CREATE TABLE suppliers(supplier\_id NOT NULL AUTO\_INCREMENT,supplier\_id int(10) NOT NULL,supplier\_name varchar(50) NOT NULL,address varchar2 NOT NULL,PRIMARY KEY(supplier\_id);

**Qno5:-**

Create a second table named as Item with columns:

Answer:-

CREATE TABLE Item();

**Qno6:-**

The first column itemname any length you want

Answer:-

CREATE TABLE Item(itemname varchar(255);

**Qno7:-**

The second column supplierId as foreignkey in item table

Answer:-

CREATE TABLE item (  
    itemname varchar(255),  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

**Qno8:-**

The third column should be itemprice In INT

Answer:-

CREATE TABLE item (  
    itemname varchar(255),

Itemprice int NOT NULL,  
    FOREIGN KEY (supplier\_id) REFERENCES Persons(supplier\_id)  
);

# Lab 20

Common Solution**:-**

**Qno1:-**

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

DECLARE

temp1 NUMBER := &input\_a\_temp;

t\_scale CHAR := '&input\_temp\_scale';

new\_temp NUMBER;

new\_scale CHAR;

BEGIN

IF t\_scale != 'C'

AND

t\_scale != 'F' THEN

dbms\_output.Put\_line ('The scale you input is not a valid scale');

new\_temp := 0;

new\_scale := 'C';

ELSE

IF t\_scale = 'C' THEN

new\_temp := ( ( 9 \* temp1 ) / 5 ) + 32;

new\_scale := 'F';

ELSE

new\_temp := ( ( temp1 - 32 ) \* 5 ) / 9;

new\_scale := 'C';

END IF;

END IF;

dbms\_output.Put\_line ('The new temperature in scale '

||new\_scale

||' is: '

||new\_temp);

END;

/

## Individual Solutions:-

**Lab 20**

### Hanzala Shahid

**FA18-BCS-014**

**Qno1:-**

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

DECLARE

temp1 NUMBER := &input\_a\_temp;

t\_scale CHAR := '&input\_temp\_scale';

new\_temp NUMBER;

new\_scale CHAR;

BEGIN

IF t\_scale != 'C'

AND

t\_scale != 'F' THEN

dbms\_output.Put\_line ('The scale you input is not a valid scale');

new\_temp := 0;

new\_scale := 'C';

ELSE

IF t\_scale = 'C' THEN

new\_temp := ( ( 9 \* temp1 ) / 5 ) + 32;

new\_scale := 'F';

ELSE

new\_temp := ( ( temp1 - 32 ) \* 5 ) / 9;

new\_scale := 'C';

END IF;

END IF;

dbms\_output.Put\_line ('The new temperature in scale '

||new\_scale

||' is: '

||new\_temp);

END;

/

### Hamza Bin Ahmed

**FA18-BCS-084**

**Qno1:-**

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

DECLARE

temp1 NUMBER := &input\_a\_temp;

t\_scale CHAR := '&input\_temp\_scale';

new\_temp NUMBER;

new\_scale CHAR;

BEGIN

IF t\_scale != 'C'

AND

t\_scale != 'F' THEN

dbms\_output.Put\_line ('The scale you input is not a valid scale');

new\_temp := 0;

new\_scale := 'C';

ELSE

IF t\_scale = 'C' THEN

new\_temp := ( ( 9 \* temp1 ) / 5 ) + 32;

new\_scale := 'F';

ELSE

new\_temp := ( ( temp1 - 32 ) \* 5 ) / 9;

new\_scale := 'C';

END IF;

END IF;

dbms\_output.Put\_line ('The new temperature in scale '

||new\_scale

||' is: '

||new\_temp);

END;

/

### Muhammad Sharjeel khan

**FA18-BCS-019**

**Qno1:-**

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

DECLARE

temp1 NUMBER := &input\_a\_temp;

t\_scale CHAR := '&input\_temp\_scale';

new\_temp NUMBER;

new\_scale CHAR;

BEGIN

IF t\_scale != 'C'

AND

t\_scale != 'F' THEN

dbms\_output.Put\_line ('The scale you input is not a valid scale');

new\_temp := 0;

new\_scale := 'C';

ELSE

IF t\_scale = 'C' THEN

new\_temp := ( ( 9 \* temp1 ) / 5 ) + 32;

new\_scale := 'F';

ELSE

new\_temp := ( ( temp1 - 32 ) \* 5 ) / 9;

new\_scale := 'C';

END IF;

END IF;

dbms\_output.Put\_line ('The new temperature in scale '

||new\_scale

||' is: '

||new\_temp);

END;

/

### Hamza Aslam

**FA18-BCS-050**

**Qno1:-**

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

DECLARE

temp1 NUMBER := &input\_a\_temp;

t\_scale CHAR := '&input\_temp\_scale';

new\_temp NUMBER;

new\_scale CHAR;

BEGIN

IF t\_scale != 'C'

AND

t\_scale != 'F' THEN

dbms\_output.Put\_line ('The scale you input is not a valid scale');

new\_temp := 0;

new\_scale := 'C';

ELSE

IF t\_scale = 'C' THEN

new\_temp := ( ( 9 \* temp1 ) / 5 ) + 32;

new\_scale := 'F';

ELSE

new\_temp := ( ( temp1 - 32 ) \* 5 ) / 9;

new\_scale := 'C';

END IF;

END IF;

dbms\_output.Put\_line ('The new temperature in scale '

||new\_scale

||' is: '

||new\_temp);

END;

/

### Abdullah Noor Niazi

**FA18-BCS-004**

**Qno1:-**

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

DECLARE

temp1 NUMBER := &input\_a\_temp;

t\_scale CHAR := '&input\_temp\_scale';

new\_temp NUMBER;

new\_scale CHAR;

BEGIN

IF t\_scale != 'C'

AND

t\_scale != 'F' THEN

dbms\_output.Put\_line ('The scale you input is not a valid scale');

new\_temp := 0;

new\_scale := 'C';

ELSE

IF t\_scale = 'C' THEN

new\_temp := ( ( 9 \* temp1 ) / 5 ) + 32;

new\_scale := 'F';

ELSE

new\_temp := ( ( temp1 - 32 ) \* 5 ) / 9;

new\_scale := 'C';

END IF;

END IF;

dbms\_output.Put\_line ('The new temperature in scale '

||new\_scale

||' is: '

||new\_temp);

END;

/

### Usman Jadoon

**FA18-BCS-100**

**Qno1:-**

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

DECLARE

temp1 NUMBER := &input\_a\_temp;

t\_scale CHAR := '&input\_temp\_scale';

new\_temp NUMBER;

new\_scale CHAR;

BEGIN

IF t\_scale != 'C'

AND

t\_scale != 'F' THEN

dbms\_output.Put\_line ('The scale you input is not a valid scale');

new\_temp := 0;

new\_scale := 'C';

ELSE

IF t\_scale = 'C' THEN

new\_temp := ( ( 9 \* temp1 ) / 5 ) + 32;

new\_scale := 'F';

ELSE

new\_temp := ( ( temp1 - 32 ) \* 5 ) / 9;

new\_scale := 'C';

END IF;

END IF;

dbms\_output.Put\_line ('The new temperature in scale '

||new\_scale

||' is: '

||new\_temp);

END;

/

### Usama Fareed

**FA18-BCS-026**

**Qno1:-**

Write a SQL function to convert temperature from Fahrenheit to Celsius scale

Answer:-

DECLARE

temp1 NUMBER := &input\_a\_temp;

t\_scale CHAR := '&input\_temp\_scale';

new\_temp NUMBER;

new\_scale CHAR;

BEGIN

IF t\_scale != 'C'

AND

t\_scale != 'F' THEN

dbms\_output.Put\_line ('The scale you input is not a valid scale');

new\_temp := 0;

new\_scale := 'C';

ELSE

IF t\_scale = 'C' THEN

new\_temp := ( ( 9 \* temp1 ) / 5 ) + 32;

new\_scale := 'F';

ELSE

new\_temp := ( ( temp1 - 32 ) \* 5 ) / 9;

new\_scale := 'C';

END IF;

END IF;

dbms\_output.Put\_line ('The new temperature in scale '

||new\_scale

||' is: '

||new\_temp);

END;

/