**Program 1**

**Create tables product and purchase with without constraint.**

**Step 1:-**

**SQL> create table Product (PID int (10) primary key, Name varchar2 (20) Not Null, Price int (8, 2));**

**Table created.**

**SQL> create table purchase (PO int (10) primary key, PRODUCT\_ID int (10) references Product (PID), Qty int (5));**

**Table created.**

**SQL> desc Product;**

**SQL> desc Purchase;**

**Step 2:- Insert five tuples into each relation**

**Insert into Product (PID, NAME, PRICE) values (10,’PRINTER’, 20000);**

**1 rows created.**

**Insert into Product (PID, NAME, PRICE) values (20,’KEYBOARD’, 20000);**

**1 rows created.**

**Insert into Product (PID, NAME, PRICE) values (30,’MONITOR’, 15000);**

**1 rows created.**

**Insert into Product (PID, NAME, PRICE) values (40,’TABLE’, 14000);**

**1 rows created.**

**Insert into Product (PID, NAME, PRICE) values (10,’SCANNER’, 20000);**

**1 rows created.**

**Insert into Purchase (PO, PRODUCT\_ID, ID\_Qty) values (101, 10, 25);**

**1 rows created.**

**Insert into Purchase (PO, PRODUCT\_ID, ID\_Qty) values (102, 40, 20);**

**1 rows created.**

**Insert into Purchase (PO, PRODUCT\_ID, ID\_Qty) values (107, 30, 40);**

**1 rows created.**

**Insert into Purchase (PO, PRODUCT\_ID, ID\_Qty) values (104, 40, 50);**

**1 rows created.**

**Insert into Purchase (PO, PRODUCT\_ID, ID\_Qty) values (105, 40, 10);**

**1 rows created.**

**Step 3:- Display all the tuples in the product and purchase table**

**SQL> select \* from Product;**

**SQL> select \* from Purchase;**

**Step 4:- update the Product name for the PID=40 as CAMERA.**

**SQL> Update Product set Name=’CAMERA’ where PID=40;**

**1 rows updated.**

**SQL> select \* from Product;**

**Step 5:- Delete information about the product whose PID=50**

**SQL> DELETE form Product where PID=50;**

**1 row deleted.**

**SQL> select \* from Product;**

**Step 5:- Perform Saving and Undoing.**

**SQL> Insert into Product (PID, Name, Price) values (50,’MOBILE’, 35000);**

**1 rows created.**

**SQL> Insert into Product (PID, Name, Price) values (50,’LAPTOP’, 70000);**

**1 rows created.**

**SQL>commit;**

**SQL>select \* from Product;**

**SQL> savepoint S1;**

**Savepoint created.**

**SQL> SQL> Insert into Product (PID, Name, Price) values (50,’TABLE’, 50000);**

**1 rows created.**

**SQL> Insert into Product (PID, Name, Price) values (50,’CHAIR’, 25000);**

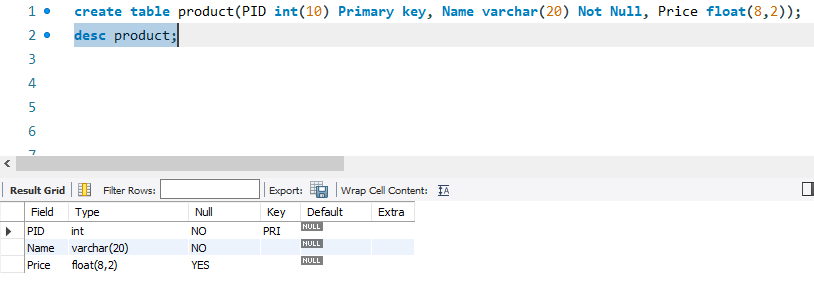
**1 rows created.**

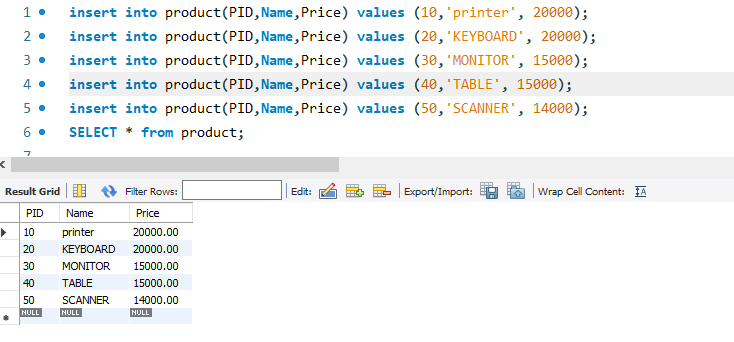
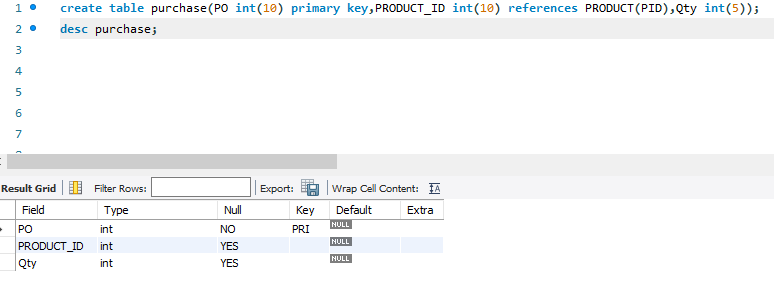
**SQL> Rollback to S1;**

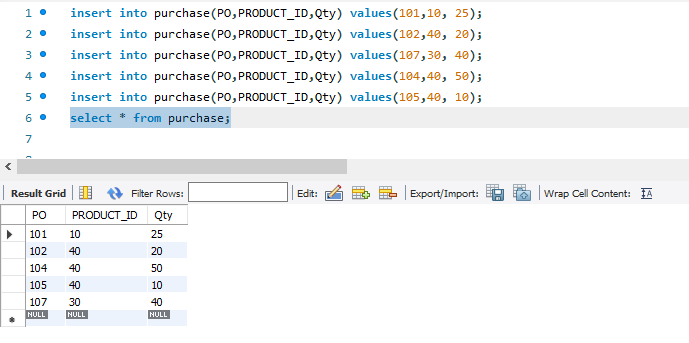
**Rollback complete.**

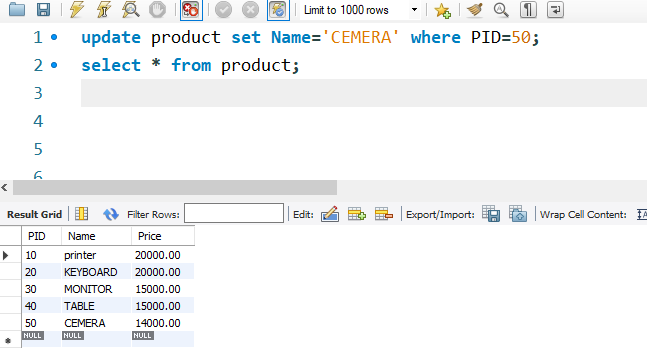
**SQL>select \* from Product;**

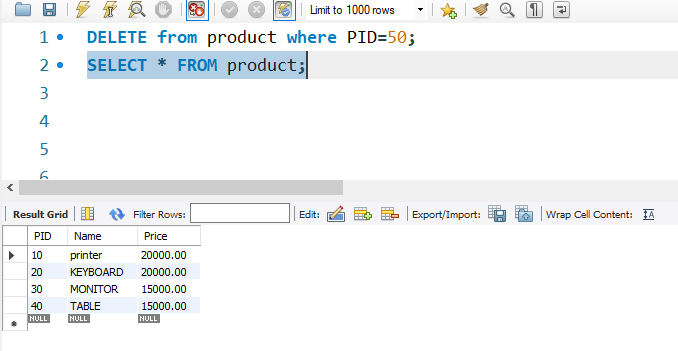
**Execution in MYSQL WORKBENCH:-**

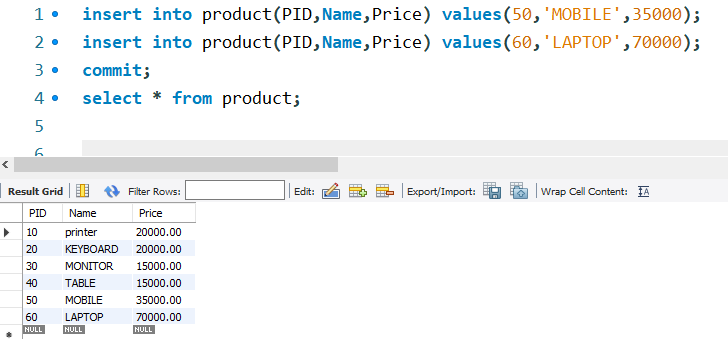
****

****

****

****

****

****

**COMMIT: -** On execution of this command all changes to the database made by you are made permanent and cannot be undone.

Commit is used to save the changes or transaction permanently.

**ROLLBACK: -** A group of related SQL commands that all have to complete successfully or otherwise

Rollback is used to undo the transaction entirely.

**SAVEPOINT:-**

**Syntax: savepoint** <savepoint name>;

Savepoint is a point where u can rollback to a point.

**Varchar: -** is a datatypes in SQL that holds characters of variable length. Varchar and Varchar2 are data types used in databases to hold character strings for specific columns (fields). This data type stores character strings of up to 255 bytes in a variable-length. The data can consist of letters, numbers, and symbols. It uses dynamic memory location.

The data can consist of letters, numbers, and symbols.

**Not Null: -** The NOT NULL constraint is used to ensure that a given column of a table is never assigned the null value. Which means that you cannot insert or update a record without adding a value to this field**.**

**Float (2, 2): -** means "use two digits, of which two are used after the decimal point".

**Primary key: -** A primary key is the column or columns that contain values that uniquely identify each row in a table.

**Constraints: - Constraints** in DBMS are the restrictions that are applied to data or operations on the data. This means that constraints allow only a particular kind of data to be inserted in the database or only some particular kind of operations to be performed on the data in the database.

**Program 2**

**Altering Table, Dropping/Truncating/Renaming a table, backing up/Restoring a Database.**

**Step 1: Create LIB table by properly specifying the constraint.**

**SQL> Create table LIB**

**(BID char (8) primary key,**

**TITLE char (20) Not Null,**

**AUTHOR char (20),**

**PUBLICATION char (20),**

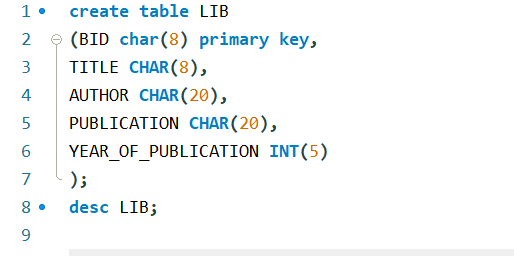
**YEAR\_OF\_PUBLICATION int (5)**

**);**

**Table created**

**SQL> DESC LIB;**

**Execution: -**

****

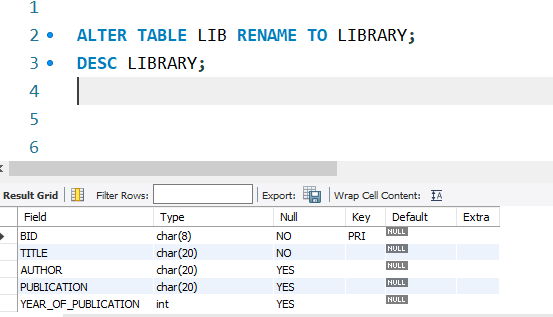
**Step 2: Rename the LIB as LIBRARY**

**SQL>ALTER TABLE LIB RENAME TO LIBRARY**

**Table altered.**

**SQL>DESC LIBRARY;**

**Execution:-**

****

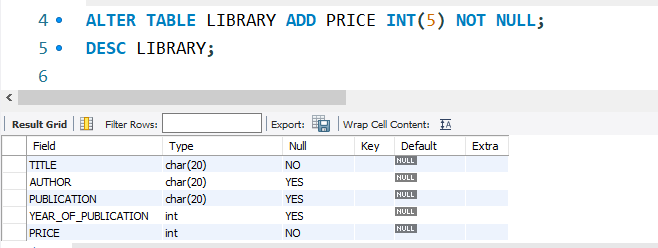
**Step 3:- Add a new column price with not null constraint to the existing table Library.**

**SQL> ALTER TABLE LIBRARY ADD PRICE INT (5) NOT NULL;**

**Table altered.**

**SQL> DESC LIBRARY;**

**Execution : -**

****

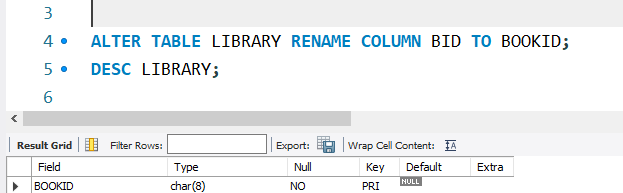
**Step 4:- RENAME THE BID TO BOOKID IN THE LIBRARY TABLE.**

**SQL> ALTER TABLE LIBRARY RENAME COLUMN BID TO BOOKID;**

**Table altered.**

**SQL> DESC LIBRARY;**

**Execution:-**

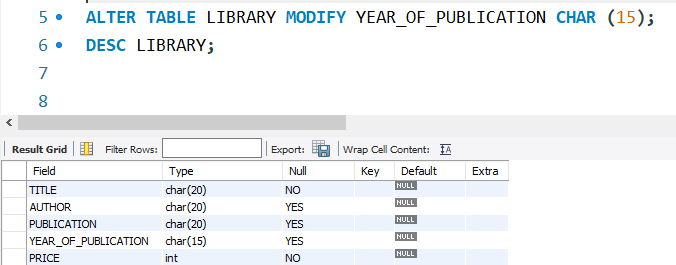
****

**Step 5:- Change the data type of the column year\_Of\_publication as text with size 15.**

**SQL> ALTER TABLE LIBRARY MODIFY YEAR\_OF\_PUBLICATION CHAR (15);**

**Table altered.**

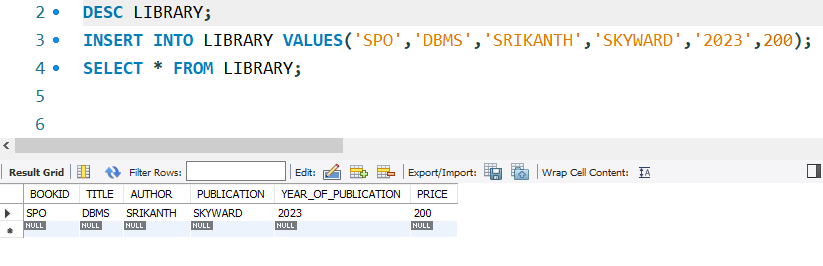
**Executions: -**

****

**Step 6:- Insert data into Library table**

**SQL> INSERT INTO LIBRARY VALUES (‘SPO’,’DBMS’,’SKYWARD PUBLISHERS’,’2023’, 200);**

**1 rows inserted.**

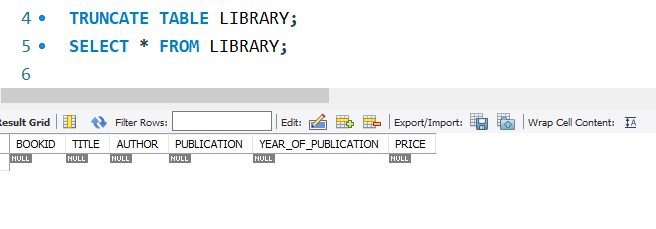
****

**Step 7:- Truncate table to delete Library**

**SQL> TRUNCATE TABLE LIBRARY;**

**Table Library Truncated.**

**Executions:-**

****

**Step 8:- Drop table**

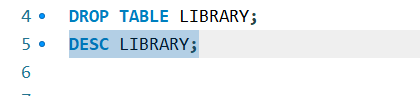
**SQL> DROP TABLE LIBRAY;**

**Table Dropped.**

**SQL> DESC LIBRARY;**

**ERROR; Object Library does not exist.**

**Execution:-**

****

****