Assignment 3 Solution

Name: Yanjun Chen, PSU ID: yfc5289

1.

```
SELECT wh_addr_id FROM warehouse
INTERSECT
SELECT cstmr_addr_id FROM customer
WHERE cstmr_secondary_contact IS NULL;
```

2.

```
SELECT litm_inv_id FROM line_item WHERE litm_quantity BETWEEN 15 AND 25;
```

3.

```
SELECT cstmr_primary_contact FROM customer
WHERE cstmr_addr_id BETWEEN 5 AND 10
UNION
SELECT cstmr_secondary_contact FROM customer
WHERE cstmr_addr_id BETWEEN 5 AND 10;
```

4.

```
SELECT cstmr_primary_contact FROM customer
WHERE cstmr_primary_contact IS NOT NULL
INTERSECT
SELECT cstmr_secondary_contact FROM customer
WHERE cstmr_secondary_contact != cstmr_primary_contact
ORDER BY cstmr_primary_contact DESC;
```

5.

```
SELECT litm_inv_id, SUM(litm_quantity)
FROM line_item
GROUP BY litm_inv_id
HAVING SUM(litm_quantity) >= 50;
```

6.

```
SELECT inv_id, COUNT(*) AS total_number, SUM(itm_unit_price * litm_quantity)
AS total_cost
FROM invoice, line_item, item
WHERE inv_payment_status_cd IN ('UNP','DEF')
 AND inv_id = litm_inv_id
 AND litm_itm_id = itm_id
GROUP BY inv_id
HAVING COUNT(*) >= 2
ORDER BY total_cost DESC;
```

7.

```
SELECT DISTINCT c1.cstmr_primary_contact
FROM customer c1, customer c2
WHERE c1.cstmr_primary_contact = c2.cstmr_secondary_contact
AND c1.cstmr_id != c2.cstmr_id
ORDER BY c1.cstmr_primary_contact DESC;
```

8.

```
SELECT cstmr_id
FROM customer, address
WHERE customer.cstmr_addr_id = address.addr_id
AND address.addr_state = 'PA'
UNION
SELECT cstmr_id
FROM customer, contact c1, contact c2
WHERE (customer.cstmr_primary_contact = c1.cntct_id
AND customer.cstmr_secondary_contact IS NULL
AND c1.cntct_phone_number IS NULL)
OR (customer.cstmr_primary_contact = c1.cntct_id
AND customer.cstmr_secondary_contact = c2.cntct_id
AND (c1.cntct_phone_number IS NULL AND c2.cntct_phone_number IS NULL));
```

9.

```
SELECT customer.cstmr_id, address.addr_state,
SUM(line_item.litm_quantity * item.itm_unit_price) AS total_cost
FROM customer, contact, invoice, line_item, address, item
WHERE customer.cstmr_primary_contact = contact.cntct_id
AND contact.cntct_firstname LIKE ’Jess%’
AND customer.cstmr_addr_id = address.addr_id
AND invoice.inv_cstmr_id = customer.cstmr_id
AND line_item.litm_inv_id = invoice.inv_id
AND line_item.litm_itm_id = item.itm_id
GROUP BY customer.cstmr_id, address.addr_state;
```

10.

```
SELECT DISTINCT contact.cntct_firstname, contact.cntct_lastname
FROM customer, invoice, contact
WHERE invoice.inv_payment_status_cd = ’UNP’
AND customer.cstmr_primary_contact = contact.cntct_id
AND invoice.inv_cstmr_id = customer.cstmr_id
AND invoice.inv_date < ’2022-10-8’;
```

11.

```
SELECT cntct_id
FROM contact
EXCEPT
SELECT cstmr_primary_contact
FROM customer
EXCEPT
SELECT cstmr_secondary_contact
FROM customer;
```

12.

```
SELECT customer.cstmr_id, cus.addr_state, stock.stk_wh_id
FROM customer, address cus, address wh, warehouse, line_item, item, invoice, stock
WHERE customer.cstmr_addr_id = cus.addr_id
AND customer.cstmr_id = invoice.inv_cstmr_id
AND invoice.inv_id = line_item.litm_inv_id
AND line_item.litm_itm_id = item.itm_id
AND invoice.inv_payment_status_cd = ’UNP’
AND item.itm_id = stock.stk_itm_id
AND stock.stk_quantity > 0
AND stock.stk_wh_id = warehouse.wh_id
AND warehouse.wh_addr_id = wh.addr_id
AND cus.addr_state = wh.addr_state;
```

13.

```
SELECT customer.cstmr_name
FROM customer, contact
WHERE customer.cstmr_primary_contact = contact.cntct_id
AND contact.cntct_email_addr LIKE ’%@apple%’;
```

14.

```
SELECT customer.cstmr_id, SUM(litm_quantity * itm_unit_price) AS outstanding_one
FROM customer, line_item, item, invoice
WHERE customer.cstmr_id = invoice.inv_cstmr_id
AND invoice.inv_payment_status_cd IN (’UNP’,’DEF’)
AND invoice.inv_id = line_item.litm_inv_id
AND line_item.litm_itm_id = item.itm_id
GROUP BY customer.cstmr_id;
```

15.

```
SELECT SUM(litm_quantity * itm_unit_price) AS total_balance
FROM invoice, line_item, item
WHERE invoice.inv_id = line_item.litm_inv_id
AND line_item.litm_itm_id = item.itm_id
AND invoice.inv_payment_status_cd = ’UNP’
AND invoice.inv_date < ’2021-10-8’;
```

16.

```
SELECT invoice.inv_id, SUM(litm_quantity * itm_unit_price)/SUM(litm_quantity)
AS avg_cost
FROM invoice, line_item, item
WHERE invoice.inv_id = line_item.litm_inv_id
AND line_item.litm_itm_id = item.itm_id
GROUP BY invoice.inv_id
ORDER BY avg_cost DESC
LIMIT 5;
```

17.

```
SELECT SUM(litm_quantity * itm_unit_price) AS total
FROM item, line_item, customer, invoice
WHERE customer.cstmr_id = invoice.inv_cstmr_id
AND invoice.inv_id = line_item.litm_inv_id
AND line_item.litm_itm_id = item.itm_id
AND item.itm_name = ’Nvidia H100’;
```

18.

```
SELECT SUM(itm_unit_price * litm_quantity) AS total_UNP_balance
FROM customer, contact, invoice, line_item, item
WHERE customer.cstmr_id = invoice.inv_cstmr_id
AND (customer.cstmr_primary_contact = contact.cntct_id
OR customer.cstmr_secondary_contact = contact.cntct_id)
AND contact.cntct_firstname = ’Aliya’
AND contact.cntct_lastname = ’Khan’
AND invoice.inv_payment_status_cd = ’UNP’
AND invoice.inv_id = line_item.litm_inv_id
AND line_item.litm_itm_id = item.itm_id
GROUP BY customer.cstmr_id
HAVING total_UNP_balance >= 20000;
```

19.

```
SELECT itm_id
FROM item
EXCEPT
SELECT stk_itm_id
FROM stock
WHERE stk_quantity > 0;
```

20.

```
SELECT customer.cstmr_id, contact.cntct_firstname
FROM customer
LEFT JOIN contact ON customer.cstmr_secondary_contact = contact.cntct_id
WHERE customer.cstmr_id IN
 (SELECT inv_cstmr_id
 FROM invoice
 WHERE invoice.inv_date >= ’2022-10-9’
 AND invoice.inv_id IN
  (SELECT litm_inv_id
  FROM line_item
  WHERE litm_itm_id =
   (SELECT itm_id
   FROM item
   WHERE itm_name = ’V.F.D.’)
  GROUP BY litm_inv_id
  HAVING SUM(litm_quantity) > 3));
```

21.

```
SELECT litm_inv_id, litm_itm_id
FROM line_item
WHERE litm_quantity > 2 * (SELECT AVG(litm_quantity) FROM line_item);
```

22.

```
SELECT inv_id
FROM invoice
JOIN line_item ON invoice.inv_id = line_item.litm_inv_id
JOIN item ON line_item.litm_itm_id = item.itm_id
WHERE invoice.inv_payment_status_cd = ’UNP’
GROUP BY inv_id
HAVING SUM(line_item.litm_quantity * item.itm_unit_price) = (
 SELECT MAX(total_value)
 FROM
  (SELECT SUM(line_item.litm_quantity * item.itm_unit_price) AS total_value
  FROM invoice
  JOIN line_item ON invoice.inv_id = line_item.litm_inv_id
  JOIN item ON line_item.litm_itm_id = item.itm_id
  WHERE invoice.inv_payment_status_cd = ’UNP’
  GROUP BY invoice.inv_id)
  AS total_values);
```

23.

```
 SELECT itm_id, itm_vendor_sku
 FROM item
 WHERE itm_id IN (
  SELECT stk_itm_id
  FROM stock
  WHERE stk_quantity = 0
  GROUP BY stk_itm_id
  HAVING COUNT(stk_wh_id) = 2
 );
```

24.

```
 CREATE TABLE item (
    itm_id INTEGER PRIMARY KEY,
    itm_name VARCHAR(120) NOT NULL,
 itm_vendor_sku VARCHAR(30) NOT NULL,
    itm_unit_price DECIMAL(8, 2) CHECK (itm_unit_price > 0 AND itm_unit_price <= 100000.00),

    itm_vendor_id INT,
    FOREIGN KEY (itm_vendor_id) REFERENCES vendor(vendor_id)
);
```

25.

```
 CREATE TABLE vendor (
  vendor_id INTEGER PRIMARY KEY,
  vendor_name VARCHAR(120) NOT NULL,
  vendor_primary_contact INTEGER NOT NULL,
  vendor_address VARCHAR(255) NOT NULL,
  vendor_currency_code CHAR(3) NOT NULL CHECK (LENGTH(vendor_currency_code) = 3),
  FOREIGN KEY (vendor_primary_contact) REFERENCES contact(cntct_id)
 );

 ALTER TABLE item
 ADD COLUMN itm_vendor_id INTEGER,
 ADD CONSTRAINT fk_vendor
 FOREIGN KEY (itm_vendor_id)
 REFERENCES vendor(vendor_id)
 ON DELETE RESTRICT
 ON UPDATE CASCADE;
```