Blinkit Data – E-commerce Portfolio

```sql
SELECT * FROM Blinkit_data
```

```sql
Select COUNT(*) from blinkit_data
```

```sql
UPDATE Blinkit_data
SET Item_Fat_Content =
CASE
WHEN Item_Fat_Content IN ('LF', 'low fat') THEN 'Low Fat'
WHEN Item_Fat_Content ='reg'THEN 'Regular'
ELSE Item_Fat_Content
END
```

```sql
select DISTINCT (Item_Fat_Content) From Blinkit_data
```

```sql
Select SUM(Sales) as Total_Sales from Blinkit_data
```

```sql
Select CAST(SUM(Sales)/1000000 as Decimal(10,2))
as Total_Sales from Blinkit_data
```

```sql
Select CONCAT(CAST(SUM(Sales)/1000000 as Decimal(10,2)), 'M')
as Total_Sales from Blinkit_data
```

```sql
Select AVG(Sales)as Average_Sales from Blinkit_data
```

```sql
Select CAST(AVG(Sales) as Decimal(10,2)) as Average_Sales from Blinkit_data

Select CAST(AVG(Sales) as Decimal(10,0)) as Average_Sales from Blinkit_data
```

```sql
Select DISTINCT(Item_Type) as Number_of_Items from Blinkit_data
```

```sql
Select COUNT(DISTINCT(Item_Type)) as Number_of_Items from Blinkit_data
```

```sql
Select COUNT(*) as Number_of_Items from Blinkit_data
where Outlet_Establishment_Year = 2022
```

```sql
Select CAST(AVG(Rating) as decimal(10,1)) as Avg_Rating from Blinkit_data
```

```sql
SELECT Item_Fat_Content,
            CAST(SUM(Sales)/100000 as Decimal(10,2)) as Total_Sales,
            CAST(AVG(Sales) as Decimal(10,1)) as Average_Sales,
            COUNT(*) as No_Of_Items,
            CAST(AVG(Rating) as decimal(10,2)) as Avg_Rating
from Blinkit_data
GROUP BY Item_Fat_Content
ORDER BY Total_Sales DESC
```

```sql
SELECT Item_Fat_Content,
            CAST(SUM(Sales)/100000 as Decimal(10,2)) as Total_Sales,
            CAST(AVG(Sales) as Decimal(10,1)) as Average_Sales,
            COUNT(*) as No_Of_Items,
            CAST(AVG(Rating) as decimal(10,2)) as Avg_Rating
from Blinkit_data
WHERE Outlet_Establishment_Year = 2022
GROUP BY Item_Fat_Content
ORDER BY Total_Sales DESC
```

```sql
SELECT Item_Type,
            CAST(SUM(Sales)/100000 as Decimal(10,2)) as Total_Sales,
            CAST(AVG(Sales) as Decimal(10,1)) as Average_Sales,
            COUNT(*) as No_Of_Items,
            CAST(AVG(Rating) as decimal(10,2)) as Avg_Rating
from Blinkit_data
WHERE Outlet_Establishment_Year = 2022
GROUP BY Item_Type
ORDER BY Total_Sales DESC
```

Bank Loan Data

1. Total Loan Applications

```
SELECT COUNT(id) FROM Bank_loan_data
```

SELECT COUNT(id) AS TOTAL_LOAN_APPLICATIONS FROM Bank_loan_data

	TOTAL_LOAN_APPLICATIONS
1	38576

(MONTH-TO-DATE) MTD_Total loan Applications

```
SELECT CONCAT(CAST(COUNT(id)/1000.0 as Decimal(10,2)),'k')
AS TOTAL_LOAN_APPLICATIONS
FROM Bank_loan_data
WHERE MONTH(issue_date) = 12 AND
YEAR(issue_date) = 2021
```

	TOTAL_LOAN_APPLICATIONS
1	4.31k

(PREVIOUS MONTH-TO-DATE) PMTD LOAN APPLICATIONS

```
SELECT CONCAT(CAST(COUNT(id)/1000.0 as Decimal(10,2)),'k')
AS TOTAL_LOAN_APPLICATIONS
FROM Bank_loan_data
WHERE MONTH(issue_date) = 11 AND
YEAR(issue_date) = 2021
```

	TOTAL_LOAN_APPLICATIONS
1	4.04k

NOTE: DIFFERENCE OF MTD AND PMTD
FORMULAE: MTD-PMTD/PMTD

2. Total Funded Amount

```
SELECT SUM(loan_amount)
AS TOTAL_FUNDED_AMOUNT
FROM Bank_loan_data
```

	TOTAL_FUNDED_AMOUNT
1	435757075

```
SELECT CONCAT(SUM(loan_amount)/1000000,'M')
AS TOTAL_FUNDED_AMOUNT
FROM Bank_loan_data
```

	TOTAL_FUNDED_AMOUNT
1	435M

MTD total Funded Amount

```
SELECT CONCAT(SUM(loan_amount)/1000000,'M')
AS TOTAL_FUNDED_AMOUNT
FROM Bank_loan_data
WHERE MONTH(issue_date) = 12
```

	TOTAL_FUNDED_AMOUNT
1	53M

PMTD total Funded Amount

```
SELECT CONCAT(SUM(loan_amount)/1000000,'M')
AS TOTAL_FUNDED_AMOUNT
FROM Bank_loan_data
WHERE MONTH(issue_date) = 11
```

	TOTAL_FUNDED_AMOUNT
1	47M

```
SELECT CONCAT(SUM(loan_amount)/1000000,'M')
AS TOTAL_FUNDED_AMOUNT
FROM Bank_loan_data
WHERE MONTH(issue_date) = 11 AND
YEAR(issue_date) = 2021
```

	TOTAL_FUNDED_AMOUNT
1	47M

3. Total Amount Received

```
SELECT SUM(total_payment)
AS TOTAL_AMOUNT_RECEIVED
FROM Bank_loan_data
```

	TOTAL_AMOUNT_RECEIVED
1	473070933

```
SELECT CONCAT(SUM(total_payment)/1000000,'M')
AS TOTAL_LOAN_APPLICATIONS
FROM Bank_loan_data
```

	TOTAL_LOAN_APPLICATIONS
1	473k

```
SELECT CONCAT(SUM(total_payment)/1000000,'M')
AS TOTAL_LOAN_APPLICATIONS
FROM Bank_loan_data
WHERE MONTH(issue_date) = 12
```

	TOTAL_LOAN_APPLICATIONS
1	58M

```
SELECT CONCAT(SUM(total_payment)/1000000,'M')
AS TOTAL_LOAN_APPLICATIONS
FROM Bank_loan_data
WHERE MONTH(issue_date) = 11
```

	TOTAL_LOAN_APPLICATIONS
1	50M

4. Average Interest Rate

```
SELECT ROUND(AVG(int_rate),4)*100 as AVG_INTEREST_RATE
FROM Bank_loan_data
```

	AVG_INTEREST_RATE
1	12.05

MTD AVG INTEREST RATE

```
SELECT ROUND(AVG(int_rate),4)*100 as AVG_INTEREST_RATE
FROM Bank_loan_data
WHERE MONTH(issue_date) = 12 AND YEAR(issue_date) = 2021
```

	AVG_INTEREST_RATE
1	12.36

PMTD AVG INTEREST RATE

```
SELECT ROUND(AVG(int_rate),4)*100 as AVG_INTEREST_RATE
FROM Bank_loan_data
WHERE MONTH(issue_date) = 11 AND YEAR(issue_date) = 2021
```

	AVG_INTEREST_RATE
1	11.94

5. Average Debt-to-Income Ratio (DTI)

```
SELECT ROUND(AVG(dti),4)*100 as AVG_INTEREST_RATE
FROM Bank_loan_data
```

	AVG_INTEREST_RATE
1	13.33

MTD Average Debt-to-Income Ratio (DTI)

```
SELECT ROUND(AVG(dti),4)*100 as AVG_INTEREST_RATE
FROM Bank_loan_data
WHERE MONTH(issue_date) = 12 AND YEAR(issue_date) = 2021
```

	AVG_INTEREST_RATE
1	13.67

PMTD Average Debt-to-Income Ratio (DTI)

```
SELECT ROUND(AVG(dti),4)*100 as AVG_INTEREST_RATE
FROM Bank_loan_data
WHERE MONTH(issue_date) = 11 AND YEAR(issue_date) = 2021
```

	AVG_INTEREST_RATE
1	13.3

6. Good Loan Application Percentage

```
SELECT
    (COUNT(CASE WHEN loan_status = 'Fully Paid' OR
        loan_status ='CURRENT' THEN id end))*100
        /
        COUNT(id) AS GOOD_LOAN_PERCENTAGE
FROM Bank_loan_data
```

	GOOD_LOAN_PERCENTAGE
1	86

```
SELECT
    (COUNT(CASE WHEN loan_status = 'Fully Paid' OR
        loan_status ='CURRENT' THEN id end))*100.0
        /
        COUNT(id) AS GOOD_LOAN_PERCENTAGE
FROM Bank_loan_data
```

	GOOD_LOAN_PERCENTAGE
1	86.175342181667

7. Total Good Loan Applications

```
SELECT COUNT(id) AS GOOD_LOAN_APPLICATIONS FROM Bank_loan_data
WHERE loan_status = 'Fully Paid' OR loan_status = 'Current'
```

	GOOD_LOAN_APPLICATIONS
1	33243

8. Total Good Loan Funded Amount

```
SELECT SUM(loan_amount) AS GOOD_LOAN_FUNDED_AMOUNT FROM Bank_loan_data
WHERE loan_status = 'Fully Paid' OR loan_status = 'Current'
```

	GOOD_LOAN_FUNDED_AMOUNT
1	370224850

9. Total Good Loan Amount Received

```
SELECT SUM(total_payment) AS GOOD_LOAN_AMOUNT_RECEIVED FROM Bank_loan_data
WHERE loan_status = 'Fully Paid' OR loan_status = 'Current'
```

	GOOD_LOAN_AMOUNT_RECEIVED
1	435786170

10. Bad Loan Application Percentage

```
SELECT
    (COUNT(CASE WHEN
        loan_status ='Charged Off' THEN id end))*100.0
        /
        COUNT(id) AS BAD_LOAN_PERCENTAGE
FROM Bank_loan_data
```

	BAD_LOAN_PERCENTAGE
1	13.824657818332

11. Total Bad Loan Applications

```
SELECT COUNT(id) AS BAD_LOAN_APPLICATIONS
FROM Bank_loan_data
WHERE loan_status = 'Charged Off'
```

	BAD_LOAN_APPLICATIONS
1	5333

12. Total Bad Loan Funded Amount

```
SELECT SUM(loan_amount) AS BAD_LOAN_FUNDED_AMOUNT
FROM Bank_loan_data
WHERE loan_status = 'Charged Off'
```

	BAD_LOAN_FUNDED_AMOUNT
1	65532225

13. Total Bad Loan Received Amount

```
SELECT SUM(total_payment) AS BAD_LOAN_AMOUNT_RECEIVED
FROM Bank_loan_data
WHERE loan_status = 'Charged Off'
```

	BAD_LOAN_AMOUNT_RECEIVED
1	37284763