Dias Adeshov
Laboratory work 6

1. Write a SQL query using Joins:

a. Combine each row of dealer table with each row of client table

```
SELECT
  *
FROM
  client
  JOIN dealer ON client.dealer_id = dealer.id;
```

b. find all dealers along with client name, city, grade, sell number, date, and amount

```
SELECT
  d.id "Dealer id",
  d.name "Dealer name",
  d.location "Dealer location",
  d.charge,
  c.name "Client name",
  c.city "Client city",
  s.id "Sell number",
  s.date,
  s.amount
FROM
  dealer as d
  LEFT JOIN sell as s ON d.id = s.dealer_id
  LEFT JOIN client as c ON client_id = c.id;
```

c. find the dealer and client who belongs to same city

```
SELECT
  d.id "Dealer id",
  d.name "Dealer name",
  d.location "Dealer location",
  d.charge,
  c.id "Client_id",
  c.name "Client name",
  c.city "Client city",
  priority
FROM
```

```
  client c
  INNER JOIN dealer d ON c.dealer_id = d.id
WHERE
  c.city = d.location
```

d. find sell id, amount, client name, city those sells where sell amount exists between 100 and 500

```
SELECT
  s.id "Sell id",
  s.amount,
  c.name,
  c.city
FROM
  sell s
  INNER JOIN client c ON s.client_id = c.id
WHERE
  amount BETWEEN 100 AND 500
```

e. find dealers who works either for one or more client or not yet join under any of the clients

```
SELECT
  d.id "Dealer id",
  d.name,
  count(c.dealer_id) "number of clients"
FROM
  dealer d
  FULL JOIN client c ON c.dealer_id = d.id
GROUP BY
  d.id,
  d.name
HAVING
  count(c.dealer_id) > 1
  OR count(c.dealer_id) = 0;
```

f. find the dealers and the clients he service, return client name, city, dealer name, commission.

```
SELECT
  c.name "client_name",
  c.city,
  d.name "dealer_name",
  ROUND((s.amount * d.charge)::DECIMAL, 2) "commission"
FROM
  client c
  INNER JOIN sell s ON c.id = s.client_id
  INNER JOIN dealer d ON c.dealer_id = d.id
```

g. find client name, client city, dealer, commission those dealers who received a commission from the sell more than 12%

```
SELECT
  c.name "client_name",
  c.city,
  d.name "dealer_name",
  ROUND((s.amount * d.charge)::DECIMAL, 2) "commission"
FROM
  client c
  INNER JOIN sell s ON c.id = s.client_id
  INNER JOIN dealer d ON c.dealer_id = d.id
WHERE d.charge > 0.12
```

h. make a report with client name, city, sell id, sell date, sell amount, dealer name and commission to find that either any of the existing clients haven’t made a purchase(sell) or made one or more purchases(sell) by their dealer or by own.

```
SELECT
  c.name "client name",
  c.city, s.amount,
  d.name "dealer name",
  ROUND((amount * d.charge)::DECIMAL, 2) "commission"
FROM sell s
INNER JOIN client c ON s.client_id = c.id
INNER JOIN dealer d ON s.dealer_id = d.id
WHERE s.dealer_id <> d.id
```

i. find dealers who either work for one or more clients. The client may have made, either one or more purchases, or purchase amount above 2000 and must have a grade, or he may not have made any purchase to the associated dealer. Print client name, client grade, dealer name, sell id, sell amount

```
```

2. Create following views:

a. count the number of unique clients, compute average and total purchase amount of client orders by each date.

```
SELECT
  date,
  COUNT(DISTINCT(client_id)) "quantity unique clients",
  ROUND(SUM(amount)::DECIMAL, 2) "total amount",
  ROUND(AVG(amount)::DECIMAL, 2) "avg amount"
FROM
  sell
GROUP BY
  date
```

b. find top 5 dates with the greatest total sell amount

```
SELECT
  date,
  SUM(amount) "total"
FROM
  sell
GROUP BY
  date
ORDER BY
  sum(amount) desc
LIMIT 5;
```

c. count the number of sales, compute average and total amount of all sales of each dealer d. compute how much all dealers earned from charge(total sell amount * charge) in each location

```
SELECT
  d.location,
  --SUM(s.amount * d.charge)
```

```
  CAST(SUM(s.amount * d.charge) as DECIMAL(10, 2)) "total
earned"
FROM
  sell s
  INNER JOIN dealer d ON s.dealer_id = d.id
GROUP BY
  d.location;
```

e. compute number of sales, average and total amount of all sales dealers made in each location

```
SELECT
  d.location,
  COUNT(d.id) "sales quantity",
  ROUND(SUM(s.amount)::DECIMAL, 2) "total amount",
  ROUND(AVG(s.amount)::DECIMAL, 2) "avg"
FROM
  sell s
  INNER JOIN dealer d ON s.dealer_id = d.id
GROUP BY
  d.location;
```

f. compute number of sales, average and total amount of expenses in each city clients made.

```
SELECT
  COUNT(s.id) "number of sales",
  ROUND(SUM(s.amount)::DECIMAL, 2) "total amount",
  ROUND(AVG(s.amount)::DECIMAL, 2) "average"
FROM
  sell s
  INNER JOIN client c ON s.client_id = c.id
GROUP BY
  c.city
```

g. find cities where total expenses more than total amount of sales in locations

```
SELECT
  cities.city,
  ROUND(expenses.expenses::DECIMAL, 2) "expenses",
  ROUND(income.income::DECIMAL, 2) "income"
FROM
  (
    SELECT
      location as city
    FROM
      dealer
    UNION ALL
    SELECT
      city
    FROM
      client
  ) as cities -- get all cities
  LEFT JOIN (
    SELECT
      SUM(amount) as expenses,
      city
    FROM
      sell s
      INNER JOIN client c ON s.client_id = c.id
    GROUP BY
      city
  ) as expenses ON cities.city = expenses.city -- cities
expenses
  LEFT JOIN (
    SELECT
      SUM(amount) as income,
      location
    FROM
      sell s
      INNER JOIN dealer d ON s.dealer_id = d.id
    GROUP BY
      location
  ) as income ON cities.city = income.location -- cities
income
GROUP BY
  cities.city,
  expenses.expenses,
  income.income
HAVING expenses.expenses > income.income OR income.income IS
NULL
```