```sql
INSERT INTO sales_reps(id, name, region_id)
values(321991, 'Alina Shein', 5), (321992, 'Alberto Quin', 8);

INSERT INTO region(id, name)
values(5, 'International'), (6, 'South'), (7, 'North');

-- Question #1
SELECT COUNT(DISTINCT id) from accounts
SELECT COUNT(DISTINCT name) from accounts -- To ensure the answer is compatible
-- Answer: 351 accounts

-- Question #2
SELECT name from region -- This is to show how many areas in total
-- Answer: 7 areas, including Northeast, Midwest, Southeast, West, International, South, and North

-- This is to see if sales actually happened at all mentioned areas or was there any area that no sale occured at
SELECT region.name as region_name, count(orders.id) as total_orders from region
full outer join sales_reps
on region.id = sales_reps.region_id
full outer join accounts
on sales_reps.id = accounts.sales_rep_id
full OUTer join orders
on accounts.id = orders.account_id
group by region.name
order by total_orders DESC
--Answer: Although they sell at 7 areas, there was no sales occured in South, North, and International regions

-- Question #3a
SELECT count(*) from orders
where standard_qty = 0 and gloss_qty = 0 and poster_qty = 0
-- They might sell some other types of papers that aren't listed here

select sum(standard_qty) as standard_paper, sum(gloss_qty) as gloss_paper, SUM(poster_qty) as poster_paper from orders
-- There are 3 types of papers that are shown on the report
-- standard_paper = 1,938,346 units, gloss_paper = 1,013,773 units ,poster_paper = 723,646 units
SELECT sum(total) from orders
-- Total of 3,675,765 units

SELECT ROUND((sum(standard_qty)*100.0/sum(total)),2) as percentage_standard_qty,
        ROUND((sum(gloss_qty)*100.0/sum(total)),2) as percentage_gloss_qty,
     ROUND((SUM(poster_qty)*100.0/sum(total)),2) as percentage_poster_qty from orders
-- Answer: -- standard_paper = 52.73% units, gloss_paper = 27.58%, poster_paper = 19.69%

-- Question #3b
select sum(standard_amt_usd) as standard_usd, sum(gloss_amt_usd) as gloss_usd, SUM(poster_amt_usd) as poster_usd from orders
-- standard_paper = $9,672,346.54 , gloss_paper = $7,593,159.77 ,poster_paper = $5,876,005.52
SELECT sum(total_amt_usd) from orders
-- Total revenue of $23,141,511.83

SELECT ROUND((sum(standard_amt_usd)*100.0/sum(total_amt_usd)),2) as percentage_standard_usd,
        ROUND((sum(gloss_amt_usd)*100.0/sum(total_amt_usd)),2) as percentage_gloss_usd,
     ROUND((SUM(poster_amt_usd)*100.0/sum(total_amt_usd)),2) as percentage_poster_usd from orders
-- Answer: -- standard_paper = 41.80% units, gloss_paper = 32.81%, poster_paper = 25.39%

-- Question 4a
SELECT Extract (year from occurred_at) as year_occured,
     sum(total_amt_usd) as annual_total_revenue,
     sum(total_amt_usd)-(lag(sum(total_amt_usd),1) over(order by Extract (year from occurred_at))) as revenue_growth from orders
where Extract (year from occurred_at) in ('2014', '2015', '2016')
group by year_occured
order by year_occured
-- We can see that from 2014 to 2015, the total revenue increased by $1,682,898.40
-- and from 2015 to 2016, the total revenue increased by $7,112,912.98

-- Better present with percentage growth
WITH yearly_revenue AS
        (SELECT EXTRACT(YEAR FROM occurred_at) AS year_occurred,
      SUM(total_amt_usd) AS annual_total_revenue FROM orders
      WHERE EXTRACT(YEAR FROM occurred_at) IN ('2014', '2015', '2016')
      GROUP BY year_occurred)

SELECT
    year_occurred,
    annual_total_revenue,
    annual_total_revenue - LAG(annual_total_revenue, 1) OVER (ORDER BY year_occurred) AS revenue_growth_usd,
    ROUND(((annual_total_revenue - LAG(annual_total_revenue, 1) OVER (ORDER BY year_occurred)) /
                LAG(annual_total_revenue, 1) OVER (ORDER BY year_occurred)) * 100, 4) AS revenue_growth_percentage
        FROM yearly_revenue
ORDER BY year_occurred
-- The revenue in each year is $4,069,106.54 in 2014, $5,752004.94 in 2015, and $12,864,917.92 in 2017
-- We can see that from 2014 to 2015, the total revenue increased by $1,682,898.40 or 41.36%
-- and from 2015 to 2016, the total revenue increased by $7,112,912.98 or 123.66%

-- Question 4b
WITH previous_year AS
        (SELECT EXTRACT(YEAR FROM occurred_at) AS year_recorded,
```

```sql
        SUM(total) AS annual_total_units,
        LAG(SUM(total),1) OVER (ORDER BY EXTRACT(YEAR FROM occurred_at)) as previous_year_units from orders
        WHERE EXTRACT(YEAR FROM occurred_at) IN ('2014', '2015', '2016')
        GROUP BY year_recorded)

SELECT
    year_recorded,
    annual_total_units,
    annual_total_units - previous_year_units AS quantity_growth,
    ROUND(((annual_total_units - previous_year_units)*100.0/
        LAG(annual_total_units, 1) OVER (ORDER BY year_recorded)), 4) AS quantity_growth_percentage
        FROM previous_year
ORDER BY year_recorded
-- The total units sold are 650,896 units in 2014, 912,972 units in 2015, and 2,041,600 units in 2016
-- We can see that from 2014 to 2015, the total units sold increased by 262,076 units or 40.2639%
-- and from 2015 to 2016, the total units sold increased by 1,128,628 units or 123.6213%


-- Question 5
SELECT COUNT(name) from sales_reps -- See how many sale reps' names in total, which is 52 names

SELECT region.name as region_name, COUNT(sales_reps.name) as sales_reps_counts from sales_reps
full outer join region
on sales_reps.region_id = region.id
group by region_name
order by region_name
-- Answer: Compatible with a total of 52 sale reps' names above.
-- Answer: There 1 sale rep in International,
        -- 9 sale reps in the Midwest,
        -- 0 sale rep in the North,
        -- 21 sale reps in the Northeast,
        -- 0 in the South,
        -- 10 in the Southeast,
        -- and 1 in the West.


-- Question #6a
select r.name as Region,
        count(distinct o.id) as total_orders,
      count(distinct(a.sales_rep_id)) as number_of_reps,
      count(distinct a.id) as number_of_accounts,
      round(sum(total_amt_usd), 2) as total_revenue,
        round(sum(total_amt_usd)/count(distinct o.id), 2) as average_revenue
from orders as o
full join accounts as a
      on o.account_id = a.id
join sales_reps as sr
      on a.sales_rep_id = sr.id
full join region as r
      on sr.region_id = r.id
where extract(year from occurred_at) = 2016
group by r.name
order by number_of_reps desc,
            total_revenue desc,
            average_revenue desc,
            total_orders desc,
            number_of_accounts desc;


-- Question #6b
select r.name as region,
        count(distinct o.id)/count(distinct sr.id) as avg_orders_per_rep,
      count(distinct a.id)/count(distinct a.sales_rep_id) as avg_accounts_per_rep,
        round(sum(total_amt_usd)/count(distinct sr.id),2) as avg_revenue_per_rep
from region as r
join sales_reps as sr
      on r.id = sr.region_id
join accounts as a
      on sr.id = a.sales_rep_id
join orders as o
      on a.id = o.account_id
where extract(year from occurred_at) = 2016
group by r.name
order by avg_revenue_per_rep desc,
            avg_accounts_per_rep desc,
            avg_orders_per_rep desc;

-- Question #6c
-- Answer is included in the presentation slides

-- Question #7
-- without space between% and Group >> This result include Citygroup and ManpowerGroup
with cte as
        (select a.name,
        case
        when name ilike '%Group' then 'Group'
        else 'Not Group'
        end as group_label,
        sum(o.total_amt_usd) as revenue
        from accounts a
```

```sql
        join orders o
        on a.id = o.account_id
        group by a.name, group_label)

-- With space between % and Group, the result returned only those companies that have a word "Group" standalone at the end
-- We recommend using this result as the group of business refer to the standalone Group at the end

select group_label as account_label, round(avg(revenue),2) avg_revenue, count(group_label)
from cte
group by group_label
order by avg_revenue

with cte as
        (select a.name,
        case
        when name ilike '% Group' then 'Group'
        else 'Not Group'
        end as group_label,
        sum(o.total_amt_usd) as revenue
        from accounts a
        join orders o
        on a.id = o.account_id
        group by a.name, group_label)

select group_label as account_label, round(avg(revenue),2) avg_revenue, count(group_label)
from cte
group by group_label
order by avg_revenue

-- Question #8
select  region, channel as least_used_channel
from
        (with cte as
                (select r.name region, channel
                from region r
                join sales_reps sr
                on r.id = sr.region_id
                join accounts a
                on a.sales_rep_id = sr.id
                join web_events w
                on a.id = w.account_id)

        select *,row_number()over(partition by region order by count(*) asc) as channel_rank
        from cte
        group by region, channel)
where channel_rank=1
order by region

-- Find the most used channel to identifyy the strategy for the new regions
select  region, channel as second_most_used_channel
from
        (with cte as
                (select r.name region, channel
                from region r
                join sales_reps sr
                on r.id = sr.region_id
                join accounts a
                on a.sales_rep_id = sr.id
                join web_events w
                on a.id = w.account_id)

        select *,row_number()over(partition by region order by count(*) DESC) as channel_rank
        from cte
        group by region, channel)
where channel_rank=2
order by region

SELECT *,
    (CASE WHEN a.name ILIKE '% group' THEN 'group' ELSE 'not group' END) AS group_label
FROM accounts a
WHERE (CASE WHEN a.name ILIKE '%group' THEN 'group' ELSE 'not group' END) IN ('group');

With accounts_g as (
 select *, (case when a.name ilike '% group' then 'group' else 'not group' end) as group_label from accounts a)

select group_label, sum(total_amt_usd)/count(distinct(o.account_id)) average_revenue
from accounts_g ag join orders o on ag.id = o.account_id
group by group_label
```