NAME : SAAD SALMAN DAR

ROLL NUMBER : L1S23BSCS0324

DBMS ASSIGNMENT 2

Q1.

create database assignment2;

use assignment2;

create table BOOK(

Book\_id int,

Title varchar(255) not null,

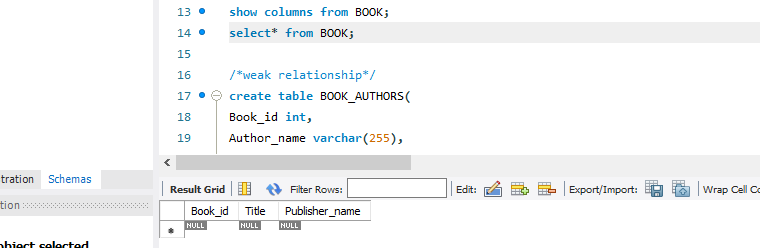
Publisher\_name varchar(255),

constraint pk\_BOOK primary key(Book\_id)

);

show columns from BOOK;

SELECT\* FROM BOOK;



/\*weak relationship\*/

create table BOOK\_AUTHORS(

Book\_id int,

Author\_name varchar(255),

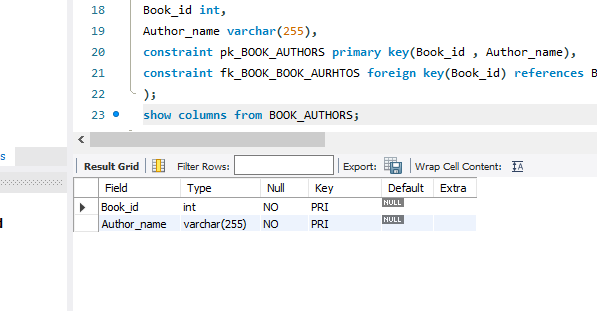
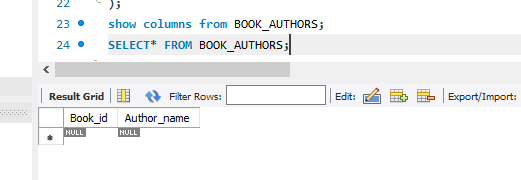
constraint pk\_BOOK\_AUTHORS primary key(Book\_id , Author\_name),

constraint fk\_BOOK\_BOOK\_AURHTOS foreign key(Book\_id) references BOOK(Book\_id) on update cascade on delete cascade

);

show columns from BOOK\_AUTHORS;

SELECT\* FROM BOOK\_AUTHORS;

/\* on deletion of the book pk , in this table the respective tuple having author name will also delete(the entire column will) thus showing

the existance of the book\_author entries are entirely reliant on the book id and would cease to exist without them\*/

create table PUBLISHER(

Name varchar(255),

Address varchar(255),

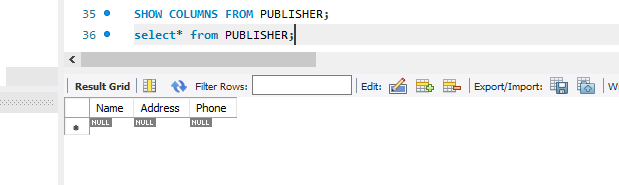
Phone varchar(255),

constraint pk\_PUBLISHER primary key(Name)

);

SHOW COLUMNS FROM PUBLISHER;

SELECT\* FROM PUBLISHER;



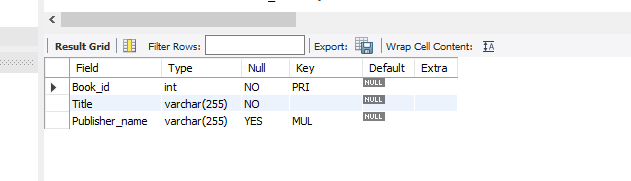
/\* creating reference between publisher\_name residing in the book relation with the publisher table\*/

show columns from BOOK;

alter table BOOK

ADD CONSTRAINT fk\_PUBLISHER\_BOOK foreign key(Publisher\_name) references PUBLISHER(Name) on update cascade on delete set null;

show columns FROM BOOK;



SELECT\* FROM BOOK;

/\* many->many relationship a seperate table will be created.

a book can be in many brancHes and a branch can have many copies of same book.\*/

create table BOOK\_COPIES(

Book\_id int,

Branch\_id int,

No\_of\_copies int,

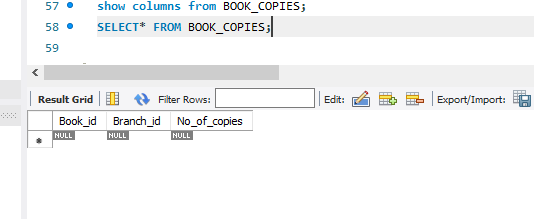
constraint pk\_BOOK\_COPIES primary key(Book\_id , Branch\_id),

constraint fk\_BOOK\_BOOK\_COPIES foreign key(Book\_id) references BOOK(Book\_id) on update cascade on delete cascade

/\* if the books get's deleted from our database no need of keeping it's info about it's copies in our database\*/

);

show columns from BOOK\_COPIES;



create table LIBRARY\_BRANCH(

Branch\_id int,

Branch\_name varchar(255) not null,

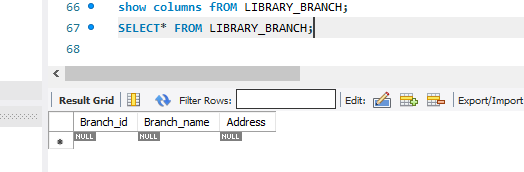
Address varchar(255) not null,

constraint pk\_LIBRARY\_BRANCH primary key(Branch\_id)

);

show columns fROM LIBRARY\_BRANCH;

SELECT\* FROM LIBRARY\_BRANCH;

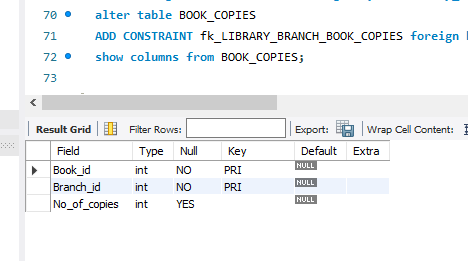


/\* creating reference of the foreign key of library\_branch residing in the book\_copies table\*/

alter table BOOK\_COPIES

ADD CONSTRAINT fk\_LIBRARY\_BRANCH\_BOOK\_COPIES foreign key(Branch\_id) references LIBRARY\_BRANCH(Branch\_id) on update cascade on delete cascade;

show columns from BOOK\_COPIES;



create table BORROWER(

Card\_no int,

Name varchar(255),

Address varchar(255),

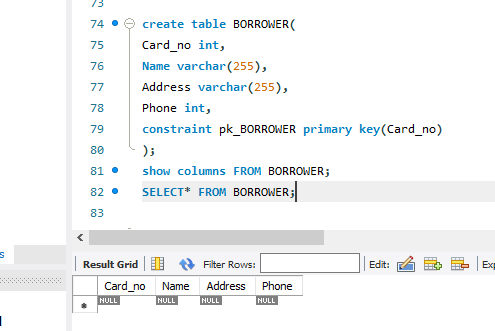
Phone int,

constraint pk\_BORROWER primary key(Card\_no)

);

show columns FROM BORROWER;

SELECT\* FROM BORROWER;



/\* TERTIARY RELATIONSHIP BETWEEN BOOK\_ID , BRANCH\_ID , CARD\_NO ALL ARE REQUIRED TO KEEP INFO ABOUT THE BOOKS LOANED OUT AND EXTRA INFO LIKE DATE\_OUT AND DUE\_DATE COMES TO US BECUASE OF THESE RELATIONSHIP ;)

/\* I WILL TRY IMPLEMENTING ON DELETE RESTRICT FOR THIS , WHAT THIS QUERY WILL DO IS STOP THE DELETION OF BOOK , BORROWER AND BRANCH INFO IF THE SPECIFIC BOOK IS LOANED OUT ALL THE INFO RELATED TO IT WILL NOT BE ABLE TO BE DELETED. BOOK\_LOAN WALEY RELATIONSHIP K PASS IKHTEYAAR AJATA HAE AND

IT DOES NOT ALLOWS FOR IT TO BE DELETED.\*/

CREATE TABLE BOOK\_LOANS(

Book\_id int,

Branch\_id int,

Card\_id int,

Date\_out date not null,

Due\_date date not null,

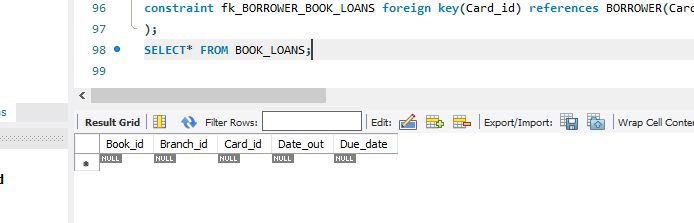
constraint pk\_BOOK\_LOANS primary key(Book\_id , Branch\_id , Card\_id),

constraint fk\_BOOK\_BOOK\_LOANS foreign key(Book\_id) references BOOK(Book\_id) on update cascade on delete restrict,

constraint fk\_LIBRARY\_BRANCH\_BOOK\_LOANS foreign key(Branch\_id) references LIBRARY\_BRANCH(Branch\_id) on update cascade on delete restrict,

constraint fk\_BORROWER\_BOOK\_LOANS foreign key(Card\_id) references BORROWER(Card\_no) on update cascade on delete restrict

);



Q2.

/\*<<<<<<<<<< Q2 INSERTING VALUES INTO THE DATABASE >>>>>>>>>>>>>>>>\*/

show columns from BOOK;

insert into BOOK

value

(8000 , 'Don Quixote' , null ),

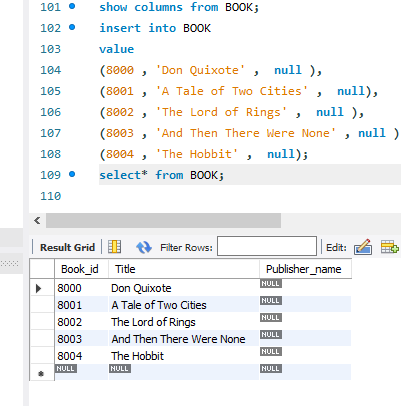
(8001 , 'A Tale of Two Cities' , null),

(8002 , 'The Lord of Rings' , null ),

(8003 , 'And Then There Were None' , null ),

(8004 , 'The Hobbit' , null);

select\* from BOOK;



insert into BOOK\_AUTHORS

value

(8000 , 'Harper Lee'),

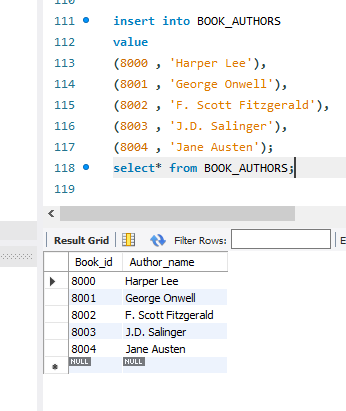
(8001 , 'George Onwell'),

(8002 , 'F. Scott Fitzgerald'),

(8003 , 'J.D. Salinger'),

(8004 , 'Jane Austen');

SELECT\* FROM BOOK\_AUTHORS;



insert into PUBLISHER

values

('Penguin Books' , '123 Penguin Lane,London<,UK' , 04231234567),

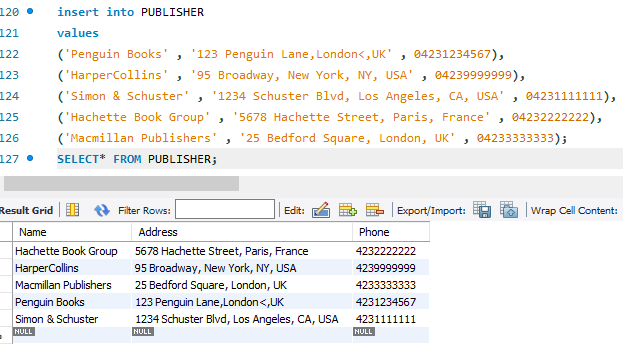
('HarperCollins' , '95 Broadway, New York, NY, USA' , 04239999999),

('Simon & Schuster' , '1234 Schuster Blvd, Los Angeles, CA, USA' , 04231111111),

('Hachette Book Group' , '5678 Hachette Street, Paris, France' , 04232222222),

('Macmillan Publishers' , '25 Bedford Square, London, UK' , 04233333333);

SELECT\* FROM PUBLISHERS;



-- adding values of the publishers in the books column

update BOOK

SET Publisher\_name = 'Penguin Books'

where Book\_id=8000;

update BOOK

SET Publisher\_name = 'HarperCollins'

where Book\_id>=8001 and Book\_id<=8004;

select\* from BOOK;

update BOOK

set publisher\_name = 'Hachette Book Group'

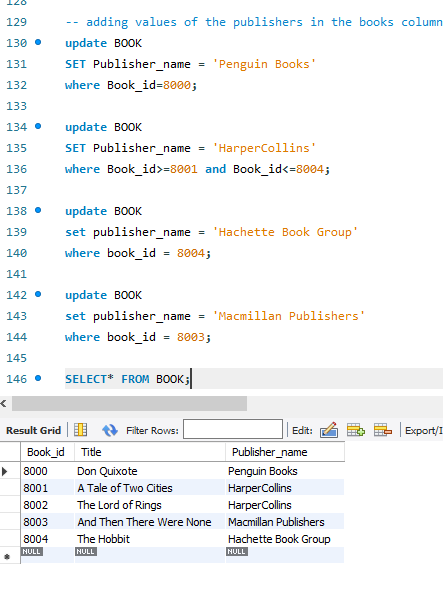
where book\_id = 8004;

update BOOK

set publisher\_name = 'Macmillan Publishers'

where book\_id = 8003;

SELECT\* FROM BOOK;



-- adding values to the relation library\_branch

show columns FROM LIBRARY\_BRANCH;

INSERT INTO LIBRARY\_BRANCH

VALUES

(900 , 'Central Library' , '123 Main Street'),

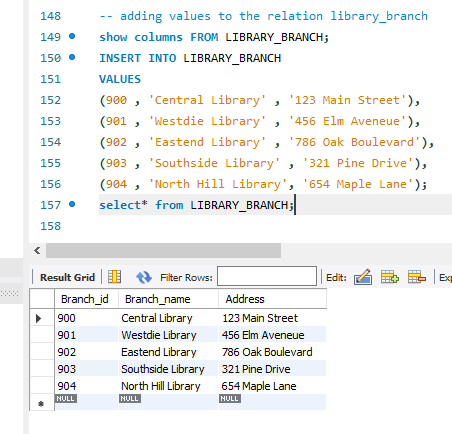
(901 , 'Westdie Library' , '456 Elm Aveneue'),

(902 , 'Eastend Library' , '786 Oak Boulevard'),

(903 , 'Southside Library' , '321 Pine Drive'),

(904 , 'North Hill Library', '654 Maple Lane');

select\* from LIBRARY\_BRANCH;



-- ADDING VALUES INTO BOOK\_COPIES --

SHOW COLUMNS FROM BOOK\_COPIES;

INSERT INTO BOOK\_COPIES

VALUES

(8000 , 900 , 10),

(8000 , 901 , 12),

(8000 , 904 , 2),

(8001 , 902 , 4),

(8001 , 903 , 5),

(8002 , 902 , 8),

(8002 , 904 , 15),

(8003 , 900 , 1),

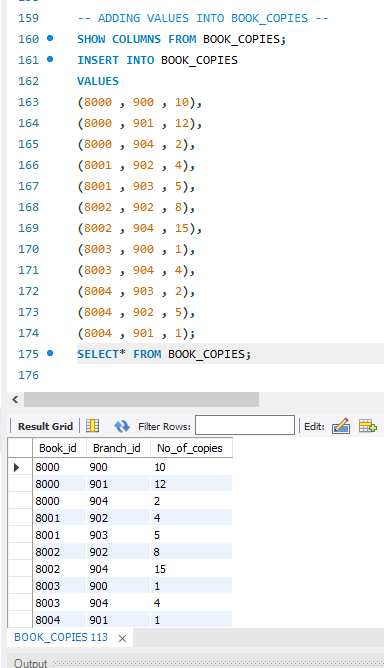
(8003 , 904 , 4),

(8004 , 903 , 2),

(8004 , 902 , 5),

(8004 , 901 , 1);

SELECT\* FROM BOOK\_COPIES;



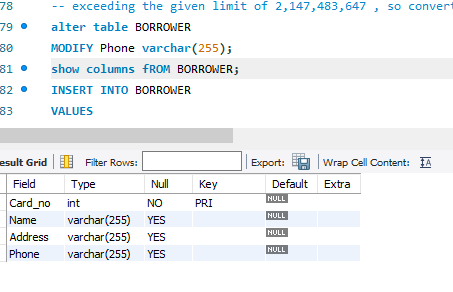
-- ADDING VALUES INTO BORROWER RELATION

-- exceeding the given limit of 2,147,483,647 , so converting int to varchar to store numbers of the users

alter table BORROWER

MODIFY Phone varchar(255);

show columns fROM BORROWER;



INSERT INTO BORROWER

VALUES

(1 ,'Saad Salman Dar' , '24 Usman Block , Bahria Town , Lhr' , '+923241234567'),

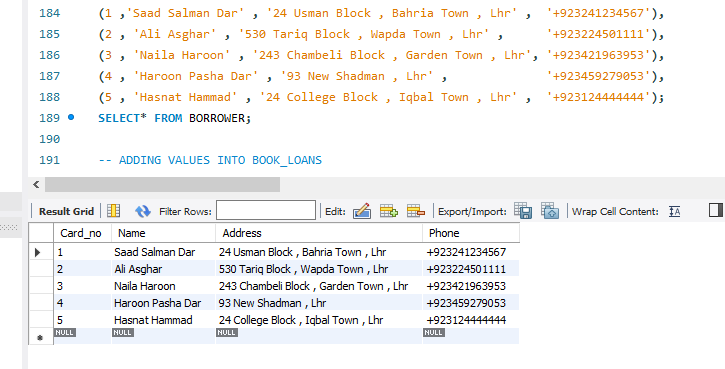
(2 , 'Ali Asghar' , '530 Tariq Block , Wapda Town , Lhr' , '+923224501111'),

(3 , 'Naila Haroon' , '243 Chambeli Block , Garden Town , Lhr', '+923421963953'),

(4 , 'Haroon Pasha Dar' , '93 New Shadman , Lhr' , '+923459279053'),

(5 , 'Hasnat Hammad' , '24 College Block , Iqbal Town , Lhr' , '+923124444444');

SELECT\* FROM BORROWER;



-- ADDING VALUES INTO BOOK\_LOANS

INSERT INTO BOOK\_LOANS

VALUES

(8000 , 900 , 1 , '2022-10-01' , '2022-10-30'),

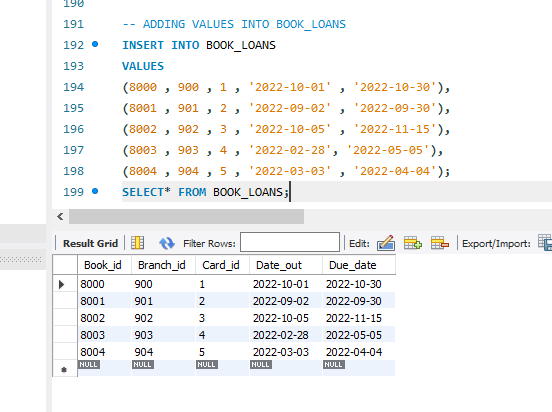
(8001 , 901 , 2 , '2022-09-02' , '2022-09-30'),

(8002 , 902 , 3 , '2022-10-05' , '2022-11-15'),

(8003 , 903 , 4 , '2022-02-28', '2022-05-05'),

(8004 , 904 , 5 , '2022-03-03' , '2022-04-04');

SELECT\* FROM BOOK\_LOANS;



Q3

-- Q3 QUERIES

-- 1

-- using the join keyword to solve this problem , although have not studied it yet ;(

insert into BOOK\_COPIES

VALUES

(8004, 904 , 20),

(8003 , 902 , 25),

(8004 , 900 , 24);

INSERT INTO BOOK\_COPIES

VALUES

(8002 , 901 , 15),

(8001 , 900 , 11),

(8002 , 903 , 12);

INSERT INTO BOOK\_COPIES

VALUES

(8001 , 904 , 14),

(8004 , 904 , 20);

SELECT B.Title, B.Book\_id, P.Name AS Publisher, P.Address, BC.No\_of\_copies

FROM BOOK B

JOIN BOOK\_COPIES BC

ON B.Book\_id = BC.Book\_id

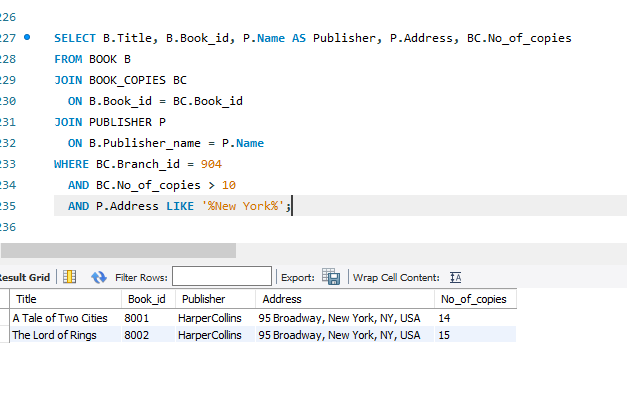
JOIN PUBLISHER P

ON B.Publisher\_name = P.Name

WHERE BC.Branch\_id = 904

AND BC.No\_of\_copies > 10

AND P.Address LIKE '%New York%';



-- 2

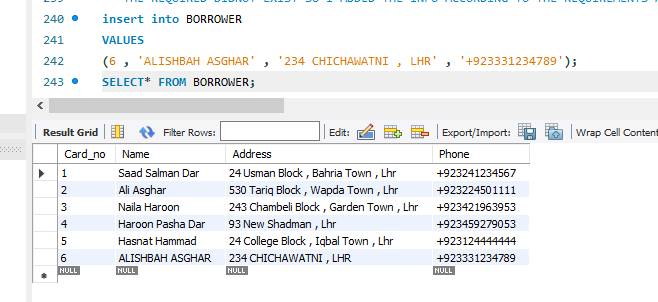
-- THE REQUIRED DIDNOT EXIST SO I ADDED THE INFO ACCORDING TO THE REQUIREMENTS AND THEN I CALLED THE QUERY.

insert into BORROWER

VALUES

(6 , 'ALISHBAH ASGHAR' , '234 CHICHAWATNI , LHR' , '+923331234789');

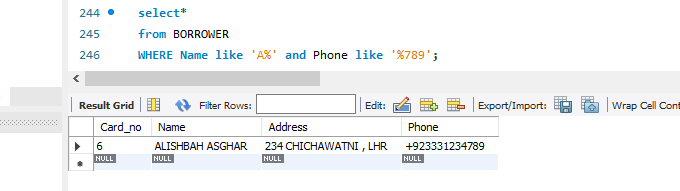
SELECT\* FROM BORROWER;



select\*

from BORROWER

WHERE Name like 'A%' and Phone like '%789';



-- 3

-- ADDING THE REQUIRED VALUES INTO THE DATABASE ACCORDING TO THE CONDITION

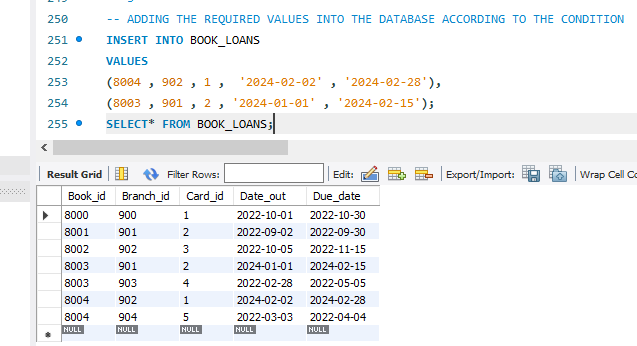
INSERT INTO BOOK\_LOANS

VALUES

(8004 , 902 , 1 , '2024-02-02' , '2024-02-28'),

(8003 , 901 , 2 , '2024-01-01' , '2024-02-15');

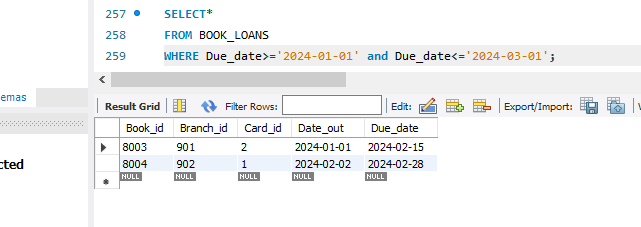
SELECT\* FROM BOOK\_LOANS;



SELECT\*

FROM BOOK\_LOANS

WHERE Due\_date>='2024-01-01' and Due\_date<='2024-03-01';



-- 4

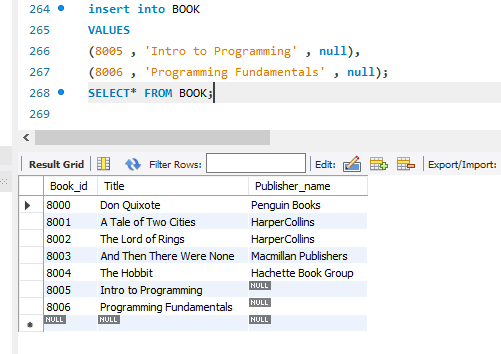
SELECT\* FROM BOOK;

insert into BOOK

VALUES

(8005 , 'Intro to Programming' , null),

(8006 , 'Programming Fundamentals' , null);

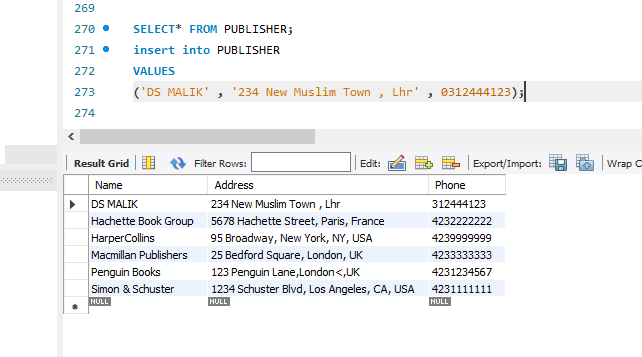


SELECT\* FROM PUBLISHER;

insert into PUBLISHER

VALUES

('DS MALIK' , '234 New Muslim Town , Lhr' , 0312444123);

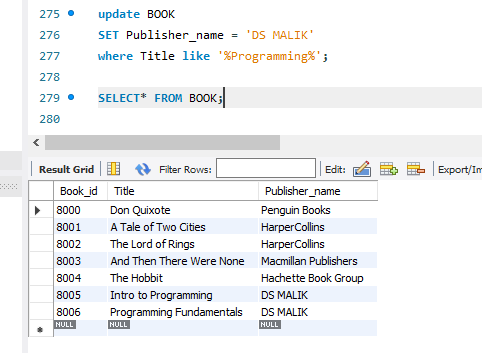


update BOOK

SET Publisher\_name = 'DS MALIK'

where Title like '%Programming%';

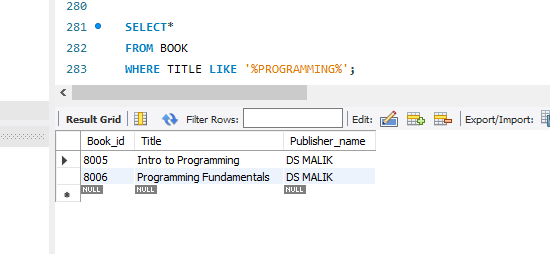
SELECT\* FROM BOOK;



SELECT\*

FROM BOOK

WHERE TITLE LIKE '%PROGRAMMING%';

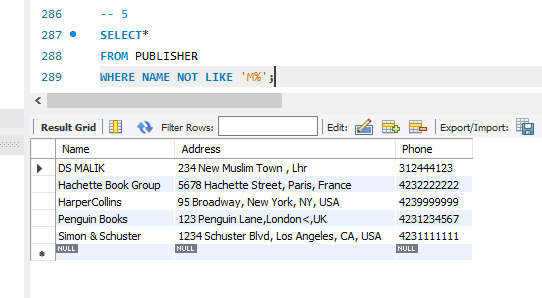


-- 5

SELECT\*

FROM PUBLISHER

WHERE NAME NOT LIKE 'M%';



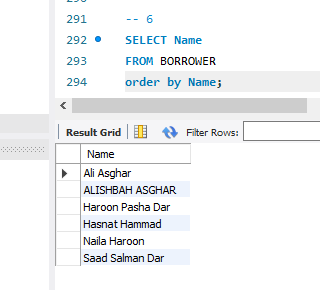
-- 6

SELECT Name

FROM BORROWER

order by Name;

select\* from BOOK;



-- 7

-- I HAVE A TOTAL OF 7 BOOKS I HAVE TO CALL ONLY THE FIRST 5 (ORDER BY NAME)

-- i am using the limit at the end of the query to limit the number of titles presented to 5

SELECT Title

from BOOK

order by Title

limit 5;

