MODULE 5 ASSIGNMENT 2

CREATING A SALECO RELATIONAL DATABASE

KEVIN JOHNSON

1. **Write a query to count the number of invoices.**

   SELECT COUNT(*) FROM INVOICE;

```
sqlite> SELECT COUNT(*) FROM INVOICE;
8
```

2. **Write a query to count the number of customers with a customer balance over $500.**

   SELECT COUNT(*) FROM CUSTOMER WHERE CUS_BALANCE > 500;

```
sqlite> SELECT COUNT(*) FROM CUSTOMER WHERE CUS_BALANCE > 500;
2
sqlite>
```

3. **Generate a listing of all purchases made by the customers.**

   SELECT CUSTOMER.CUS_CODE, INVOICE.INV_NUMBER, LINE.LINE_NUMBER, LINE.LINE_UNITS, LINE.LINE_PRICE

   FROM CUSTOMER, INVOICE, LINE

   WHERE INVOICE.INV_NUMBER = LINE.INV_NUMBER AND INVOICE.CUS_CODE = CUSTOMER.CUS_CODE

   ORDER BY CUSTOMER.CUS_CODE ASC;

```
sqlite> SELECT CUSTOMER.CUS_CODE, INVOICE.INV_NUMBER, LINE.LINE_NUMBER, LINE.LINE_UNITS, LINE.LINE_PRIC
   ...> FROM CUSTOMER, INVOICE, LINE
   ...> WHERE INVOICE.INV_NUMBER = LINE.INV_NUMBER AND INVOICE.CUS_CODE = CUSTOMER.CUS_CODE
   ...> ORDER BY CUSTOMER.CUS_CODE ASC;
10011|1002|1|2|4.99
10011|1004|1|3|4.99
10011|1004|2|2|9.95
10011|1008|1|5|5.87
10011|1008|2|3|119.95
10011|1008|3|1|9.95
10012|1003|1|1|38.95
10012|1003|2|1|39.95
10012|1003|3|5|14.99
10014|1001|1|1|14.99
10014|1001|2|1|9.95
10014|1006|1|3|6.99
10014|1006|2|1|109.92
10014|1006|3|1|9.95
10014|1006|4|1|256.99
10015|1007|1|2|14.99
10015|1007|2|1|4.99
10018|1005|1|12|5.87
```

4. **Generate the listing of customer purchases including the subtotals for each of the invoice line numbers.**

   SELECT CUSTOMER.CUS_CODE, INVOICE.INV_NUMBER, LINE.LINE_NUMBER, LINE.LINE_UNITS, LINE.LINE_PRICE, LINE.LINE_UNITS * LINE.LINE_PRICE AS LINE_SUBTOTAL

   FROM CUSTOMER, INVOICE, LINE

   WHERE INVOICE.INV_NUMBER = LINE.INV_NUMBER AND INVOICE.CUS_CODE = CUSTOMER.CUS_CODE

   ORDER BY CUSTOMER.CUS_CODE ASC;

```
sqlite> SELECT CUSTOMER.CUS_CODE, INVOICE.INV_NUMBER, LINE.LINE_NUMBER, LINE.LINE_UNITS, LINE.LINE_PRICE, LINE.LINE_UNITS * LINE.LINE_PRICE AS LINE_SUBTOTAL
   ...> FROM CUSTOMER, INVOICE, LINE
   ...> WHERE INVOICE.INV_NUMBER = LINE.INV_NUMBER AND INVOICE.CUS_CODE = CUSTOMER.CUS_CODE
   ...> ORDER BY CUSTOMER.CUS_CODE ASC;
10011|1002|1|2|4.99|9.98
10011|1004|1|3|4.99|14.97
10011|1004|2|2|9.95|19.9
10011|1008|1|5|5.87|29.35
10011|1008|2|3|119.95|359.85
10011|1008|3|1|9.95|9.95
10012|1003|1|1|38.95|38.95
10012|1003|2|1|39.95|39.95
10012|1003|3|5|14.99|74.95
10014|1001|1|1|14.99|14.99
10014|1001|2|1|9.95|9.95
10014|1006|1|3|6.99|20.97
10014|1006|2|1|109.92|109.92
10014|1006|3|1|9.95|9.95
10014|1006|4|1|256.99|256.99
10015|1007|1|2|14.99|29.98
10015|1007|2|1|4.99|4.99
10018|1005|1|12|5.87|70.44
```

5. **List the balance characteristics of the customers who have made purchases during the current invoice cycle – that is, for the customers who appear in the INVOICE table.**

   SELECT CUSTOMER.CUS_CODE, SUM(LINE.LINE_UNITS * LINE.LINE_PRICE) AS TOTAL_PURCHASES

   WHERE INVOICE.INV_NUMBER = LINE.INV_NUMBER AND INVOICE.CUS_CODE = CUSTOMER.CUS_CODE

   GROUP BY CUSTOMER.CUS_CODE

   ORDER BY CUSTOMER.CUS_CODE ASC;

```
sqlite> SELECT CUSTOMER.CUS_CODE, SUM(LINE.LINE_UNITS * LINE.LINE_PRICE) AS TOTAL_PURCHASES
   ...> FROM CUSTOMER, INVOICE, LINE
   ...> WHERE INVOICE.INV_NUMBER = LINE.INV_NUMBER AND INVOICE.CUS_CODE = CUSTOMER.CUS_CODE
   ...> GROUP BY CUSTOMER.CUS_CODE
   ...> ORDER BY CUSTOMER.CUS_CODE ASC;
10011|444.0
10012|153.85
10014|422.77
10015|34.97
10018|70.44
```

6. **Find the listing of customers who did not make purchases during the invoicing period.**

```
SELECT CUSTOMER.CUS_CODE, CUSTOMER.CUS_FNAME, CUSTOMER.CUS_LNAME

FROM CUSTOMER LEFT JOIN INVOICE ON CUSTOMER.CUS_CODE = INVOICE.CUS_CODE

WHERE INVOICE.INV_NUMBER IS NULL

ORDER BY CUSTOMER.CUS_CODE ASC;
```

```
sqlite> SELECT CUSTOMER.CUS_CODE, CUSTOMER.CUS_FNAME, CUSTOMER.CUS_LNAME
   ...> FROM CUSTOMER LEFT JOIN INVOICE ON CUSTOMER.CUS_CODE = INVOICE.CUS_CODE
   ...> WHERE INVOICE.INV_NUMBER IS NULL
   ...> ORDER BY CUSTOMER.CUS_CODE ASC;
10010|Alfred|Ramas
10013|Paul|Olowski
10016|James|Brown
10017|George|Williams
10019|Olette|Smith
```

7. **Create a query to produce the summary of the value of products currently in inventory.**

```
SELECT P_CODE, P_QOH, P_PRICE, P_QOH * P_PRICE AS INVENTORY_VALUE
FROM PRODUCT WHERE P_QOH > 0
ORDER BY P_CODE ASC;

SELECT SUM(P_QOH * P_PRICE) AS TOTAL_VALUE
FROM PRODUCT;
```

```
sqlite> SELECT P_CODE, P_QOH, P_PRICE, P_QOH * P_PRICE AS INVENTORY_VALUE
   ...> FROM PRODUCT WHERE P_QOH > 0
   ...> ORDER BY P_CODE ASC;
11QER/31|8|109.99|879.92
13-Q2/P2|32|14.99|479.68
14-Q1/L3|18|17.49|314.82
1546-QQ2|15|39.95|599.25
1558-QW1|23|43.99|1011.77
2232/QTY|8|109.92|879.36
2232/QWE|6|99.87|599.22
2238/QPD|12|38.95|467.4
23109-HB|23|9.95|228.85
23114-AA|8|14.4|115.2
54778-2T|43|4.99|214.57
89-WRE-Q|11|256.99|2826.89
PVC23DRT|188|5.87|1103.56
SM-18277|172|6.99|1202.28
SW-23116|237|8.45|2002.65
WR3/TT3|18|119.95|2159.1
sqlite> SELECT SUM(P_QOH * P_PRICE) AS TOTAL_VALUE
   ...> FROM PRODUCT;
15084.52
```