Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

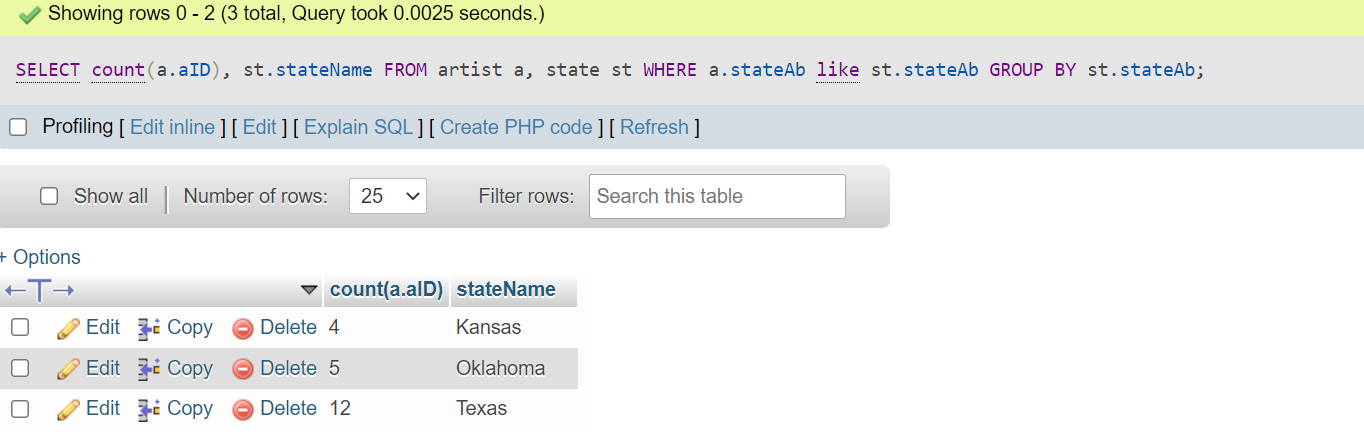
Description automatically generated

PART 2)

i)

SELECT COUNT(A.AID), ST.STATENAME FROM ARTIST A, STATE ST

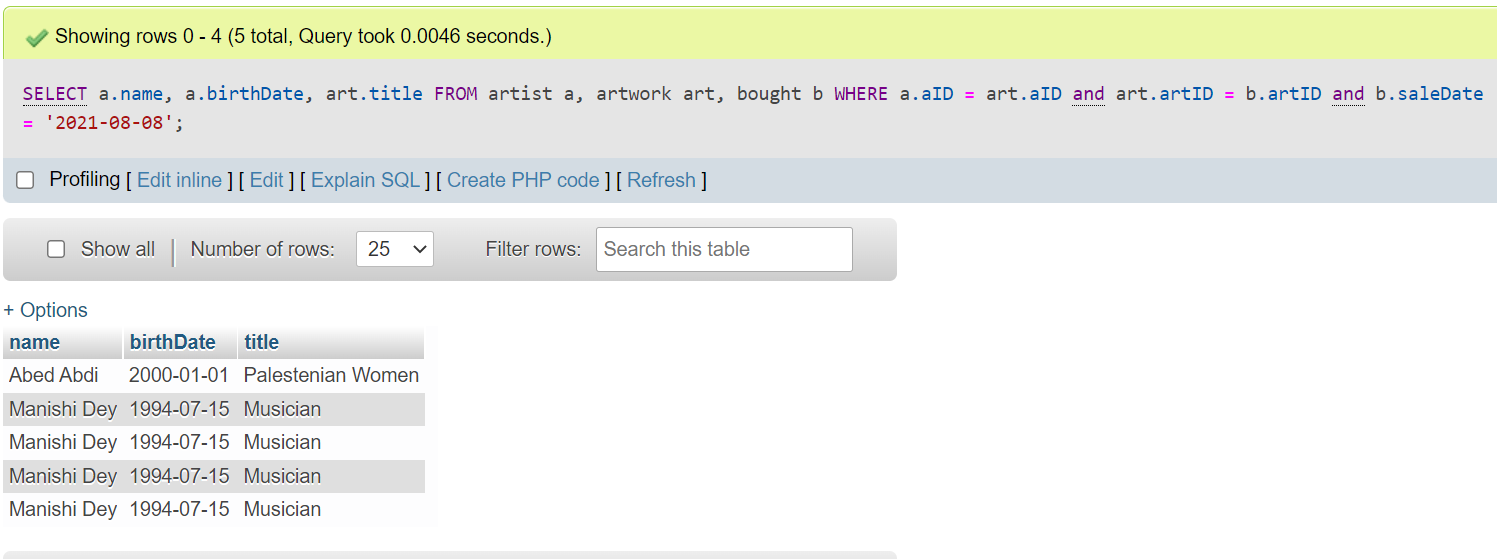
WHERE A.STATEAB LIKE ST.STATEAB GROUP BY ST.STATEAB



ii)

SELECT A.NAME, A.BIRTHDATE, ART.TITLE FROM ARTIST A, ARTWORK ART, BOUGHT B

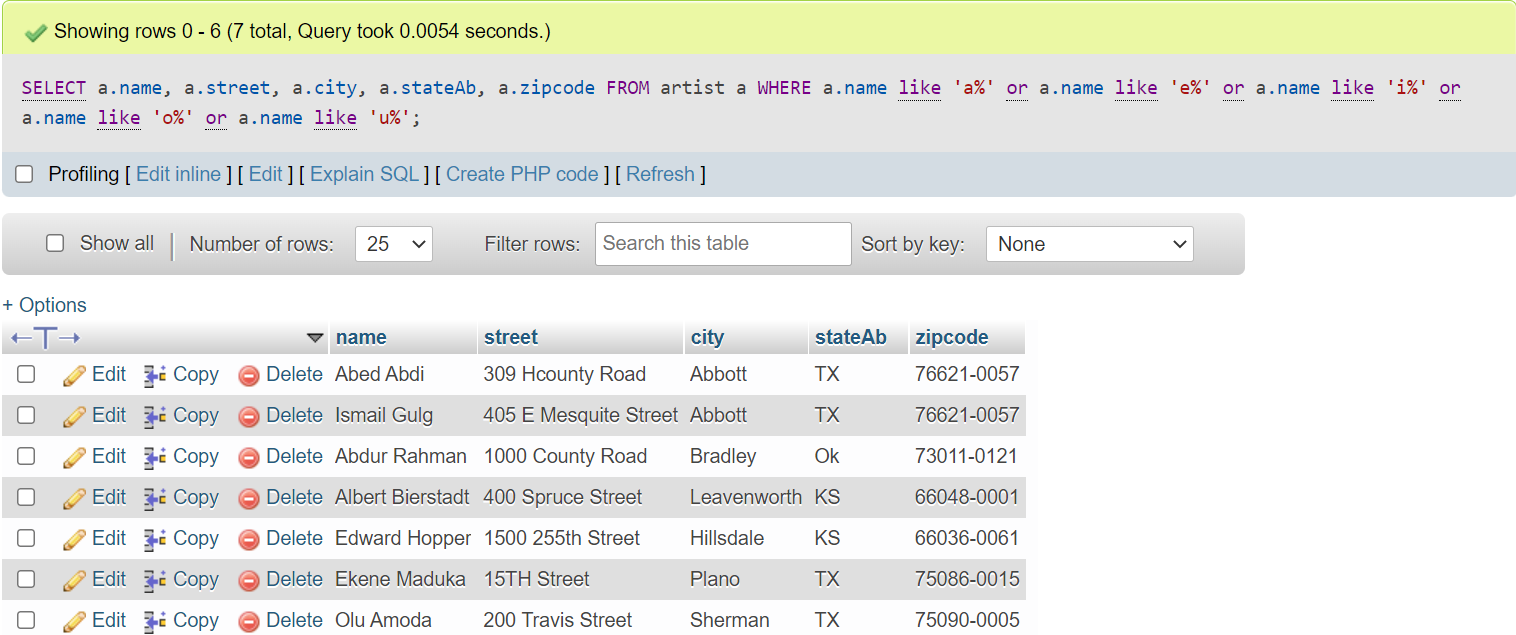
WHERE A.AID = ART.AID AND ART.ARTID = B.ARTID AND B.SALEDATE = '2021-08-08'



iii)

SELECT A.NAME, A.STREET, A.CITY, A.STATEAB, A.ZIPCODE FROM ARTIST A

WHERE A.NAME LIKE 'A%' OR A.NAME LIKE 'E%' OR A.NAME LIKE 'I%' OR A.NAME LIKE 'O%' OR A.NAME LIKE 'U%'

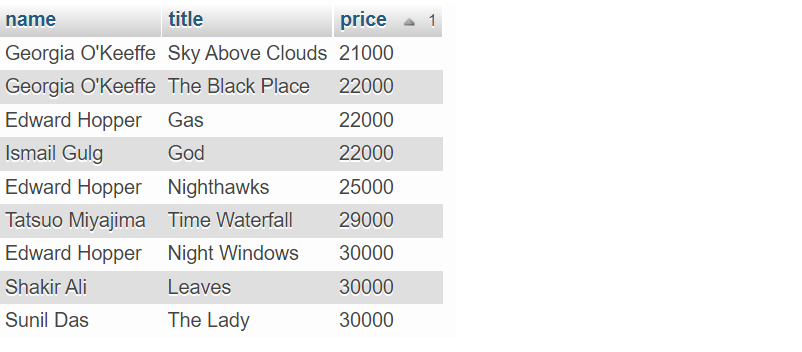


iv)

SELECT A.NAME, ART.TITLE, ART.PRICE FROM ARTIST A, ARTWORK ART

WHERE A.AID = ART.AID AND ART.PRICE > 20000 AND ART.PRICE < 40000

ORDER BY ART.PRICE



v)

SELECT ART.TITLE, ART.PRICE, A.NAME FROM ARTIST A, ARTWORK ART

WHERE A.AID = ART.AID ORDER BY ART.PRICE LIMIT 10

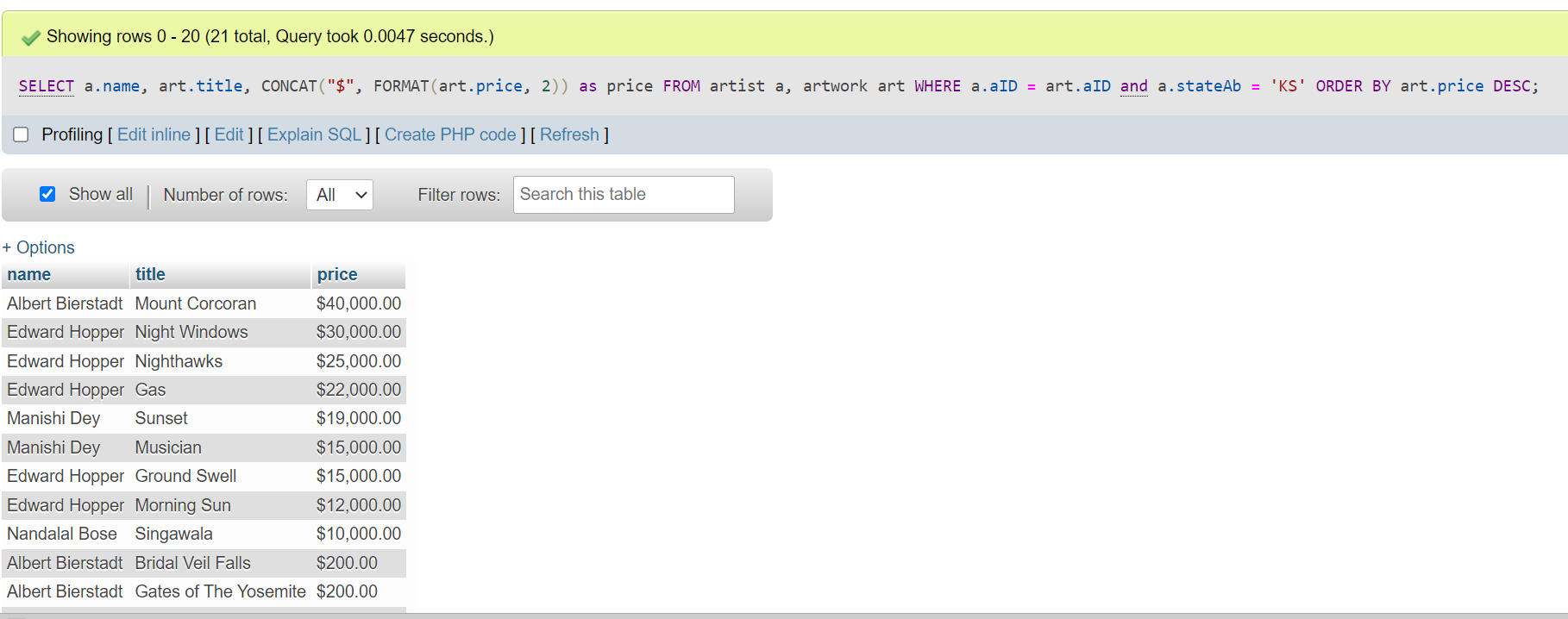


vi)

SELECT A.NAME, ART.TITLE, CONCAT("$", FORMAT(ART.PRICE, 2)) AS PRICE

FROM ARTIST A, ARTWORK ART WHERE A.AID = ART.AID AND A.STATEAB = 'KS'

ORDER BY ART.PRICE DESC





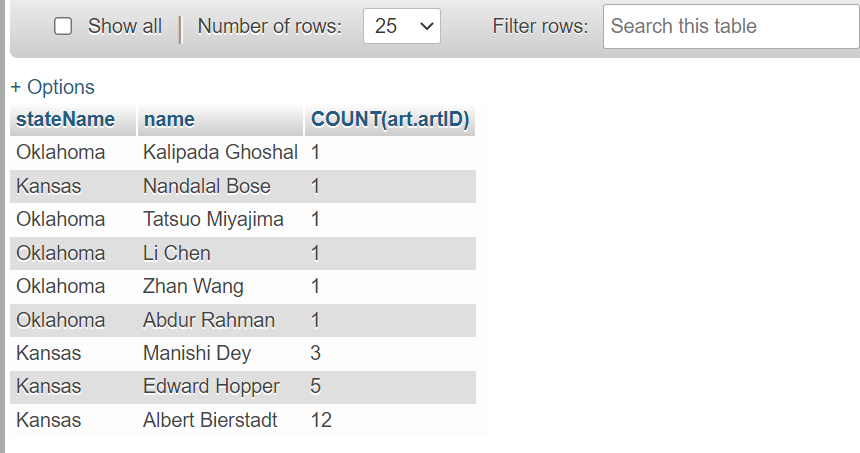
vii)

SELECT ST.STATENAME, A.NAME, COUNT(ART.ARTID)

FROM ARTIST A, ARTWORK ART, STATE ST

WHERE A.STATEAB = ST.STATEAB AND A.AID = ART.AID AND ST.STATENAME NOT IN ('TEXAS')

GROUP BY A.AID ORDER BY COUNT(ART.ARTID);



viii)

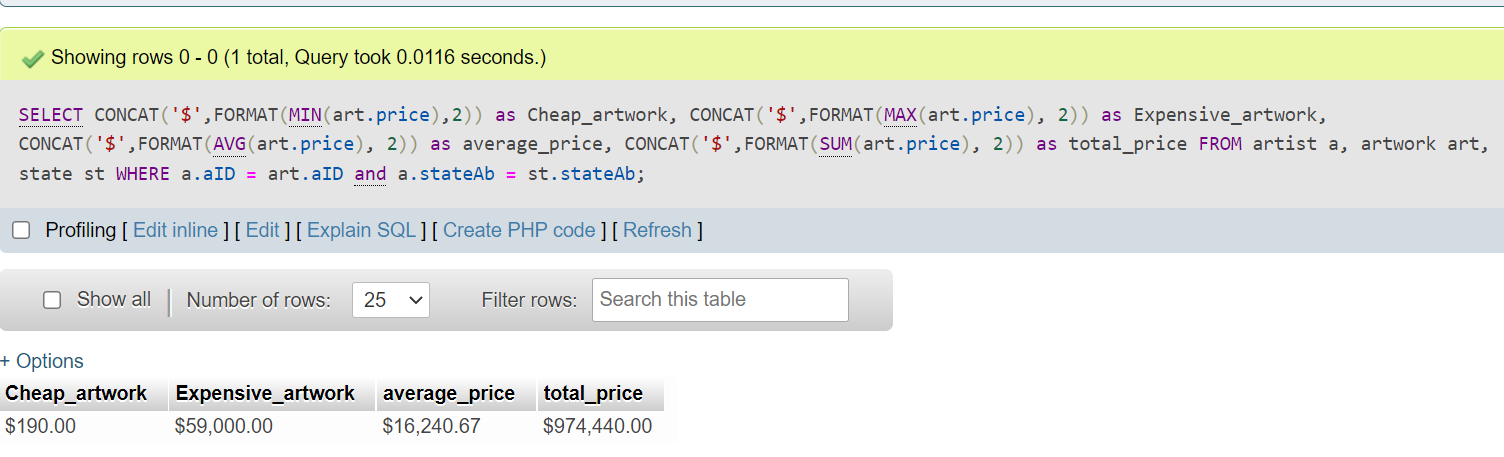
SELECT CONCAT('$',FORMAT(MIN(ART.PRICE),2)) AS CHEAP\_ARTWORK, CONCAT('$',FORMAT(MAX(ART.PRICE), 2)) AS EXPENSIVE\_ARTWORK,

CONCAT('$',FORMAT(AVG(ART.PRICE), 2)) AS AVERAGE\_PRICE,

CONCAT('$',FORMAT(SUM(ART.PRICE), 2)) AS TOTAL\_PRICE

FROM ARTIST A, ARTWORK ART, STATE ST

WHERE A.AID = ART.AID AND A.STATEAB = ST.STATEAB

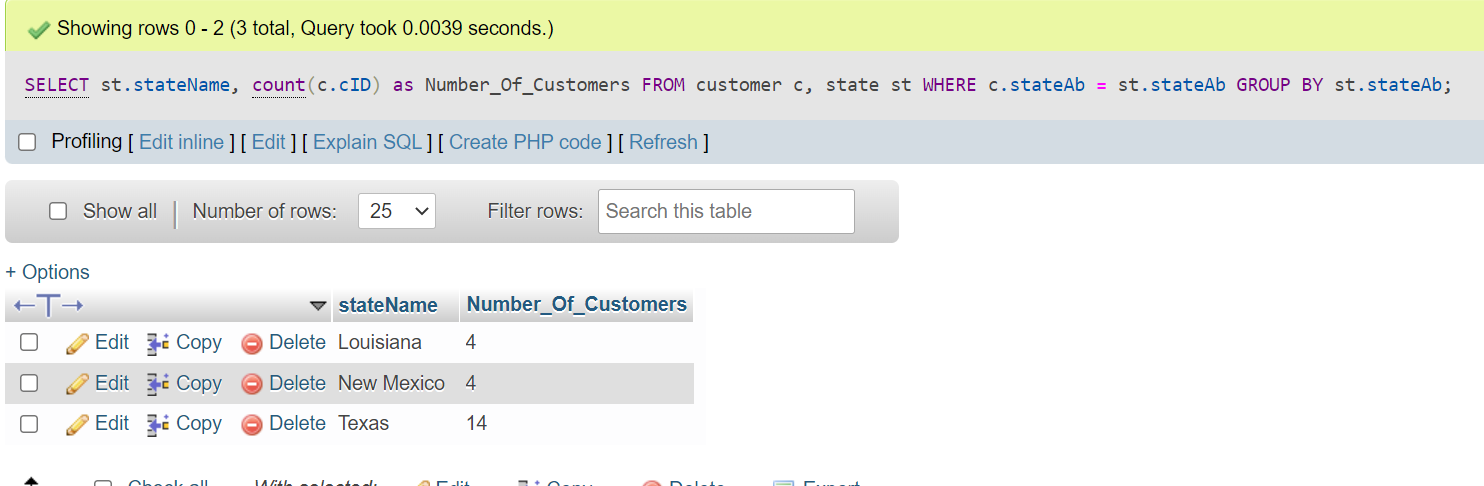


ix)

SELECT ST.STATENAME, COUNT(C.CID) AS NUMBER\_OF\_CUSTOMERS

FROM CUSTOMER C, STATE ST WHERE C.STATEAB = ST.STATEAB

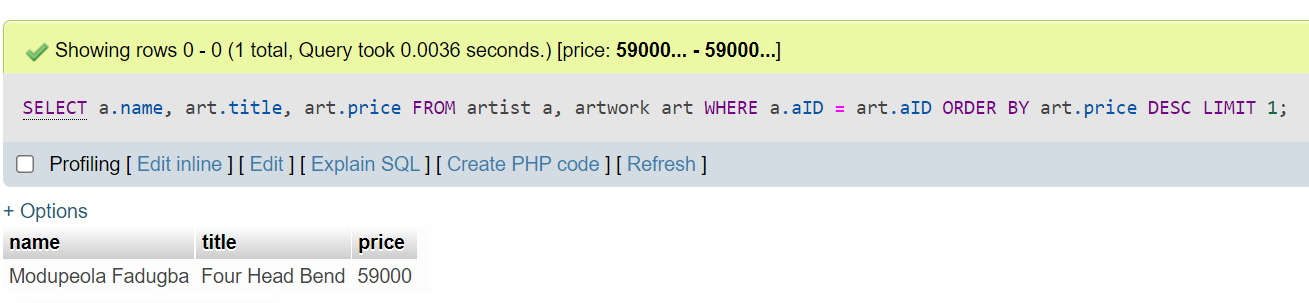
GROUP BY ST.STATEAB



x)

SELECT A.NAME, ART.TITLE, ART.PRICE FROM ARTIST A, ARTWORK ART

WHERE A.AID = ART.AID ORDER BY ART.PRICE DESC LIMIT 1

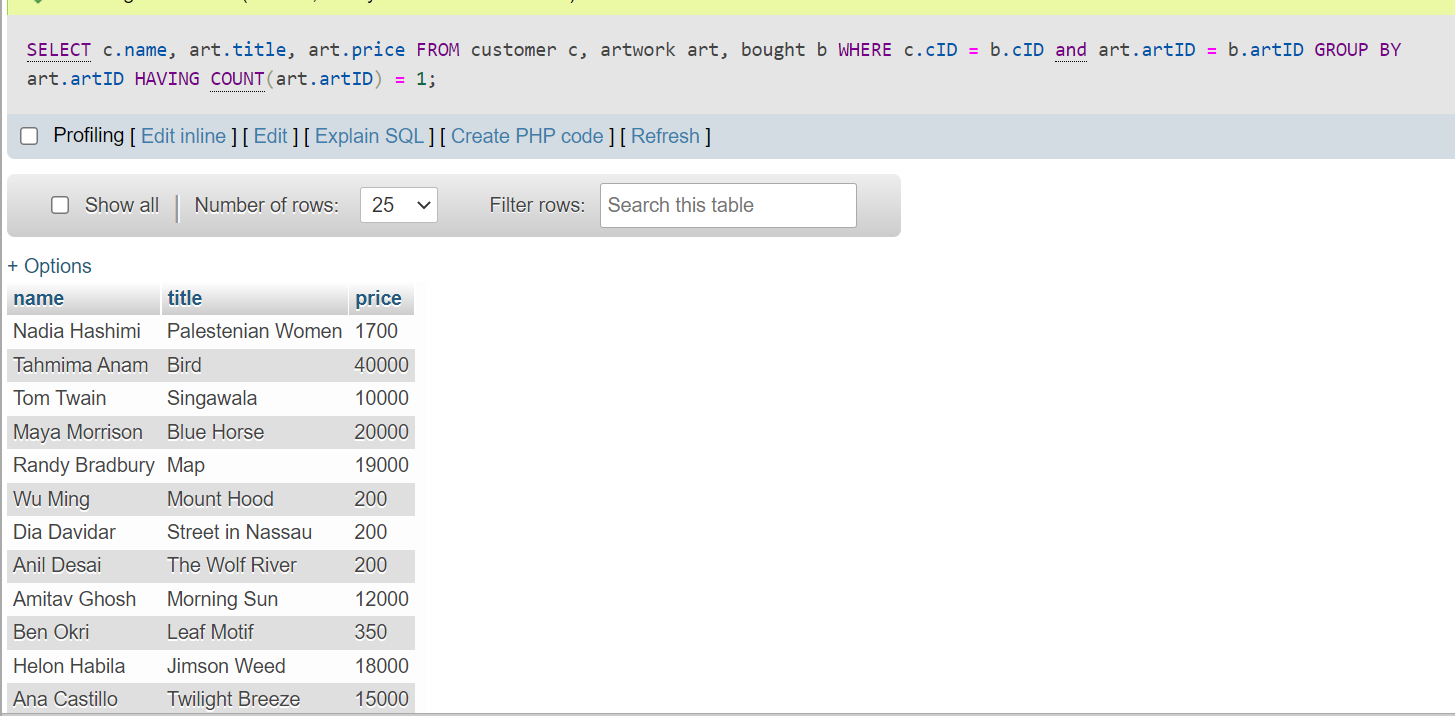


xi)

SELECT C.NAME, ART.TITLE, ART.PRICE FROM CUSTOMER C, ARTWORK ART, BOUGHT B

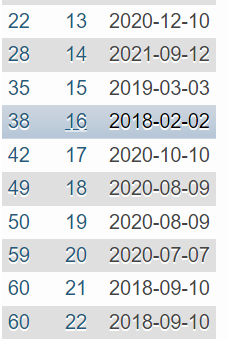
WHERE C.CID = B.CID AND ART.ARTID = B.ARTID GROUP BY ART.ARTID

HAVING COUNT(ART.ARTID) = 1

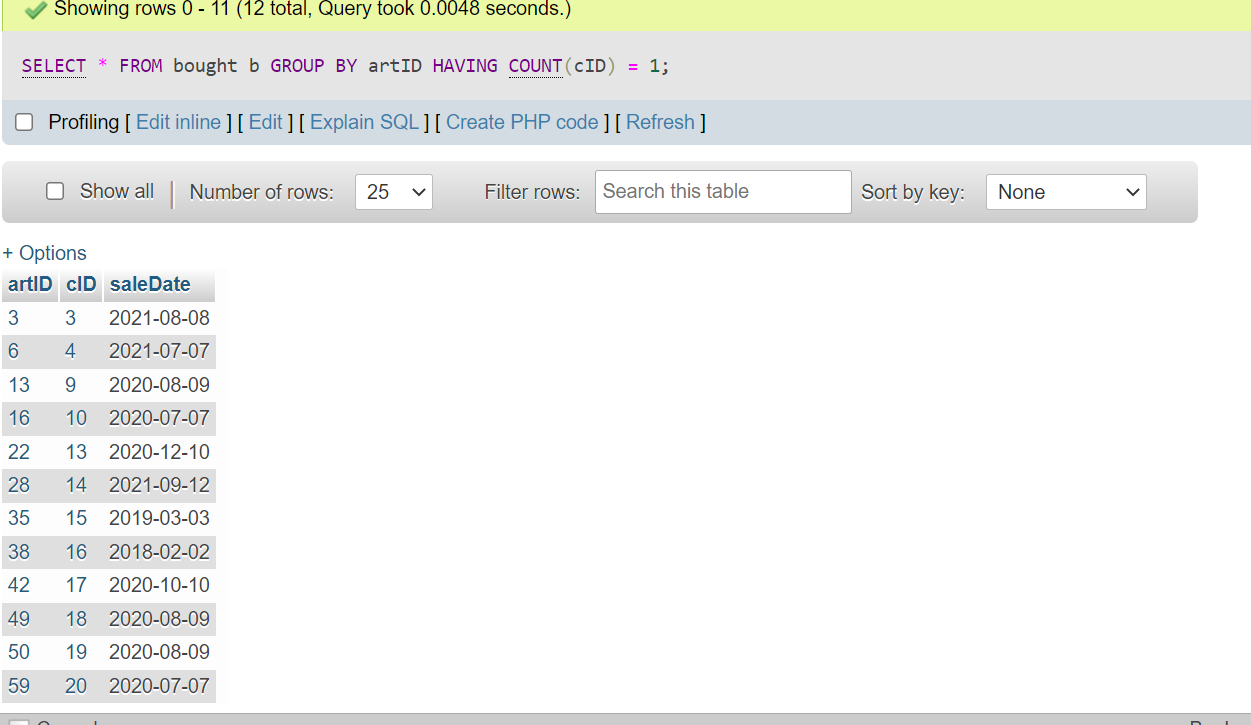


How Verified:

IN TABLE "BOUGHT" CHECK ALL ARTWORK ID'S(ARTID) WHICH HAVE ONLY ONE CUSTOMERID(CID)



WE CAN SEE ARTID 3,6,13,16,22,28,35,38,42,49,50,59 ARE ONLY BOUGHT BY ONE CUSTOMER



THEN IN TABLE "ARTWORK" CHECK RECORDS WITH THOSE ID'S AND WILL SEE THAT TITLES WE EXPECTED AND SHOWN AFTER RUNNING QUERY MATCH.

SELECT C.NAME, ART.TITLE, ART.PRICE FROM ARTWORK ART, BOUGHT B, CUSTOMER C

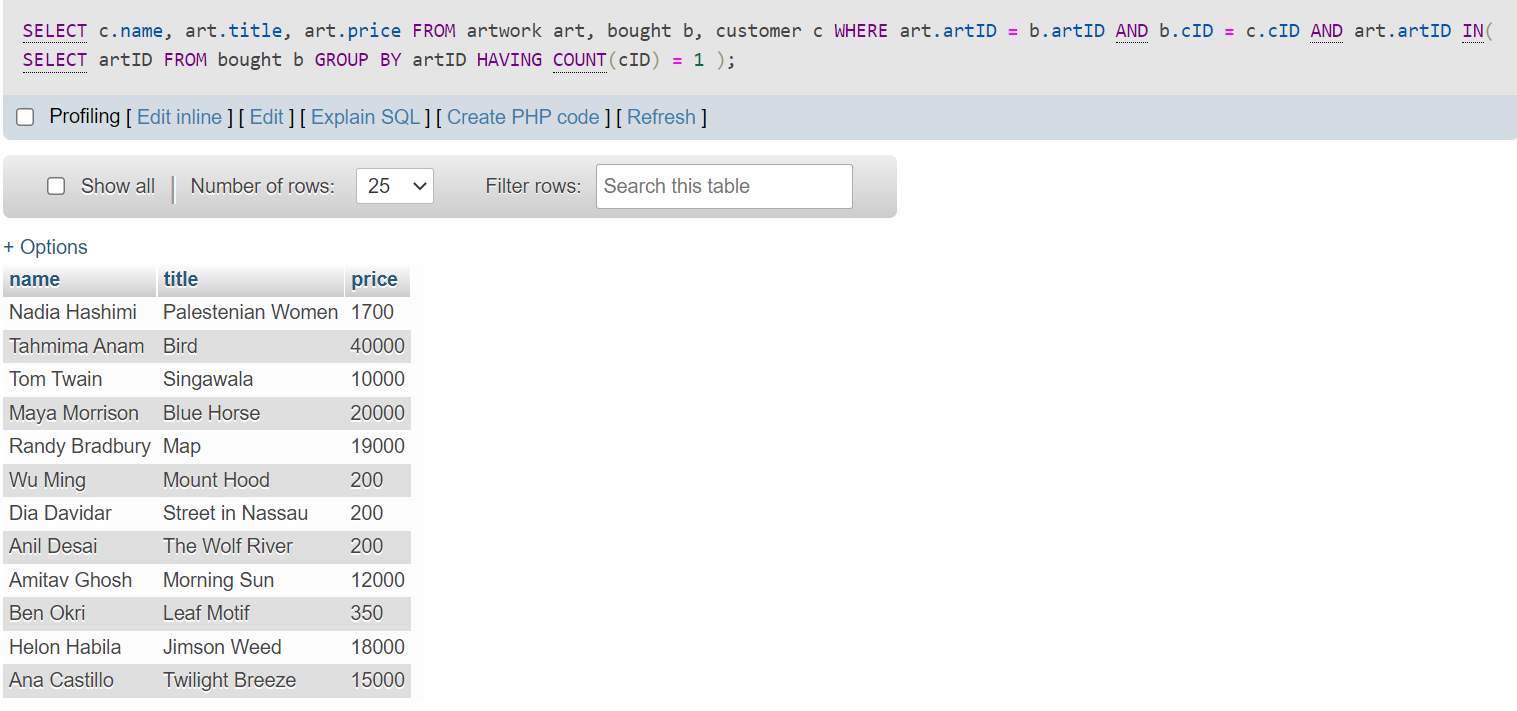
WHERE ART.ARTID = B.ARTID AND B.CID = C.CID AND

ART.ARTID IN(

SELECT ARTID FROM BOUGHT B

GROUP BY ARTID

HAVING COUNT(CID) = 1)



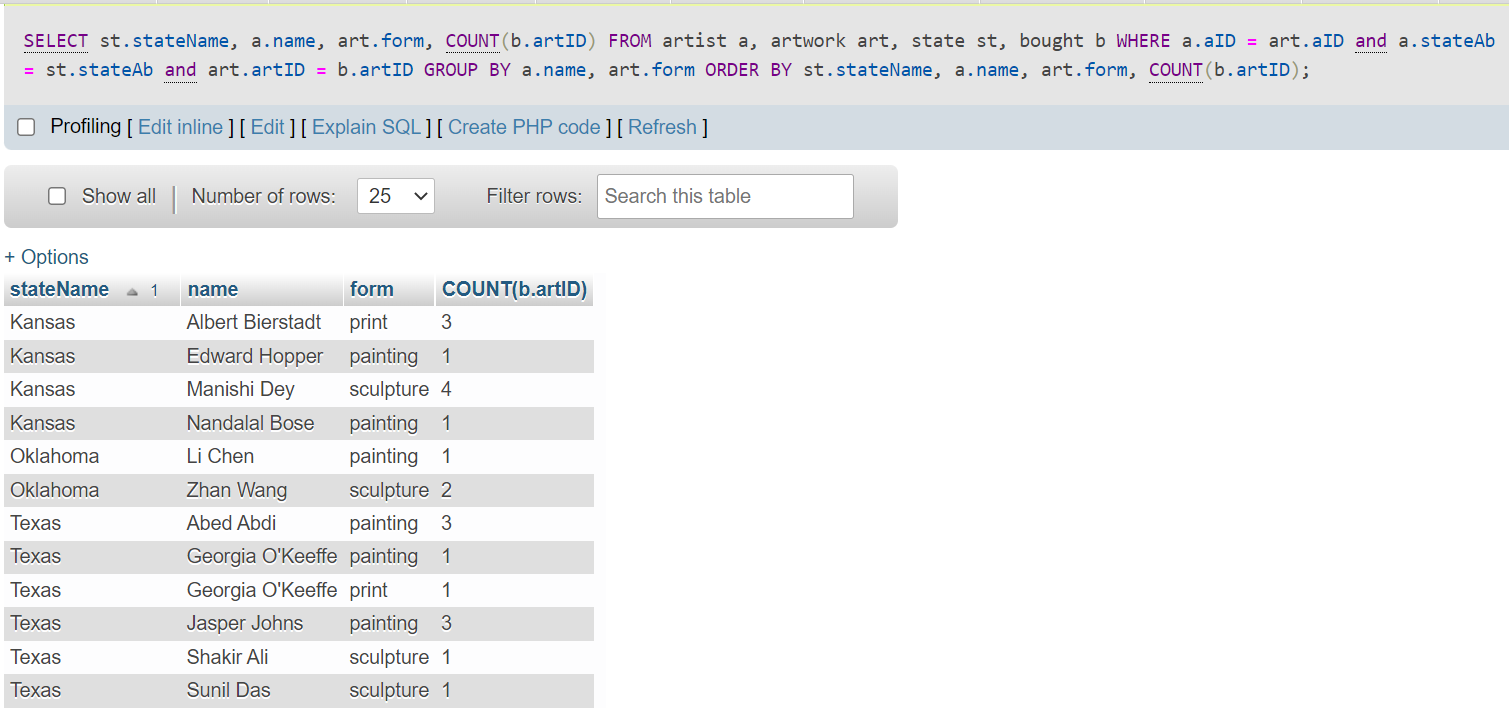
xii)

SELECT ST.STATENAME, A.NAME, ART.FORM, COUNT(B.ARTID)

FROM ARTIST A, ARTWORK ART, STATE ST, BOUGHT B

WHERE A.AID = ART.AID AND A.STATEAB = ST.STATEAB AND ART.ARTID = B.ARTID

GROUP BY A.NAME, ART.FORM ORDER BY ST.STATENAME, A.NAME, ART.FORM, COUNT(B.ARTID)



HOW VERIFIED:

FROM 'ARTWORK' TABLE, GET ALL ARTIST ID'S(ARTWORKS ARE ACQUIRED BY GALLERY FROM ARTISTS) ALONG WITH NUMBER OF RECORDS WITH SAME ARTISTID(AID) GROUPED BY SAME ARTFORM(FORM IN ARTWORK) THAT WERE SOLD(RECORD MUST BE IN BOUGHT)



FOR EACH ARTIST ID(AID) IN TABLE 'ARTIST', CHECK RECORDS MATCHING IN 'ARTIST'

SELECT A.NAME, ST.STATENAME FROM ARTIST A, STATE ST

WHERE A.STATEAB = ST.STATEAB AND A.AID IN(1,12,13,14,10,20,6,7,3,9,21)



FOR EACH OF THIS ARTIST CHECK STATE THEY BELONG TO BASED IN FOREIGN ATTRIBUTE STATEAB IN TABLE 'STATE'

NOW VERIFY COUNT OF ARTWORKS BY AN ARTIST GROUPED UNDER ARTFORM AND ALSO STATE THEY ARE FORM AND WE SEE THAT RESULTS ON RUNNING QUERY MATCH THAT

XIII)

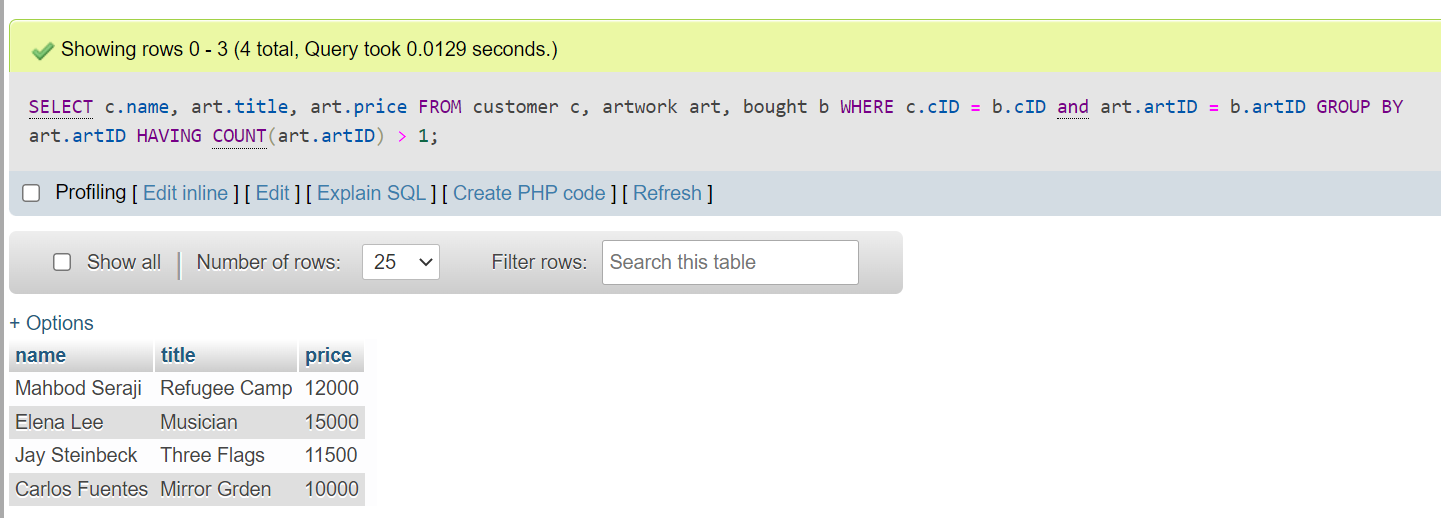
SELECT C.NAME, ART.TITLE, ART.PRICE

FROM CUSTOMER C, ARTWORK ART, BOUGHT B

WHERE C.CID = B.CID AND ART.ARTID = B.ARTID

GROUP BY ART.ARTID

HAVING COUNT(ART.ARTID) > 1



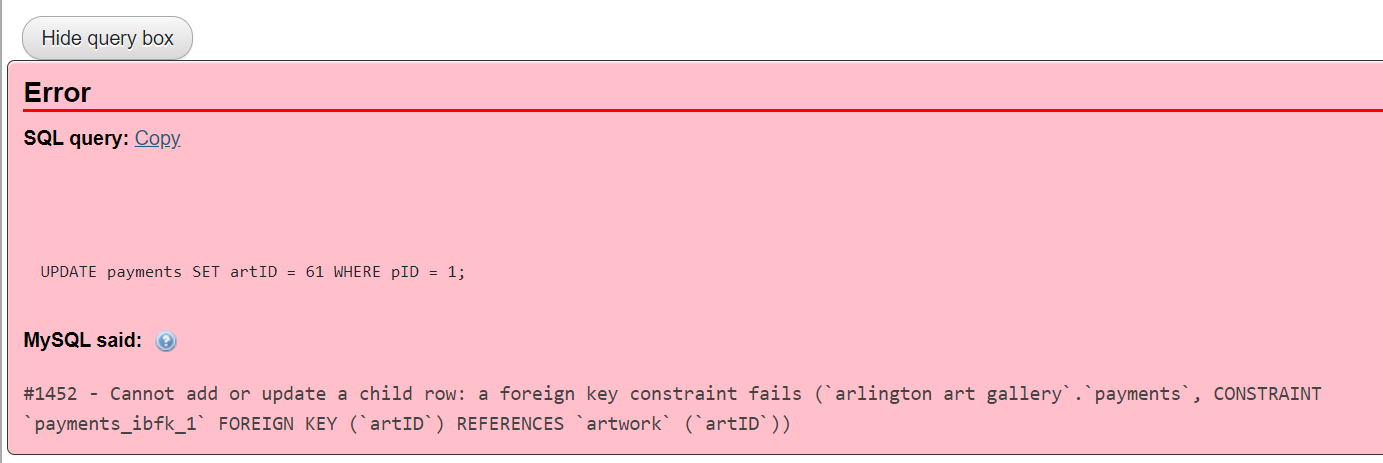
XIV)

DELETE FROM ARTIST WHERE AID = 1



xv)

UPDATE PAYMENTS SET ARTID = 61 WHERE PID = 1



xvi)

**KEY CONSTRAINT**

->INSERT INTO ARTIST(AID, NAME, BIRTHDATE, DEATHDATE, COMMISSION, STREET, CITY, STATEAB, ZIPCODE) VALUES (21, 'JOHN DOE', '2022-03-22', '2022-03-30', 20, 'SLAUGHTER', 'ARLINGTON', 'TX','76011')



**-->**

**ENTITY INTEGRITY**

->INSERT INTO ARTIST(AID, NAME, BIRTHDATE, DEATHDATE, COMMISSION, STREET, CITY, STATEAB, ZIPCODE) VALUES (NULL, 'JOHN DOE', '2022-03-22', '2022-03-30', 20, 'SLAUGHTER', 'ARLINGTON', 'TX','76011')



**-->**

**REFERENTIAL INTEGRITY CONSTRAINT**

INSERT INTO ARTIST(AID, NAME, BIRTHDATE, DEATHDATE, COMMISSION, STREET, CITY, STATEAB, ZIPCODE) VALUES (22, 'JOHN DOE', '2022-03-22', '2022-03-30', 20, 'SLAUGHTER', 'ARLINGTON', 'NY','76011')

