

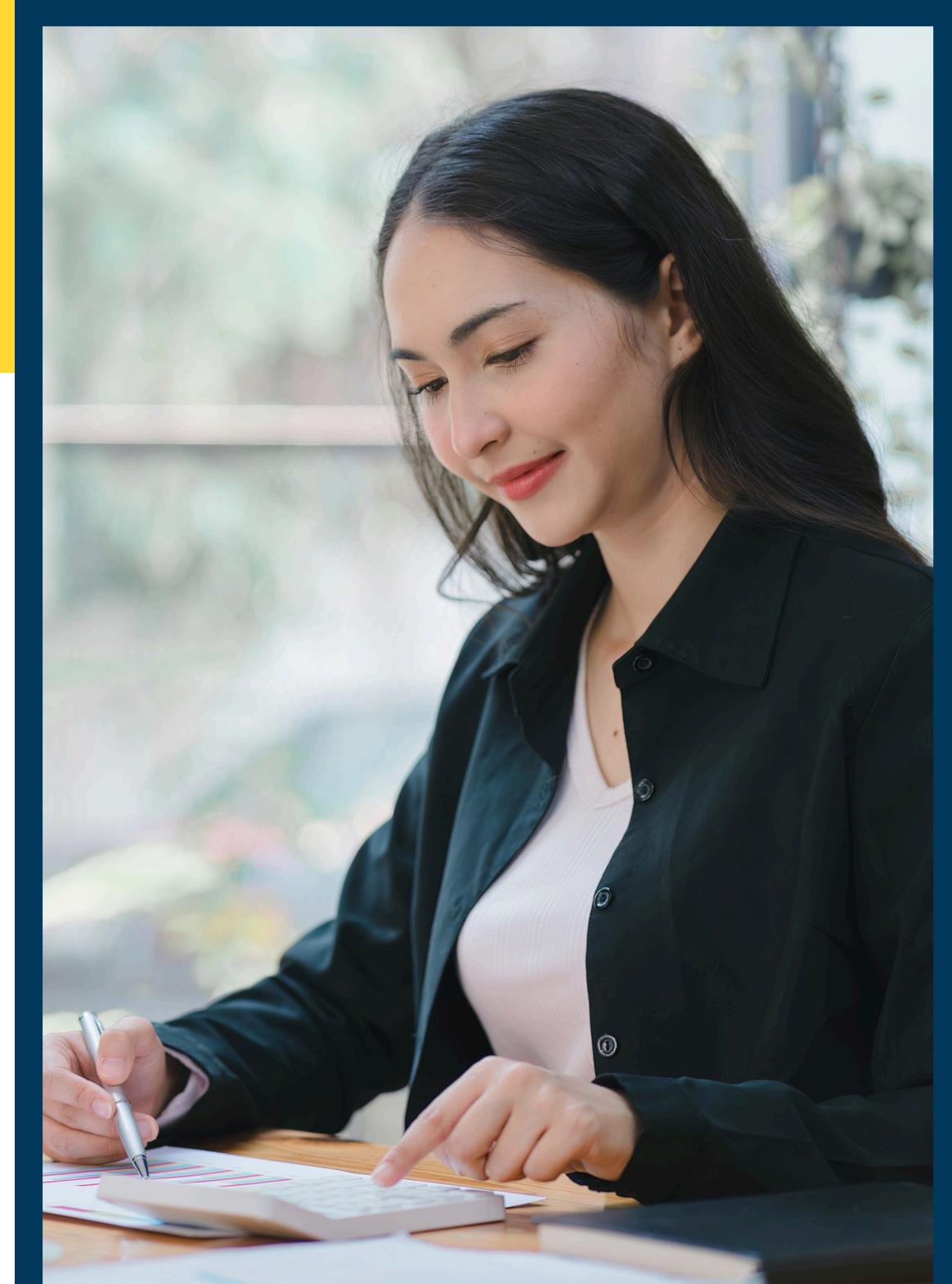


AMAZON.COM, INC

# SALES REPORT - SQL

---

A detailed and complete sales report is very helpful to see which parts need to be changed or improved.



# WHAT IS A SALES REPORT

A sales report, also known as a sales report, is a summary prepared periodically (usually monthly, quarterly or annually) which contains information about various aspects of a company's sales. This report includes data such as the number of units sold, revenue generated, production costs, net profits, sales trends, and much more.



# DATA WRANGLING

```
#CREATE DATABASE  
create database AmazonSales;
```

# DATA WRANGLING

```
#CREATE TABLE
create table Amazon_Sales(
    invoice_id varchar(30) not null primary key,
    branch varchar(10) not null,
    city varchar(30) not null,
    customer_type varchar(30) not null,
    Gender varchar(15) not null,
    Product_line varchar(100) not null,
    unit_price decimal(10,2) not null,
    quantity int not null,
    tax_pct float not null,
    total decimal(12,4) not null,
    date date not null,
    time time not null,
    payment varchar(30) not null,
    cogs decimal(10,2) not null,
    gross_margin_pct float,
    gross_income decimal(12,4),
    rating decimal(2,1));
```

# FEATURE ENGINEERING

```
#TIME_OF_DAY  
  
alter table amazon_sales  
add column time_of_day varchar(30);  
  
update amazon_sales  
set time_of_day =  
(case when time between '00:00:00' and '12:00:00' then 'Morning'  
when time between '12:00:01' and '16:00:00' then 'Afternoon'  
else 'Evening'  
end);
```

# FEATURE ENGINEERING

```
#DAY_NAME  
  
alter table amazon_sales  
add column day_name varchar(30)
```

```
update amazon_sales  
set day_name =  
dayname(date);
```

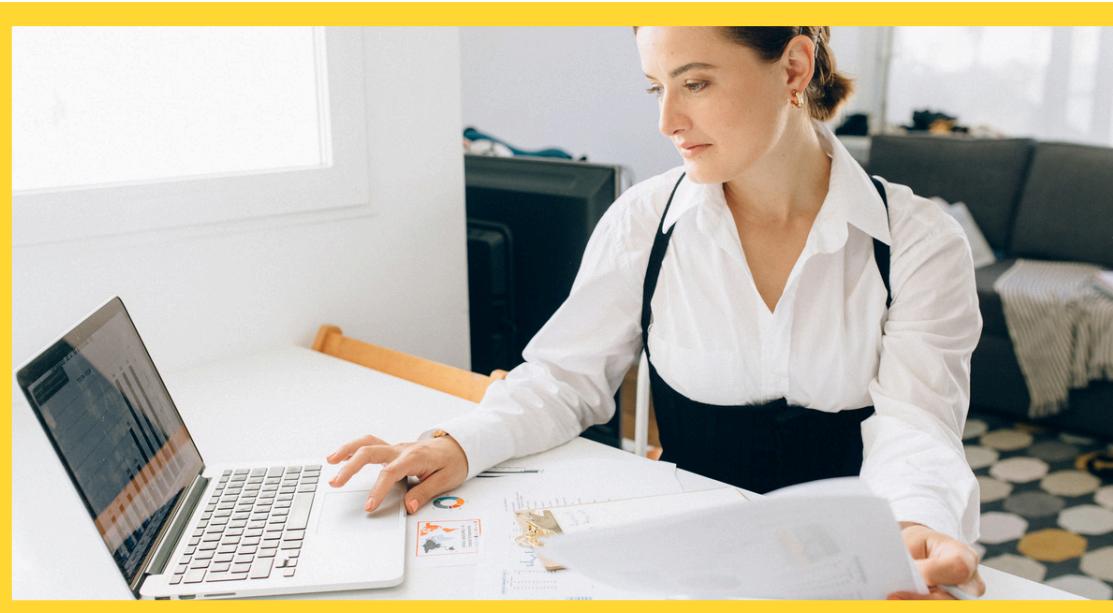
# FEATURE ENGINEERING

#MONTHNAME

```
alter table amazon_sales  
add column month_name varchar(30)
```

```
update amazon_sales  
set month_name =  
monthname(date);
```

# PRODUCT ANALYSIS



#1. What is the count of distinct cities in the dataset?

```
select count(distinct city) as unique_cities  
from amazon_sales;
```

#2. For each branch, what is the corresponding city?

```
select distinct branch, city  
from amazon_sales;
```

#3. What is the count of distinct product lines in the dataset?

```
select count(distinct product_line) as unique_product_line  
from amazon_sales;
```

#4. Which payment method occurs most frequently?

```
select payment, count(payment) as payment_frequency  
from amazon_sales  
group by payment  
order by payment_frequency desc  
limit 1;
```

#5. Which product line has the highest sales?

```
select product_line, sum(total) as sales  
from amazon_sales  
group by product_line  
order by sum(total) desc  
limit 1;
```

#6. How much revenue is generated each month?

```
select month_name, sum(total) as total_revenue  
from amazon_sales  
group by month_name  
order by total_revenue desc;
```

#7. In which month did the cost of goods sold reach its peak?

```
select month_name, sum(cogs) as total_cogs  
from amazon_sales  
group by month_name  
order by total_cogs desc  
limit 1;
```

#8. Which product line generated the highest revenue?

```
select product_line, sum(total) as revenue  
from amazon_sales  
group by product_line  
order by revenue desc  
limit 1;
```

#9. In which city was the highest revenue recorded?

```
select city, sum(total) as revenue  
from amazon_sales  
group by city  
order by revenue desc  
limit 1;
```

#10. Which product line incurred the highest Value Added Tax?

```
select product_line, avg(tax_pct) as total_tax  
from amazon_sales  
group by product_line  
order by total_tax desc  
limit 1;
```

#11. For each product line, add a column indicating "Good"  
-- if its sales are above average, otherwise "Bad."

```
select product_line, sum(total) as total_sales,
```

```
case when sum(total)>
```

```
(select avg(total_sales)
```

```
from
```

```
(select product_line,
```

```
sum(total) as total_sales
```

```
from amazon_sales
```

```
group by product_line) as sales )
```

```
then 'good'
```

```
else 'bad'
```

```
end as performance
```

```
from amazon_sales
```

```
group by product line;
```

Identify the branch that exceeded the average number of products sold.

```
select branch, sum(quantity) as total_quantity  
from amazon_sales  
group by branch  
having sum(quantity) > (select avg(total_quantity)  
from  
(select branch, sum(quantity) as total_quantity  
from amazon_sales  
group by branch) as quantity_sold)
```

#13. Which product line is most frequently associated with each gender.

```
with gender_product_rank as
(
  select gender, product_line, count(*) as frequency,
  dense_rank() over (partition by gender order by count(*) desc) as ranking
  from amazon_sales
  group by gender, product_line
)
select gender, product_line, frequency
from gender_product_rank
where ranking=1;
```

#14. Calculate the average rating for each product line.

```
select product_line,
avg(rating) as average_rating
from amazon_sales
group by product_line
order by average_rating desc;
```

# SALES ANALYSIS



#15. Count the sales occurrences for each time of day on every weekday.

```
select day_name, time_of_day,  
       count(invoice_id) as sales_occurrences  
  from amazon_sales  
 where day_name in('Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday')  
 group by day_name, time_of_day  
 order by day_name, time_of_day;
```

#16. Identify the customer type contributing the highest revenue.

```
select customer_type, sum(total) as revenue  
  from amazon_sales  
 group by customer_type  
 order by revenue desc  
 limit 1;
```

#17. Determine the city with the highest VAT percentage.

```
select city, sum(tax_pct) as VAT_pct  
from amazon_sales  
group by city  
order by VAT_pct desc;
```

#18. Identify the customer type with the highest VAT payments.

```
select customer_type, sum(cogs * (tax_pct/100)) as VAT_payment  
from amazon_sales  
group by customer_type  
order by VAT_payment desc  
limit 1;
```

Identify the day of the week with the highest average ratings.

```
select day_name, avg(rating) as average_rating  
from amazon_sales  
group by day_name  
order by average_rating desc  
limit 1;
```

What is the count of distinct payment methods in the dataset?

```
select count(distinct payment) as distinct_payment_method  
from amazon_sales;
```

Determine the day of the week  
with the highest average ratings for each branch.

with ranks as

```
(  
    select branch, day_name, avg(rating) as average_rating,  
    dense_rank() over(partition by branch order by avg(rating) desc) as ranking  
    from amazon_sales  
    group by branch, day_name  
)  
  
select branch, day_name  
from ranks  
where ranking = 1;
```

# CUSTOMER ANALYSIS



Which customer type occurs most frequently?

```
select customer_type, count(*) as frequency  
from amazon_sales  
group by customer_type  
order by frequency desc  
limit 1;
```

Identify the customer type with the highest purchase frequency.

```
select customer_type,  
count(invoice_id) as purchase_frequency  
from amazon_sales  
group by customer_type  
order by purchase_frequency desc  
limit 1;
```

Determine the predominant gender among customers.

```
select gender, count(invoice_id) as total_count  
from amazon_sales  
group by gender  
order by total_count desc  
limit 1;
```

Examine the distribution of genders within each branch.

```
select branch,  
count(case when gender= 'male' then 1 end) as 'Male_count',  
count(case when gender = 'female' then 1 end) as 'Female_count'  
from amazon_sales  
group by branch  
order by branch;
```

Identify the time of day when customers provide the most ratings.

```
select time_of_day, count(rating) as ratings  
from amazon_sales  
group by time_of_day  
order by ratings;
```

Determine the time of day with the highest customer ratings for each branch.

```
with ratings as  
(  
select branch, time_of_day,  
dense_rank() over (partition by branch order by sum(rating) desc) as ranking  
from amazon_Sales  
group by branch, time_of_day  
)  
select branch, time_of_day  
from ratings  
where ranking = 1;
```

# AMAZON.COM, INC

## PRESENTED BY:

---

### PARUL SAHU



# THANK YOU

---