NAME: KARTHIK S
USN: 1BM19CS070
DATE: 05/05/21

# **DBMS LAB PROGRAM – 2**

# **2.BOOK DEALER DATABASE**

### **QUESTION:**

The following tables are maintained by a book dealer: AUTHOR(author-id: int, name: String, city: String, country:

String)

PUBLISHER(publisher-id: int, name: String, city: String,

country: String)

CATALOG(book-id: int, title: String, author-id: int, publisher-id: int, category-id: int,

year: int,price: int)

CATEGORY(category-id: int, description: String)

ORDER-DETAILS(order-no: int, book-id: int, quantity: int)

- i)Create the above tables by properly specifying the primary keys and the foreign keys.
- ii) Enter at least five tuples for each relation.
- iii) Give the details of the authors who have 2 or more books in the catalog and the price of the books in the

catalog and the year of publication is after 2000.

- iv) Find the author of the book which has maximum sales.
- v) Demonstrate how you increase the price of books published by a specific publisher by10%.

### **PROGRAM CODE:**

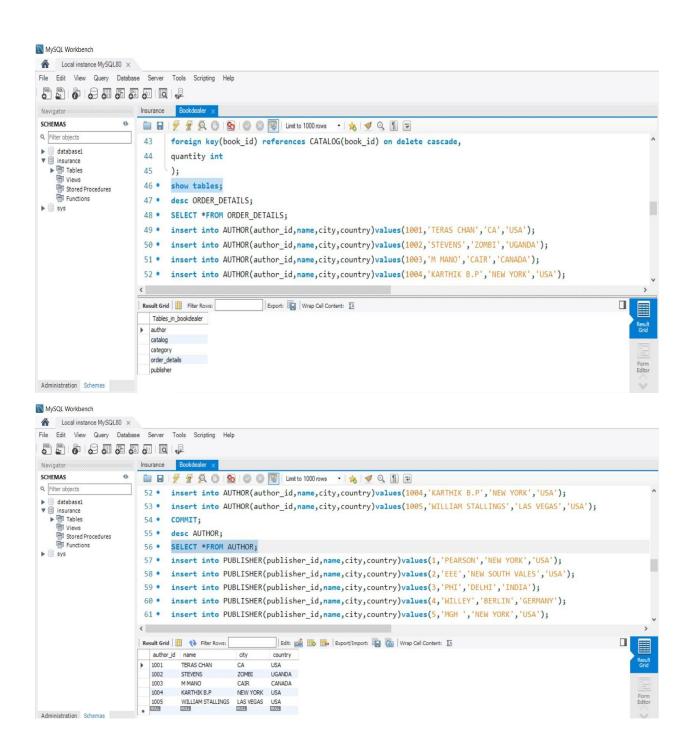
```
create database bookdealer;
use bookdealer;
create table AUTHOR (
  author_id int,
  name varchar(20),
  city varchar(15),
  country varchar(15),
  primary key(author_id)
  );
  show tables;
  desc AUTHOR;
  SELECT *FROM AUTHOR;
  create table PUBLISHER (
  publisher_id int,
```

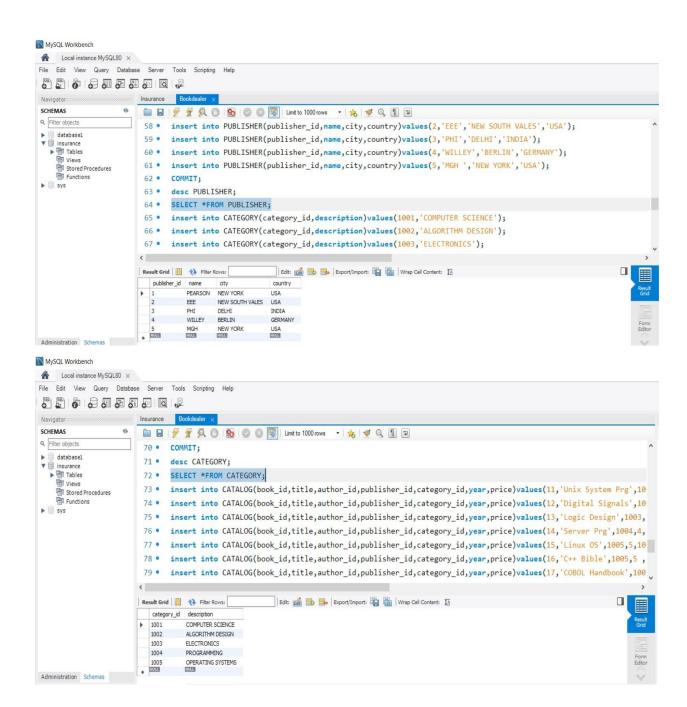
```
name varchar(20),
city varchar(15),
country varchar(15),
primary key(publisher_id)
);
show tables;
desc PUBLISHER;
SELECT *FROM PUBLISHER;
create table CATEGORY(
category_id int,
description varchar(20),
primary key(category_id)
);
show tables;
desc CATEGORY;
SELECT *FROM CATEGORY;
create table CATALOG (
book_id int,
title varchar(15),
author_id int,
publisher_id int,
category_id int,
foreign key(author_id) references AUTHOR(author_id) on delete cascade,
foreign key(publisher_id) references PUBLISHER(publisher_id) on delete cascade,
foreign key(category_id) references CATEGORY(category_id) on delete cascade,
year int,
price int,
primary key(book_id)
);
show tables;
desc CATALOG;
SELECT *FROM CATALOG;
create table ORDER DETAILS (
order_no int,
book_id int,
foreign key(book_id) references CATALOG(book_id) on delete cascade,
quantity int
);
show tables;
desc ORDER_DETAILS;
SELECT *FROM ORDER_DETAILS;
insert into AUTHOR(author_id,name,city,country)values(1001,'TERAS CHAN','CA','USA');
insert into
```

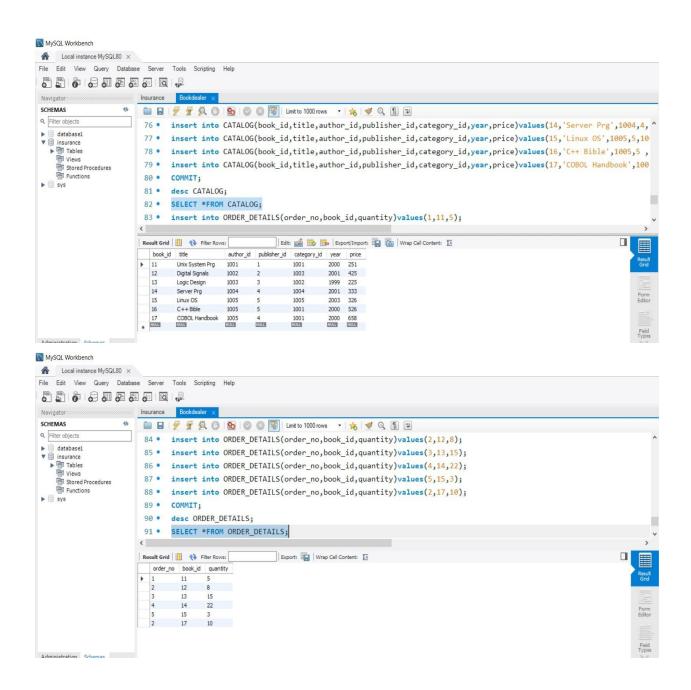
```
AUTHOR(author_id,name,city,country)values(1002,'STEVENS','ZOMBI','UGANDA');
insert into AUTHOR(author_id, name, city, country) values (1003, 'M
MANO', 'CAIR', 'CANADA');
insert into AUTHOR(author_id,name,city,country)values(1004,'KARTHIK B.P','NEW
YORK', 'USA');
insert into AUTHOR(author_id,name,city,country)values(1005,'WILLIAM STALLINGS','LAS
VEGAS','USA');
COMMIT;
desc AUTHOR;
SELECT *FROM AUTHOR;
insert into PUBLISHER(publisher_id,name,city,country)values(1,'PEARSON','NEW
YORK', 'USA');
insert into PUBLISHER(publisher_id,name,city,country)values(2,'EEE','NEW SOUTH
VALES','USA');
insert into
PUBLISHER(publisher_id,name,city,country)values(3,'PHI','DELHI','INDIA');
insert into
PUBLISHER(publisher_id, name, city, country) values(4, 'WILLEY', 'BERLIN', 'GERMANY');
insert into PUBLISHER(publisher_id,name,city,country)values(5,'MGH ','NEW
YORK', 'USA');
COMMIT;
desc PUBLISHER;
SELECT *FROM PUBLISHER;
insert into CATEGORY(category_id,description)values(1001,'COMPUTER SCIENCE');
insert into CATEGORY(category_id,description)values(1002,'ALGORITHM DESIGN');
insert into CATEGORY(category_id,description)values(1003,'ELECTRONICS');
insert into CATEGORY(category_id,description)values(1004,'PROGRAMMING');
insert into CATEGORY(category_id,description)values(1005,'OPERATING SYSTEMS');
COMMIT;
desc CATEGORY;
SELECT *FROM CATEGORY;
insert into
CATALOG(book_id, title, author_id, publisher_id, category_id, year, price) values(11, 'Unix
System Prg',1001,1,1001,2000,251);
insert into
CATALOG(book_id,title,author_id,publisher_id,category_id,year,price)values(12,'Digit
al Signals',1002,2,1003,2001,425);
insert into
CATALOG(book_id,title,author_id,publisher_id,category_id,year,price)values(13,'Logic
Design',1003,3,1002,1999,225);
insert into
CATALOG(book_id,title,author_id,publisher_id,category_id,year,price)values(14,'Serve
r Prg',1004,4,1004,2001,333);
```

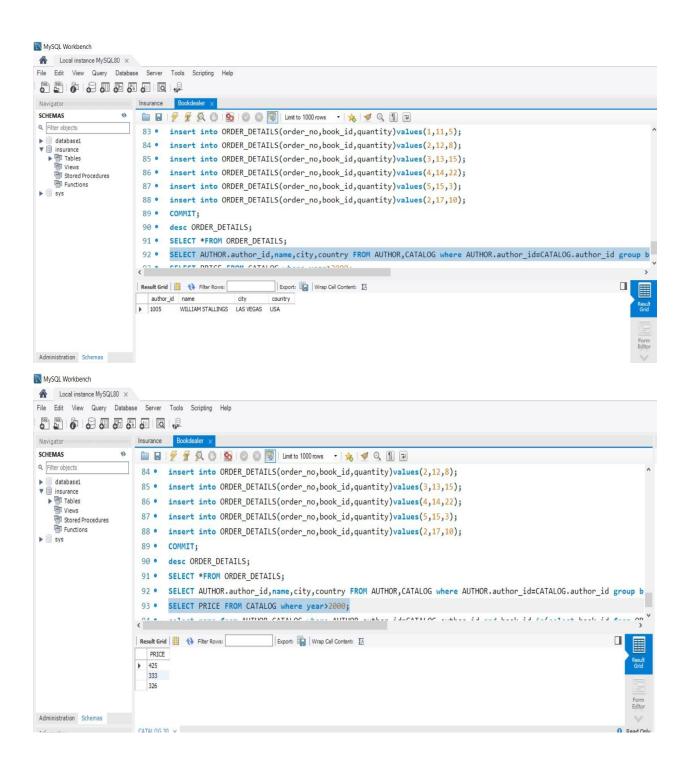
```
insert into
CATALOG(book_id,title,author_id,publisher_id,category_id,year,price)values(15,'Linux
OS',1005,5,1005,2003,326);
insert into
CATALOG(book_id,title,author_id,publisher_id,category_id,year,price)values(16,'C++
Bible',1005,5 ,1001,2000,526);
insert into
CATALOG(book_id,title,author_id,publisher_id,category_id,year,price)values(17,'COBOL
Handbook',1005,4,1001,2000,658);
COMMIT;
desc CATALOG;
SELECT *FROM CATALOG;
insert into ORDER_DETAILS(order_no,book_id,quantity)values(1,11,5);
insert into ORDER_DETAILS(order_no,book_id,quantity)values(2,12,8);
insert into ORDER_DETAILS(order_no,book_id,quantity)values(3,13,15);
insert into ORDER_DETAILS(order_no,book_id,quantity)values(4,14,22);
insert into ORDER_DETAILS(order_no,book_id,quantity)values(5,15,3);
insert into ORDER_DETAILS(order_no,book_id,quantity)values(2,17,10);
COMMIT;
desc ORDER_DETAILS;
SELECT *FROM ORDER DETAILS;
/*Give the details of the authors who have 2 or more books in the catalog and the
price of the books in the
          catalog and the year of publication is after 2000*/
SELECT AUTHOR.author_id,name,city,country FROM AUTHOR,CATALOG where
AUTHOR.author_id=CATALOG.author_id group by CATALOG.author_id having
count(CATALOG.author id)>=2;
SELECT price from CATALOG where year>2000;
/*Find the author of the book which has maximum sales.*/
select name from AUTHOR, CATALOG where AUTHOR.author_id=CATALOG.author_id and book_id
in(select book_id from ORDER_DETAILS where quantity=(select max(quantity) from
ORDER_DETAILS));
/*Demonstrate how you increase the price of books published by a specific publisher
by 10%.*/
update CATALOG set price=1.1*price where publisher_id in(select publisher_id from
PUBLISHER where name='PEARSON');
select *from CATALOG;
```

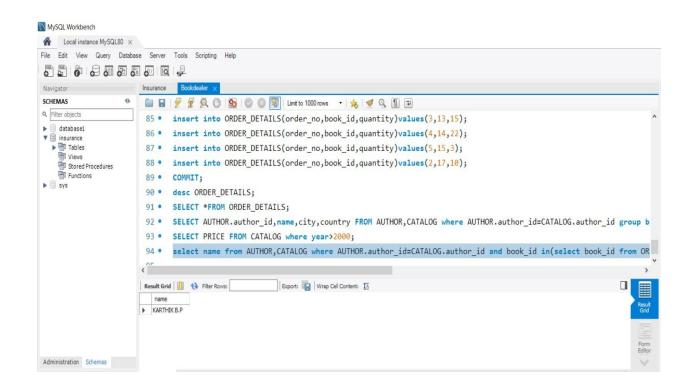
#### **SCREENSHOT OF THE PROGRAM OUTPUT:**

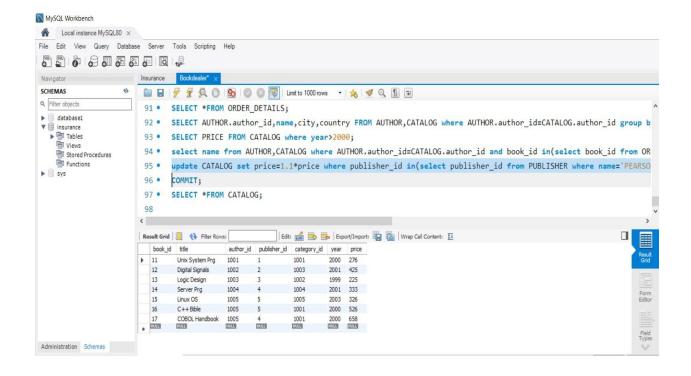












\*\* END OF WEEK 2 PROGRAM \*\*