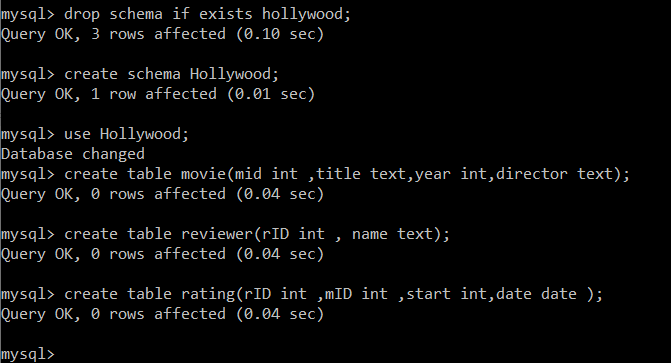
CIND-110 Class (Section DHD) ,Mohsen Selseleh, 500726502

**Question#1**

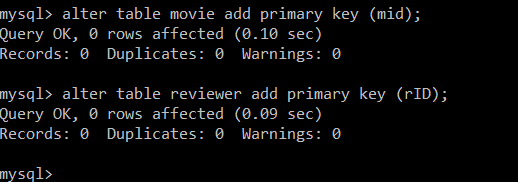
Schema creation:



1. **Solution**

alter table movie add primary key (mid);

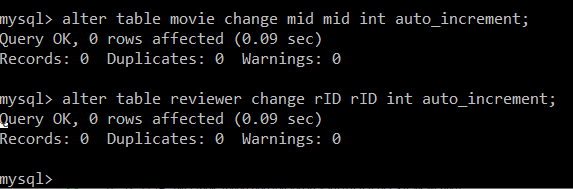
alter table reviewer add primary key (rID);



1. **Solution**:

alter table movie change mid mid int auto\_increment;

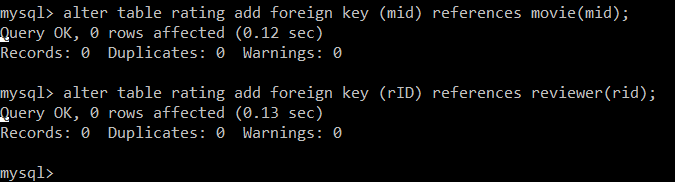
alter table reviewer change rID rID int auto\_increment;



1. **Solution**:

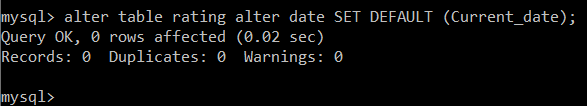
alter table rating add foreign key (mid) references movie(mid);

alter table rating add foreign key (rID) references reviewer(rid);



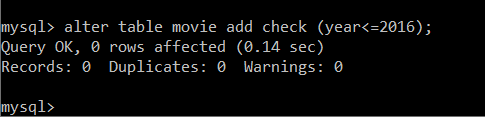
1. **Solution**:

alter table rating alter date SET DEFAULT (Current\_date);



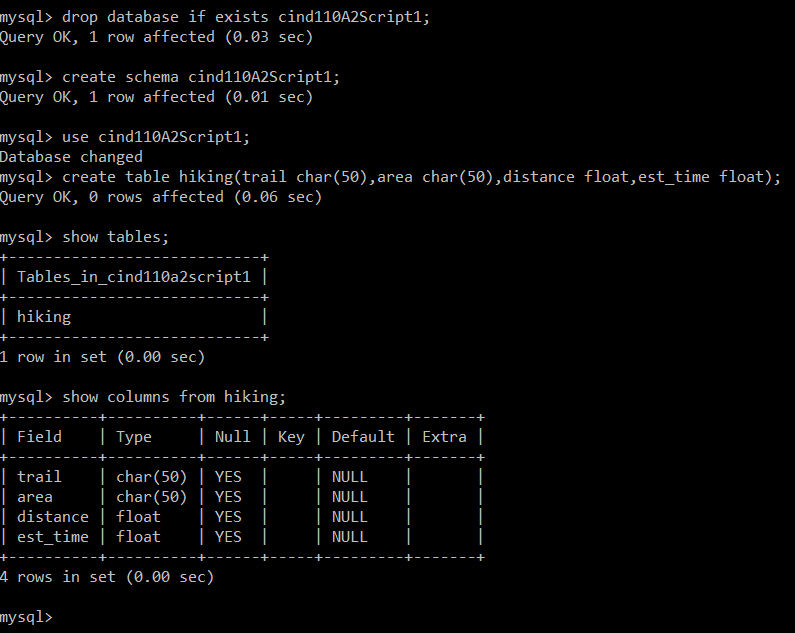
1. **Solution**:

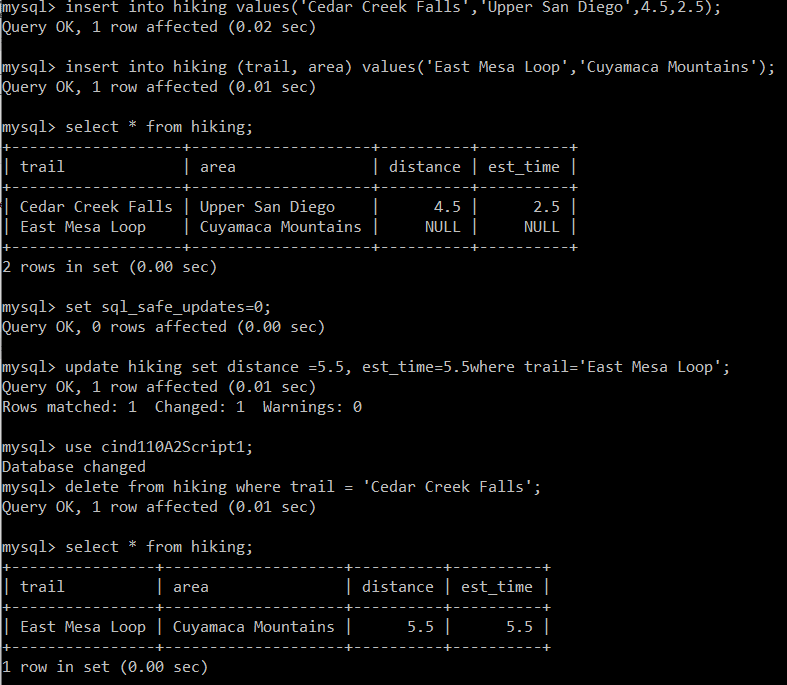
alter table movie add check (year<=2016);



**Question#2**

Running pre-requisites





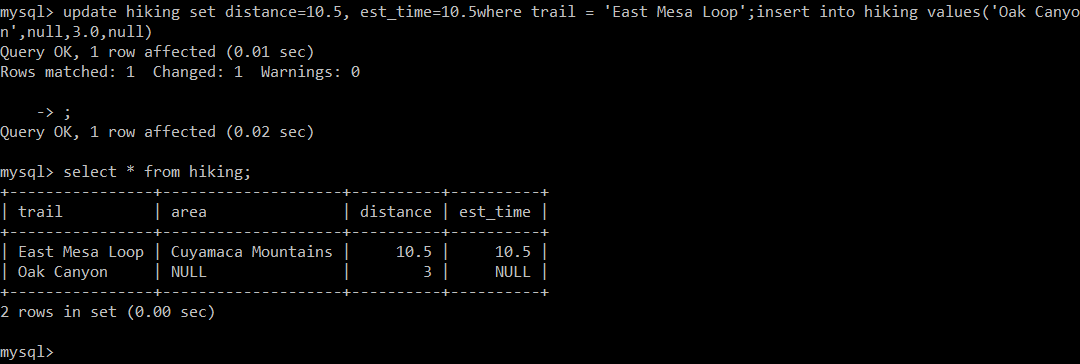
1. **Solution**:

update hiking

set distance=10.5, est\_time=10.5

where trail = 'East Mesa Loop';

insert into hiking values('Oak Canyon',null,3.0,null);



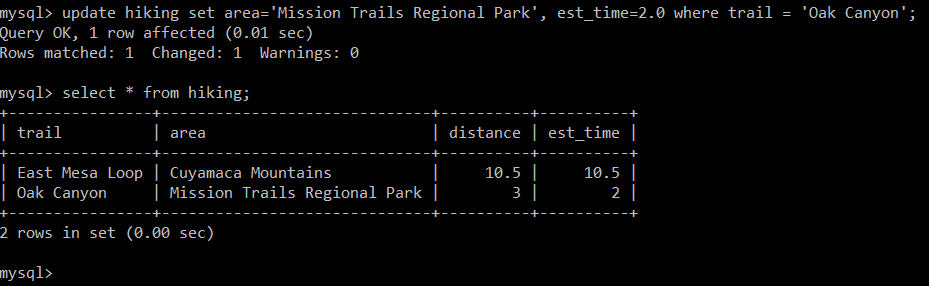
1. **Solution**:

update hiking

set area='Mission Trails Regional Park', est\_time=2.0

where trail = 'Oak Canyon';

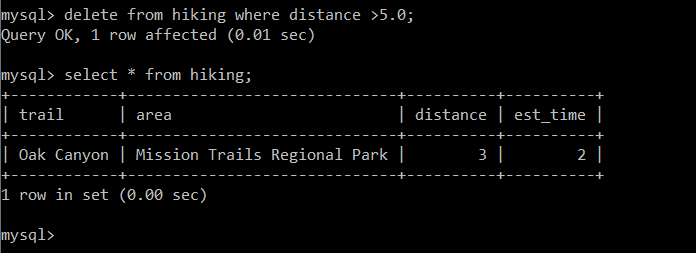
select \* from hiking;



1. **Solution**:

delete from hiking where distance >5.0;

select \* from hiking;



1. **Solution**:

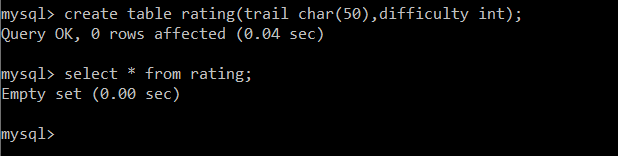
create table rating

(

trail char(50),

difficulty int

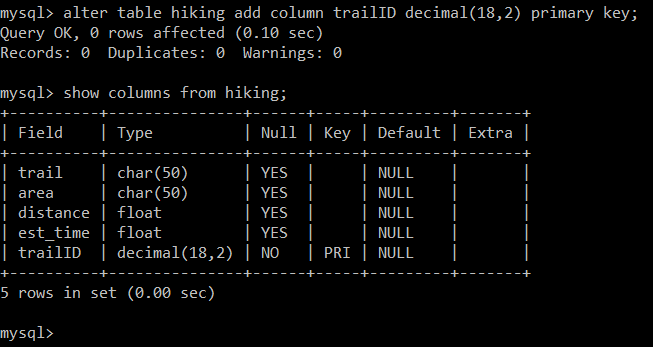
);



1. **Solution**:

alter table hiking

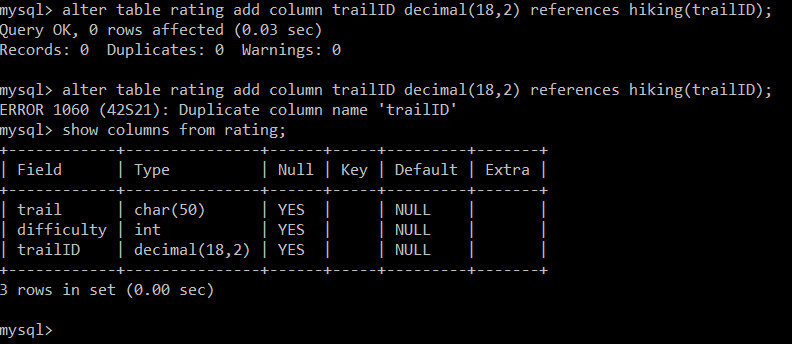
add column trailID decimal(18,2) primary key;



1. **Solution**:

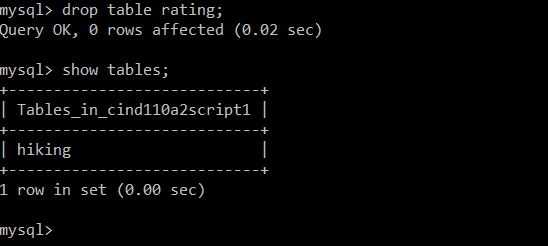
alter table rating

add column trailID decimal(18,2) references hiking(trailID);



1. **Solution:**

drop table rating;



**Question#3**

Schema creation and data insertion.

create schema q3;

use q3;

drop table if exists salesMan;

create table salesMan

(

salesManId int primary key,

salesManName varchar(50),

city varchar(50),

comission float

);

drop table if exists customer;

create table customer

(

customerId int,

customerName varchar(50),

city varchar(50),

greade int,

salesmanId int references salesMan(salesmanid)

);

drop table if exists orders;

create table orders

(

orderNo int primary key,

purchaseAmt float,

orderDate date,

customerId int references customer(customerid),

salesManID int references salesMan(salesManID)

);

insert into salesMan values (5001,'James Hoog','New York',0.15);

insert into salesMan values (5002,'Nail Knite','Paris',0.13);

insert into salesMan values (5005,'Pit Alex','London',0.11);

insert into salesMan values (5006,'Mc Lyon','Paris',0.15);

insert into salesMan values (5003,'Lauson Hen','',0.12);

insert into salesMan values (5007,'Paul Adam','Rome',0.13);

select \* from salesMan;

insert into customer values (3002,'Nick Rimando','New York',100,5001);

insert into customer values (3005,'Graham Zusi','California',200,5002);

insert into customer values (3001,'Brad Guzan','London',null,5005);

insert into customer values (3004,'Fabian Johns','Paris',300,5006);

insert into customer values (3007,'Brad Davis','New York',200,5001);

insert into customer values (3009,'Geoff Camero','Berlin',100,5003);

insert into customer values (3008,'Julian Green','London',300,5002);

insert into customer values (3003,'Jozy Altidor','Moscow',200,5007);

select \* from customer;

insert into orders values (70001,150.5,'2012-10-05',3005,5002);

insert into orders values (70009,270.65,'2012-09-10',3001,5005);

insert into orders values (70002,65.26,'2012-10-05',3002,5001);

insert into orders values (70004,110.5,'2012-08-17',3009,5003);

insert into orders values (70007,948.5,'2012-09-10',3005,5002);

insert into orders values (70005,2400.6,'2012-07-27',3007,5001);

insert into orders values (70008,5760,'2012-09-10',3002,5001);

insert into orders values (70010,1983.43,'2012-10-10',3004,5006);

insert into orders values (70003,2480.4,'2012-10-10',3009,5003);

insert into orders values (70012,250.45,'2012-06-27',3008,5002);

insert into orders values (70011,75.29,'2012-08-17',3003,5007);

insert into orders values (70013,3045.6,'2012-04-25',3002,5001);

1. **Solution**

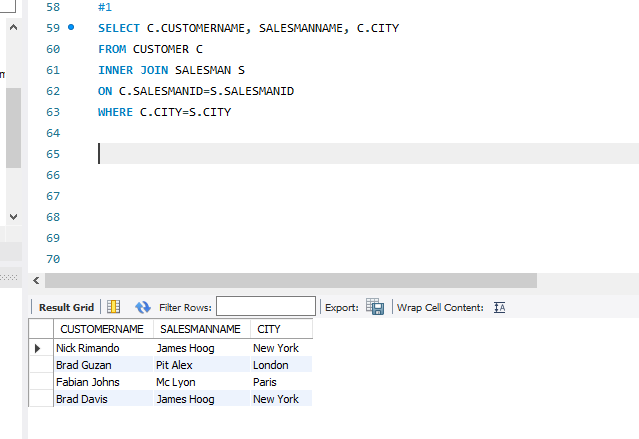
SELECT C.CUSTOMERNAME, SALESMANNAME, C.CITY

FROM CUSTOMER C

INNER JOIN SALESMAN S

ON C.SALESMANID=S.SALESMANID

WHERE C.CITY=S.CITY;



1. **Solution**

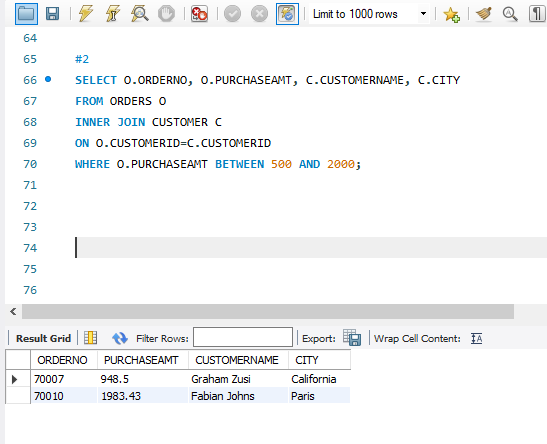
SELECT O.ORDERNO, O.PURCHASEAMT, C.CUSTOMERNAME, C.CITY

FROM ORDERS O

INNER JOIN CUSTOMER C

ON O.CUSTOMERID=C.CUSTOMERID

WHERE O.PURCHASEAMT BETWEEN 500 AND 2000;



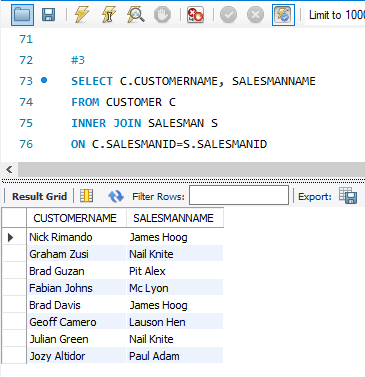
1. **Solution**

SELECT C.CUSTOMERNAME, SALESMANNAME

FROM CUSTOMER C

INNER JOIN SALESMAN S

ON C.SALESMANID=S.SALESMANID;



1. **Solution**

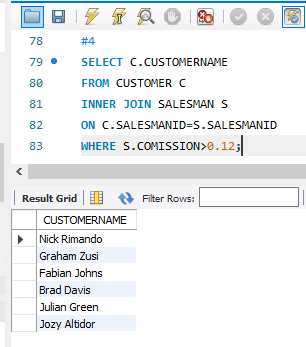
SELECT C.CUSTOMERNAME

FROM CUSTOMER C

INNER JOIN SALESMAN S

ON C.SALESMANID=S.SALESMANID

WHERE S.COMISSION>0.12;



1. **Solution**

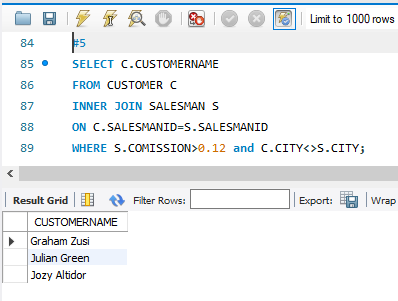
SELECT C.CUSTOMERNAME

FROM CUSTOMER C

INNER JOIN SALESMAN S

ON C.SALESMANID=S.SALESMANID

WHERE S.COMISSION>0.12 and C.CITY<>S.CITY;



1. **Solution**

SELECT O.ORDERNO,O.ORDERDATE, O.PURCHASEAMT, C.CUSTOMERNAME, S.SALESMANNAME, S.COMISSION

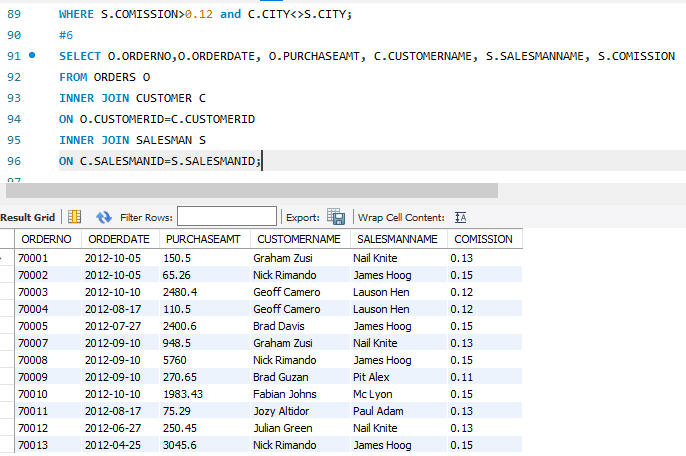
FROM ORDERS O

INNER JOIN CUSTOMER C

ON O.CUSTOMERID=C.CUSTOMERID

INNER JOIN SALESMAN S

ON C.SALESMANID=S.SALESMANID;



1. **Solution**

SELECT O.ORDERNO, O.ORDERDATE, O.PURCHASEAMT, C.CUSTOMERID, C.CUSTOMERNAME, C.CITY CUSTOMERCITY,

C.GREADE CUSTOMERGRADE, S.SALESMANID, S.SALESMANNAME, S.CITY SALESMAN\_CITY, S.COMISSION

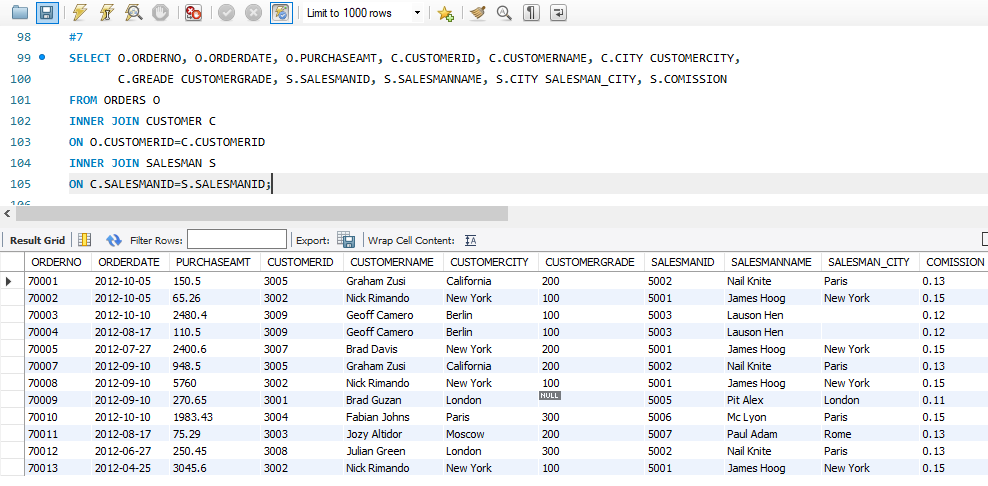
FROM ORDERS O

INNER JOIN CUSTOMER C

ON O.CUSTOMERID=C.CUSTOMERID

INNER JOIN SALESMAN S

ON C.SALESMANID=S.SALESMANID;



**Question #4**

**Solution:**

Based on the given schema below are the possible foreign keys and their reference relations.

1. Relation : ENROLL, Foreign key : Ssn , References STUDENT (Ssn)
2. Relation : ENROLL, Foreign Key : Course# , References COURSE(course#)
3. Relation : BOOK ADOPTION, Foreign Key : Course# , References COURSE(course#)
4. Relation : BOOK ADOPTION, Foreign Key : Bookisbn , References TEXT(Bookisbn)