```
select * from sales;
select * from customer;
select * from region;
select * from salesmen;
select * from office;
select * from product;
select * from salesmen_final;
select * from total_sales;
select * from Salesby_Salesmen;

-- Query 1
-- Calculate Total Sales per SalesID

Drop table if exists Total_Sales;
Create table Total_Sales as
select *,ROUND(Lbs*`Unit Price`,2) as Total_Price
from sales;

-- Query 2
-- Allocation of Region to each Salesperson

Drop table if exists Salesmen_Final;
create table Salesmen_Final as
select x.*,`Customer Region` from
(select *,row_number() over (partition by Office order by
`SalesPerson_ID`) Order_1 from
salesmen)x left join
(select *,row_number() over (partition by RegionHQ_ID order by `Customer
Region`) Order_2
from region)y
on x.Office=y.RegionHQ_ID
and Order_1=Order_2
order by office,salesperson_ID;

-- Questions:

-- Query 3
-- Sum all revenue, compare year by year

select Year_of_Sales,concat('$ ',ifnull(round(sum(total_price),2),0))
Total_Sales
from
(
select *,right(Date,4) Year_of_Sales from Salesby_Salesmen
)x
group by Year_of_Sales;

-- Query 4
-- Salesperson performance

Drop table if exists Salesby_Salesmen;
create table Salesby_Salesmen as
select s.*,c.*,SalesPerson_ID,sf.Name as SalesPerson_Name,
sf.Office,sf.`Salary (USD)` from total_sales s
```

```
left join customer c
on s.`Cust Vend Num`=c.`Customer Vendor Number`
left join Salesmen_Final sf
on c.Region=sf.`Customer Region`;

-- Query 5
-- The performance of sales by each person

select
ifnull(SalesPerson_ID,'na') SalesPerson_ID,
ifnull(Salesperson_Name,'na') Salesperson_Name,
concat('$ ',round(sum(total_price),2)) Total_Sales,
concat('$ ',round(min(total_price),2)) Min_Sales,
concat('$ ',round(max(total_price),2)) Max_Sales,
concat('$ ',round(Avg(total_price),2)) Avg_Sales
from (select * from Salesby_Salesmen)y
group by SalesPerson_ID,Salesperson_Name
order by round(sum(total_price),2) desc;

-- Query 6
-- The performance of sales by each person during a determined period
-- Example Feb 2023

select
ifnull(SalesPerson_ID,'na') SalesPerson_ID,
ifnull(Salesperson_Name,'na') Salesperson_Name,
concat('$ ',round(sum(total_price),2)) Total_Sales,
period_y,period_m
from (select *,right(Date,4) Period_y,left(Date,1) Period_m from
Salesby_Salesmen)y
group by SalesPerson_ID,Salesperson_Name,Period_y,Period_m
having Period_y=2023 and Period_m=2
order by period_y,period_m asc, round(sum(total_price),2) desc;

-- Query 7
-- Comparison between periods to identify the biggest drops/gains in
sales.
-- Example: For Bob

select
ifnull(SalesPerson_ID,'na') SalesPerson_ID,
ifnull(Salesperson_Name,'na') Salesperson_Name,
concat('$ ',round(sum(total_price),2)) Total_Sales,
period_y,period_m
from (select *,right(Date,4) Period_y,left(Date,1) Period_m from
Salesby_Salesmen)y
group by SalesPerson_ID,Salesperson_Name,Period_y,Period_m
having SalesPerson_Name like 'Bob'
order by Period_y,Period_m;

-- Query 8
-- Geographical performance
```

```
-- The sales distribution in revenue, number of customers and different
products will be retrieved by region/state to identify what locations the
executive team has opportunities.
-- Min, max, average (group by region & state)
-- How many products are sold in a region: Select count distinct product
-- How many customers are in a region: Select count distinct customer

select
ifnull(`Customer Region`,'na') `Customer Region`,
ifnull(`Region Name`,'na') `Region Name`,
ifnull(max(o.RegionalHQ_Name),'na') Regional_HQ,
ifnull(`Customer State`,'na') `Customer State`,
count(distinct `Customer Vendor Number`) Customer_count,
count(distinct `Prod Code`) Product_count,
concat('$ ',round(sum(total_price),2)) Total_Sales,
concat('$ ',round(min(total_price),2)) Min_Sales,
concat('$ ',round(max(total_price),2)) Max_Sales,
concat('$ ',round(Avg(total_price),2)) Avg_Sales
from total_sales s
left join customer c
on s.`Cust Vend Num`=c.`Customer Vendor Number`
left join region r
on c.region=r.`Customer Region`
left join office o
on o.RegionHQ_ID=r.RegionHQ_ID
group by `Customer Region`,`Region Name`,`Customer State`
order by `Customer State`,`Region Name` asc,sum(total_price) desc;

-- Query 9
-- Customer stats
-- Ranking the bottom and top 10 customers in sales to understand the
benchmark as well as making action plans to improve the sales in the
biggest drops.
-- Top consumers: order by, descending, limit 10 (Group by: CustVendNum)
-- Customer retention: which customers order every year

select  `Customer Vendor Number`, `Customer Name`,`Customer
State`,`Business Type`,`Region`,
concat('$ ',ifnull(round(sum(total_price),2),0)) Total_Sales,
concat('$ ',ifnull(round(min(total_price),2),0)) Min_Sales,
concat('$ ',ifnull(round(max(total_price),2),0)) Max_Sales,
concat('$ ',ifnull(round(Avg(total_price),2),0)) Avg_Sales
from customer c
left join total_sales s
on s.`Cust Vend Num`=c.`Customer Vendor Number`
group by `Customer Vendor Number`,`Customer Name`,`Customer
State`,`Business Type`,`Region`
having round(sum(total_price),2) >0
order by round(sum(total_price),2) asc
limit 10;

-- Query 10
```

```
-- Product performance: Ranking the bottom and top 10 products in sales to
understand the benchmark as well as make action plans to improve the sales
in the biggest drops.
-- Top selling product: Order by, descending, limit 10, (Group by:
ProdCode)


select  `Product Code`, `Product Data.GCP3`,`Product Data.GCP10`,
concat('$ ',ifnull(round(sum(total_price),2),0)) Total_Sales,
concat('$ ',ifnull(round(min(total_price),2),0)) Min_Sales,
concat('$ ',ifnull(round(max(total_price),2),0)) Max_Sales,
concat('$ ',ifnull(round(Avg(total_price),2),0)) Avg_Sales
from product c
left join total_sales s
on c.`Product Code`=s.`prod code`
group by `Product Code`, `Product Data.GCP3`,`Product Data.GCP10`
having round(sum(total_price),2) >0
order by round(sum(total_price),2) asc
limit 10;
```