

Emily Wantland

CIS 310-01

Professor Karimi

5 April 2020

### Assignment 10

9.

```
SELECT
    COUNT( * )
FROM
    INVOICE;
```

10.

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

11.

```
SELECT
    CUS.CUS_CODE,
    INV.INV_NUMBER,
    INV.INV_DATE,
    PROD.P_DESCRIPT,
    LINE.LINE_UNITS,
    LINE.LINE_PRICE
FROM
    CUSTOMER CUS
    JOIN INVOICE INV ON CUS.CUS_CODE = INV.CUS_CODE
    JOIN LINE LINE ON LINE.INV_NUMBER = INV.INV_NUMBER
    JOIN PRODUCT PROD ON PROD.P_CODE = LINE.P_CODE
ORDER BY
    CUS.CUS_CODE ASC,
    INV.INV_NUMBER ASC,
    PROD.P_DESCRIPT ASC;
```

Emily Wantland

CIS 310-01

Professor Karimi

5 April 2020

12.

```
SELECT
    CUS.CUS_CODE,
    INV.INV_NUMBER,
    PROD.P_DESCRIPT,
    LINE.LINE_UNITS AS 'Units Bought',
    LINE.LINE_PRICE AS 'Unit Price',
    ROUND( LINE.LINE_UNITS * LINE.LINE_PRICE, 2 ) AS 'Subtotal'
FROM
    CUSTOMER CUS
    JOIN INVOICE INV ON CUS.CUS_CODE = INV.CUS_CODE
    JOIN LINE LINE ON LINE.INV_NUMBER = INV.INV_NUMBER
    JOIN PRODUCT PROD ON PROD.P_CODE = LINE.P_CODE
ORDER BY
    CUS.CUS_CODE ASC,
    INV.INV_NUMBER ASC,
    PROD.P_DESCRIPT ASC;
```

13.

```
SELECT
    CUS.CUS_CODE,
    CUS.CUS_BALANCE,
    ROUND( SUM( LINE.LINE_UNITS * LINE.LINE_PRICE ), 2 ) AS 'Total
Purchases'
FROM
    CUSTOMER CUS
    JOIN INVOICE INV ON CUS.CUS_CODE = INV.CUS_CODE
    JOIN LINE LINE ON LINE.INV_NUMBER = INV.INV_NUMBER
GROUP BY
    CUS.CUS_CODE, CUS_BALANCE
ORDER BY
    CUS.CUS_CODE, CUS_BALANCE ASC;
```

Emily Wantland

CIS 310-01

Professor Karimi

5 April 2020

14.

```
SELECT
    CUS.CUS_CODE,
    CUS.CUS_BALANCE,
    ROUND( SUM( LINE.LINE_UNITS * LINE.LINE_PRICE ), 2 ) AS 'Total
Purchases',
    COUNT( INV.CUS_CODE ) AS 'Number of Purchases'
FROM
    CUSTOMER CUS
    JOIN INVOICE INV ON CUS.CUS_CODE = INV.CUS_CODE
    JOIN LINE LINE ON LINE.INV_NUMBER = INV.INV_NUMBER
GROUP BY
    CUS.CUS_CODE, CUS_BALANCE
ORDER BY
    CUS.CUS_CODE, CUS_BALANCE ASC;
```

15.

```
SELECT
    CUS.CUS_CODE,
    CUS.CUS_BALANCE,
    ROUND( SUM( LINE.LINE_UNITS * LINE.LINE_PRICE ), 2 ) AS 'Total
Purchases',
    COUNT( INV.CUS_CODE ) AS 'Number of Purchases',
    ROUND( SUM( LINE.LINE_UNITS * LINE.LINE_PRICE ) / COUNT(
INV.CUS_CODE ), 2 ) AS 'Average Purchase Amount'
FROM
    CUSTOMER CUS
    JOIN INVOICE INV ON CUS.CUS_CODE = INV.CUS_CODE
    JOIN LINE LINE ON LINE.INV_NUMBER = INV.INV_NUMBER
GROUP BY
    CUS.CUS_CODE, CUS_BALANCE
ORDER BY
    CUS.CUS_CODE, CUS_BALANCE ASC;
```

Emily Wantland

CIS 310-01

Professor Karimi

5 April 2020

16.

```
SELECT
    INV.INV_NUMBER,
    ROUND( SUM( LINE.LINE_PRICE * LINE.LINE_UNITS), 2 ) AS 'Invoice Total'
FROM
    INVOICE INV
    JOIN LINE LINE ON INV.INV_NUMBER = LINE.INV_NUMBER
GROUP BY
    INV.INV_NUMBER
ORDER BY
    INV.INV_NUMBER ASC;
```