Database Concepts Assignment 5

Instructions:

1. Please type the answers below the questions directly. You may insert tables or figures. Scans of handwritten papers are not acceptable.
2. When it is done, rename the file to firstname-lastname.docx and submit it online by the deadline.
3. Academic integrity is strictly reinforced. Detected plagiarized works will receive zero points and potentially a failure of the whole course.

Problem description:

James River Jewelry is a small jewelry shop. While James River Jewelry does sell typical jewelry purchased form jewelry vendors, including such items as rings, necklaces, earrings, and watches, it specializes in hard-to-find Asian jewelry. Although some Asian jewelry is manufactured jewelry purchased from vendors in the same manner as the standard jewelry is obtained, many of the Asian jewelry pieces are often unique single items purchased directly from the artisan who created the piece (the term "manufactured" would be an inappropriate description of these pieces). James River Jewelry has a small but loyal clientele, and it wants to further increase customer loyalty by creating a frequent buyer program. In this program, after every 10 purchases, a customer will receive a credit equal to 50 percent of the average of his or her 10 most recent purchases. This credit must be applied to the next (or 11th) purchase.

Assume that James River designs a database with the following tables.

CUSTOMER (CustomerID, LastName, FirstName, Phone, EmailAddress)

PURCHASE (InvoiceNumber, InvoiceDate, PreTaxAmount, *CustomerID*)

PURCHASE_ITEM (*InvoiceNumber*, InvoiceLineNumber, *ItemNumber*, RetailPrice)

ITEM (ItemNumber, ItemDescription, Cost, ArtistLastName, ArtistFirstName)

The referential integrity constraints are:

CustomerID in PURCHASE must exist in CustomerID of CUSTOMER

InvoiceNumber in PURCHASE_ITEM must exist in InvoiceNumber of PURCHASE

ItemNumber in PURCHASE_ITEM must exist in ItemNumber of ITEM

Assume that CustomerID of CUSTOMER, ItemNumber of ITEM, and InvoiceNumber of PURCHASE are all surrogate keys with values as follows:

CustomerID	***Start at 1***	***Increment by 1***
InvoiceNumber	***Start at 1001***	***Increment by 1***
ItemNumber	***Start at 1***	***Increment by 1***

Write SQL statements and answer questions for this database as follows, based on the database you created in assignment 3.

C. Write SQL statements to list all columns for all tables (5 points).

Answer:

```
SELECT * FROM CUSTOMER;
SELECT * FROM ITEM;
SELECT * FROM PURCHASE_ITEM;
SELECT * FROM PURCHASE;
```

D. Write an SQL statement to list ItemNumber and ItemDescription for all items that cost more than $100. (5 points)

Answer:

```
SELECT ItemNumber, ItemDescription FROM ITEM
WHERE Cost > 100.00;
```

ItemNumber	ItemDescription
1	Gold Bracelet
2	Gold Necklace
4	Gold Bracelet
5	Silver Necklace
9	Gold Necklace
13	Gold Necklace
15	Gold Bracelet
18	Gold Bracelet
19	Silver Necklace
NULL	NULL

*E. Write an SQL statement to list LastName and FirstName of customers who have made at least one purchase with PreTaxAmount greater than $200. Use a **subquery**. (10 points)*

Answer:

```
SELECT LastName, FirstName
FROM CUSTOMER
WHERE CustomerID IN (
        SELECT CustomerID
        FROM PURCHASE
        WHERE PreTaxAmount > 200.00
);
```

	LastName	FirstName
▶	Stanley	Elizabeth
	Price	Fred
	Romez	Ricardo

*E. Write an SQL statement to list LastName and FirstName of customers who have made at least one purchase with PreTaxAmount greater than $200. Use a **subquery**. (10 points)*

Answer:

```
SELECT DISTINCT C.LastName, C.FirstName
FROM CUSTOMER C
INNER JOIN PURCHASE P ON C.CustomerID = P.CustomerID
WHERE P.PreTaxAmount > 200.00;
```

	LastName	FirstName
▶	Price	Fred
	Romez	Ricardo
	Stanley	Elizabeth

Using DISTINCT: If you use DISTINCT, you ensure that each customer appears in the result set only once, regardless of how many qualifying purchases they have made. This is useful when you want a list of unique customers who meet the criteria, and you don't want duplicate customer entries.

Not using DISTINCT: If you omit DISTINCT, the query may return multiple rows for the same customer if that customer has made multiple purchases with PreTaxAmount greater than $200. In this case, you'll get a result set that includes duplicate customer entries for those who meet the criteria.

G. Write an SQL statement to list LastName and FirstName of customers who have purchased an item that costs more than $50. Use a subquery. (10 points)

Answer:

```
SELECT LastName, FirstName
FROM CUSTOMER C
WHERE EXISTS (
   SELECT 1
   FROM PURCHASE P
   WHERE P.CustomerID = C.CustomerID
   AND EXISTS (
      SELECT 1
      FROM PURCHASE_ITEM PI
      WHERE PI.InvoiceNumber = P.InvoiceNumber
      AND EXISTS (
         SELECT 1
```

```
        FROM ITEM I
        WHERE I.ItemNumber = PI.ItemNumber
        AND I.Cost > 50
      )
   )
);
```

	LastName	FirstName
▶	Stanley	Elizabeth
	Price	Fred
	Romez	Ricardo
	Jackson	Samantha

*H. Answer **Question G** but use a **JOIN**. What are the consequences of using (or not using) the DISTINCT keyword in this version of the query? (10 points)*

Answer:

```
SELECT LastName, FirstName
FROM CUSTOMER
WHERE CustomerID IN (
   SELECT DISTINCT C.CustomerID
   FROM CUSTOMER C
   JOIN PURCHASE P ON C.CustomerID = P.CustomerID
   JOIN PURCHASE_ITEM PI ON P.InvoiceNumber = PI.InvoiceNumber
   JOIN ITEM I ON PI.ItemNumber = I.ItemNumber
   WHERE I.Cost > 50
);
```

	LastName	FirstName
▶	Stanley	Elizabeth
	Price	Fred
	Romez	Ricardo
	Jackson	Samantha

Using DISTINCT:

If you use the DISTINCT keyword, the query ensures that each customer appears in the result set only once, regardless of how many qualifying purchases they have made with items costing more than $50. This is useful when you want a list of unique customers who meet the criteria, and you don't want duplicate customer entries.

Not using DISTINCT:

If you omit the DISTINCT keyword, the query may return multiple rows for the same customer if that customer has made multiple qualifying purchases with items costing more than $50.

In this case, you'll get a result set that includes duplicate customer entries for those who meet the criteria.
The result set will include each qualifying purchase separately for customers who have made multiple such purchases.

*I. Write an SQL statement to list LastName and FirstName of customers who have purchased an item that was created by an artist with a LastName that begins with the letter "J". Use a **subquery**. (10 points)*

Answer:

```
SELECT LastName, FirstName
FROM CUSTOMER
WHERE CustomerID IN (
   SELECT CustomerID
   FROM PURCHASE
   WHERE InvoiceNumber IN (
      SELECT InvoiceNumber
      FROM PURCHASE_ITEM
      WHERE ItemNumber IN (
         SELECT ItemNumber
         FROM ITEM
         WHERE ArtistLastName LIKE 'J%'
      )
   )
);
```

	LastName	FirstName
▶	Stanley	Elizabeth
	Becky	Linda
	Birch	Pamela

*J. Answer **Question I** but use a **JOIN**. What are the consequences of using (or not using) the DISTINCT keyword in this version of the query? (10 points)*

Answer:

```
SELECT DISTINCT C.LastName, C.FirstName
FROM CUSTOMER C
INNER JOIN PURCHASE P ON C.CustomerID = P.CustomerID
INNER JOIN PURCHASE_ITEM PI ON P.InvoiceNumber = PI.InvoiceNumber
INNER JOIN ITEM I ON PI.ItemNumber = I.ItemNumber
```

WHERE LEFT(I.ArtistLastName, 1) = 'J';

LastName	FirstName
Stanley	Elizabeth
Becky	Linda
Birch	Pamela

Using DISTINCT: If you use DISTINCT, the query ensures that each customer appears in the result set only once, regardless of how many qualifying purchases they have made. This is useful when you want a list of unique customers who meet the criteria, and you don't want duplicate customer entries.

Not using DISTINCT: If you omit DISTINCT, the query may return multiple rows for the same customer if that customer has made multiple qualifying purchases of items created by artists with LastName starting with "J." In this case, you'll get a result set that includes duplicate customer entries for those who meet the criteria.

*K. Write an SQL statement to show the CustomerID, LastName, FirstName, and sum of PreTaxAmount for each customer. Use a **JOIN**. (10 points)*

Answer:

SELECT C.CustomerID, C.LastName, C.FirstName, SUM(P.PreTaxAmount) AS TotalPreTaxAmount
FROM CUSTOMER AS C
INNER JOIN PURCHASE AS P ON C.CustomerID = P.CustomerID
GROUP BY C.CustomerID, C.LastName, C.FirstName;

CustomerID	LastName	FirstName	SUM(P.PreTaxAmount)
1	Stanley	Elizabeth	625.00
2	Price	Fred	203.00
3	Becky	Linda	120.00
4	Birch	Pamela	67.00
5	Romez	Ricardo	330.00
6	Jackson	Samantha	72.00

*L. Write an SQL statement to show the sum of PreTaxAmount for each artist (Hint: the result will have only one line per each artist). Use **JOIN** and sort the results by ArtistLastName then ArtistFirstName in ascending order. Note this should include the full PreTaxAmount for any purchase in which the artist had an item, even if other items in the purchase were created by other artists. (10 points)*

***Answer**:*

SELECT
I.ArtistLastName,

```
   I.ArtistFirstName,
   SUM(P.PreTaxAmount) AS TotalPreTaxAmount
FROM
   item AS I
JOIN
   purchase_item AS PI ON I.ItemNumber = PI.ItemNumber
JOIN
   purchase AS P ON PI.InvoiceNumber = P.InvoiceNumber
GROUP BY
   I.ArtistLastName, I.ArtistFirstName
ORDER BY
   I.ArtistLastName ASC, I.ArtistFirstName ASC;
```

	ArtistLastName	ArtistFirstName	TotalPreTaxAmount
▶	Baker	Samantha	533.00
	Baxter	Sam	445.00
	Josephson	Mary	879.00
	Lintz	John	847.00

*M. Write an SQL statement to show the sum of PreTaxAmount for each artist, as in **Question L**, but exclude any purchases with PreTaxAmount over $25. Use **JOIN** and sort the results by ArtistLastName and ArtistFirstName in descending order. (10 points)*

Answer:

```
SELECT I.ArtistLastName, I.ArtistFirstName, I.ItemDescription, SUM(P.PreTaxAmount)
FROM item AS I
JOIN purchase_item AS PI ON I.ItemNumber = PI.ItemNumber
JOIN purchase AS P ON PI.InvoiceNumber = P.InvoiceNumber
WHERE P.PreTaxAmount <= 25.00
GROUP BY I.ArtistLastName, I.ArtistFirstName, I.ItemDescription
ORDER BY I.ArtistLastName DESC, I.ArtistFirstName DESC;
```

	ArtistLastName	ArtistFirstName	ItemDescription	SUM(P.PreTaxAmount)
▶	Josephson	Mary	Bead Earrings	25.00