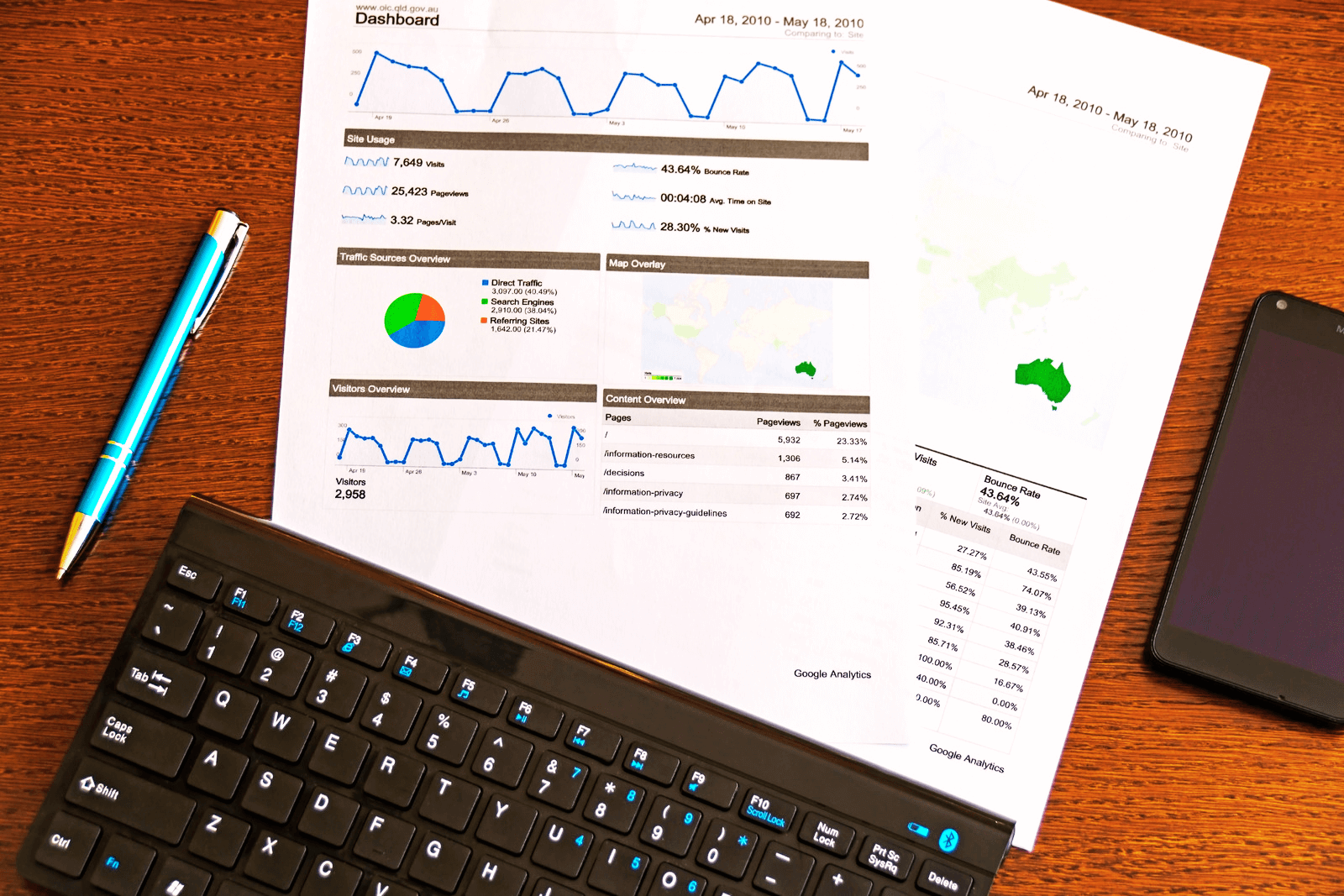
**PHASE-5 SUBMISSION**

**TEAM MEMBER: K.DIVYA**

**PROJECT TITLE : PRODUCT SALES ANALYSIS**

****

**Project Definition** : The project involves using IBM Cognos to analyze sales data and extract insights about top selling products, peak sales periods, and customer preferences. The objective is to help businesses improve inventory management and marketing strategies by understanding sales trends and customer behavior. This project includes defining analysis objectives, collecting sales data, designing relevant visualizations in IBM Cognos, and deriving actionable insights.

**Design of Thinking:**

**1.Analysis Objective:** These objectives guide the entire analysis process and help ensure that you derive meaningful insights to drive decision-making and business growth. Here are some common analysis objectives for product sales analysis:

* **Sales Performance Assessment**
* **Product Trend Identification**
* **Market Segmentation**
* **Customer Behaviour Analysis**

**2.Data Collection**: The quality and completeness of your data will significantly impact the accuracy and value of your analysis. Here a data collection for product sales analysis:

* **Sales Transactions Data**
* **Product Information**
* **Customer Data**

**3.Visualization Strategy:** Creating a visualization strategy for product sales analysis in IBM Cognos involves designing effective dashboards and reports that provide actionable insights to users.

* **Define Objectives**: Define what specific questions or insights do you want to address through visualization?
* **Select Visualization Types**: Choose appropriate visualization types based on the nature of your data and the insights you want to convey. Common visualization types in IBM Cognos include:
  + - **Bar charts** and **column charts** for comparing sales across products, regions, or time periods.
    - **Line charts** for showing trends over time.
    - **Pie charts** for illustrating product sales distribution.
    - **Heat maps** for analyzing sales by geographic regions.
    - **Tables** and **cross-tab** reports for displaying detaileddata.
* **Create Dashboards**: Design interactive dashboards that allow users to explore data and drill down into details. IBM Cognos offers a dashboarding feature that enables you to combine multiple visualizations on a single page.

**4.Actionable Insights**: Generating actionable insights from product sales analysis is crucial for making informed business decisions and improving sales strategies.

* **Top-Performing Products**
* **Seasonal Trends**
* **Pricing Optimization**
* **Inventory Management**

**Dataset link**: <https://www.kaggle.com/datasets/ksabishek/product-sales-data>

REC corp LTD. is small-scaled business venture established in India. They have been selling FOUR PRODUCTS for OVER TEN YEARS .

The products are P1, P2, P3 and P4.

Q1- Total unit sales of product 1

Q2- Total unit sales of product 2

Q3- Total unit sales of product 3

Q4- Total unit sales of product 4

S1- Total revenue from product 1

S2- Total revenue from product 2

S3- Total revenue from product 3

S4- Total revenue from product 4

**DATA PREPROCESSING**

Importing data:

In[1]: import pandas as pd

df=pd.read\_csv([https://www.kaggle.com/datasets/ksabishek/product- sales-data](https://www.kaggle.com/datasets/ksabishek/product-%20sales-data))

df.head(5)

Out[1]

| Unnamed: 0 | Date | Q-P1 | Q-P2 | Q-P3 | Q-P4 | S-P1 | S-P2 | S-P3 | S-P4 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 13-06-2010 | 5422 | 3725 | 576 | 907 | 17187.74 | 23616.50 | 3121.92 | 6466.91 |
| 1 | 14-06-2010 | 7047 | 779 | 3578 | 1574 | 22338.99 | 4938.86 | 19392.76 | 11222.62 |
| 2 | 15-06-2010 | 1572 | 2082 | 595 | 1145 | 4983.24 | 13199.88 | 3224.90 | 8163.85 |
| 3 | 16-06-2010 | 5657 | 2399 | 3140 | 1672 | 17932.69 | 15209.66 | 17018.80 | 11921.36 |
| 4 | 17-06-2010 | 3668 | 3207 | 2184 | 708 | 11627.56 | 20332.38 | 11837.28 | 5048.04 |

# Fetching rows and columns

In[2]: df.shape

Out[2]: (4600, 10)

#Fetching column names

In[3]: df.columns

Out[3]: Index(['Unnamed: 0', 'Date', 'Q-P1', 'Q-P2', 'Q-P3', 'Q-P4', 'S-P1', 'S-P2', 'S-P3', 'S-P4'], dtype='object')

# basic info

In[4]: df.info()

Out[4]: <class 'pandas.core.frame.DataFrame'>

RangeIndex: 4600 entries, 0 to 4599

Data columns (total 10 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Unnamed: 0 4600 non-null int64

1 Date 4600 non-null object

2 Q-P1 4600 non-null int64

3 Q-P2 4600 non-null int64

4 Q-P3 4600 non-null int64

5 Q-P4 4600 non-null int64

6 S-P1 4600 non-null float64

7 S-P2 4600 non-null float64

8 S-P3 4600 non-null float64

9 S-P4 4600 non-null float64

dtypes: float64(4), int64(5), object(1)

# Checking null values

In[5]: df.isnull().sum()

Out[5]: Unnamed: 0 0

Date 0

Q-P1 0

Q-P2 0

Q-P3 0

Q-P4 0

S-P1 0

S-P2 0

S-P3 0

S-P4 0

dtype: int64

No null values.

In[6]: df.duplicated().sum()

Out[6]: 0

No duplicate values.

In[7]: df.describe().T

Out[7]:

| Unnamed: 0 | count | mean | Std | min | 25% | 50% | 75% | max |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Unnamed: 0 |  | 4600.0 | 2299.500000 | 1328.049949 | 0.00 | 1149.750 | 2299.500 | 3449.250 |
| Q-P1 | 4600.0 | 4121.849130 | 2244.271323 | 254.00 | 2150.500 | 4137.000 | 6072.000 | 7998.00 |
| Q-P2 | 4600.0 | 2130.281522 | 1089.783705 | 251.00 | 1167.750 | 2134.000 | 3070.250 | 3998.00 |
| Q-P3 | 4600.0 | 3145.740000 | 1671.832231 | 250.00 | 1695.750 | 3202.500 | 4569.000 | 6000.00 |
| Q-P4 | 4600.0 | 1123.500000 | 497.385676 | 250.00 | 696.000 | 1136.500 | 1544.000 | 2000.00 |
| S-P1 | 4600.0 | 13066.261743 | 7114.340094 | 805.18 | 6817.085 | 13114.290 | 19248.240 | 25353.66 |
| S-P2 | 4600.0 | 13505.984848 | 6909.228687 | 1591.34 | 7403.535 | 13529.560 | 19465.385 | 25347.32 |
| S-P3 | 4600.0 | 17049.910800 | 9061.330694 | 1355.00 | 9190.965 | 17357.550 | 24763.980 | 32520.00 |
| S-P4 | 4600.0 | 8010.555000 | 3546.359869 | 1782.50 | 4962.480 | 8103.245 | 11008.720 | 14260.00 |

# Changing dtype

In[8]: from datetime import datetime as dt

df[df["Date"]=="31-9-2010"]

Out[8]:

| Unnamed: 0 | Date | Q-P1 | Q-P2 | Q-P3 | Q-P4 | S-P1 | S-P2 | S-P3 | S-P4 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 109 | 31-9-2010 | 4986 | 342 | 4978 | 558 | 15805.62 | 2168.28 | 26980.76 | 3978.54 |

# **Exploratory Data Analysis**

*# Total unit sales Product 1, Product 2, Product 3, Product 4*

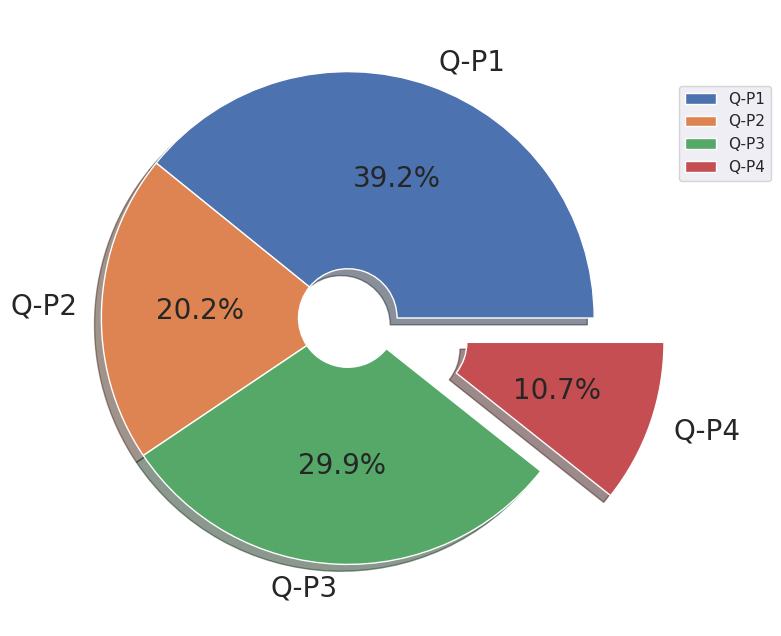
q = df[["Q-P1","Q-P2","Q-P3","Q-P4"]].sum()

plt.figure(figsize=(8,8))

plt.pie(q,labels=df[["Q-P1","Q-P2","Q-P3","Q-P4"]].sum().index,shadow=True,autopct="**%0.01f%%**",textprops={"fontsize":20},wedgeprops={'width': 0.8},explode=[0,0,0,0.3])

plt.legend(loc='center right', bbox\_to\_anchor=(1.2, 0.8));

Output:



*# Total Revenue percent from sales from Product 1, Product 2, Product 3, Product 4*

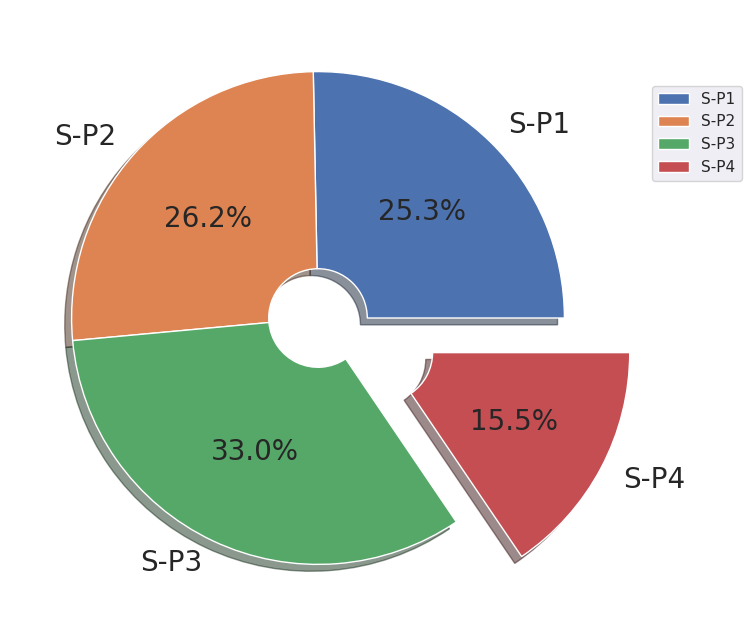
s=df[["S-P1","S-P2","S-P3","S-P4"]].sum()

plt.figure(figsize=(8,8))

plt.pie(s,labels=df[["S-P1","S-P2","S-P3","S-P4"]].sum().index,shadow=True,autopct="**%0.01f%%**",textprops={"fontsize":20},wedgeprops={'width': 0.8},explode=[0,0,0,0.3])

plt.legend(loc='center right', bbox\_to\_anchor=(1.2, 0.8))

Output:

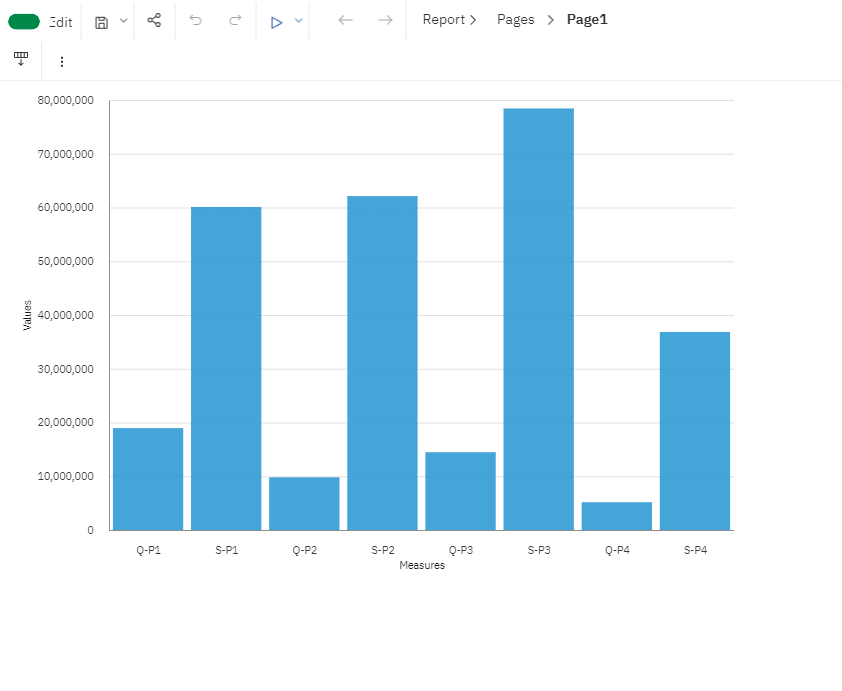


**DATA VISUALIZATION USING IBM COGNOS**

Products are analysed and visualized by creating different reports and dashboard.

And deriving actionable insights from the analysis of products sales

**BAR CHART**:

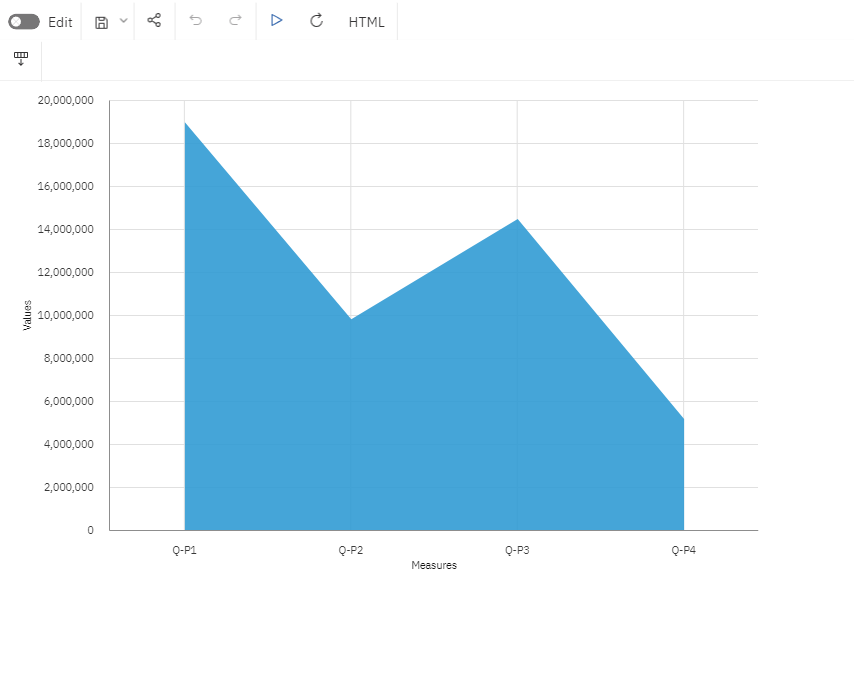


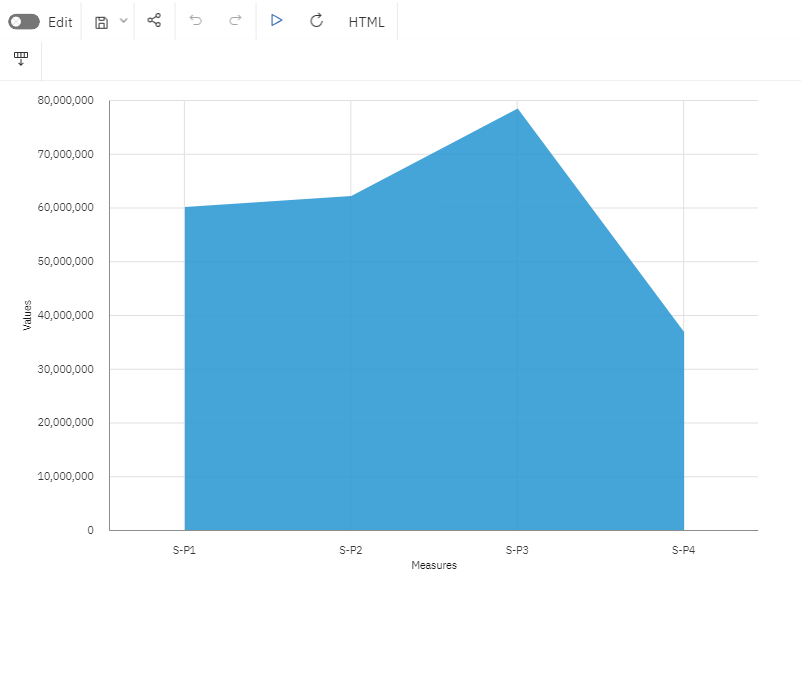
Above represents **bar chart visualisation**,

by comparing the total sales of product Q-P1,Q-P2,Q-P3,Q-P4 with total revenue from product

of S-P1,S-P2,S-P3 and S-P4.

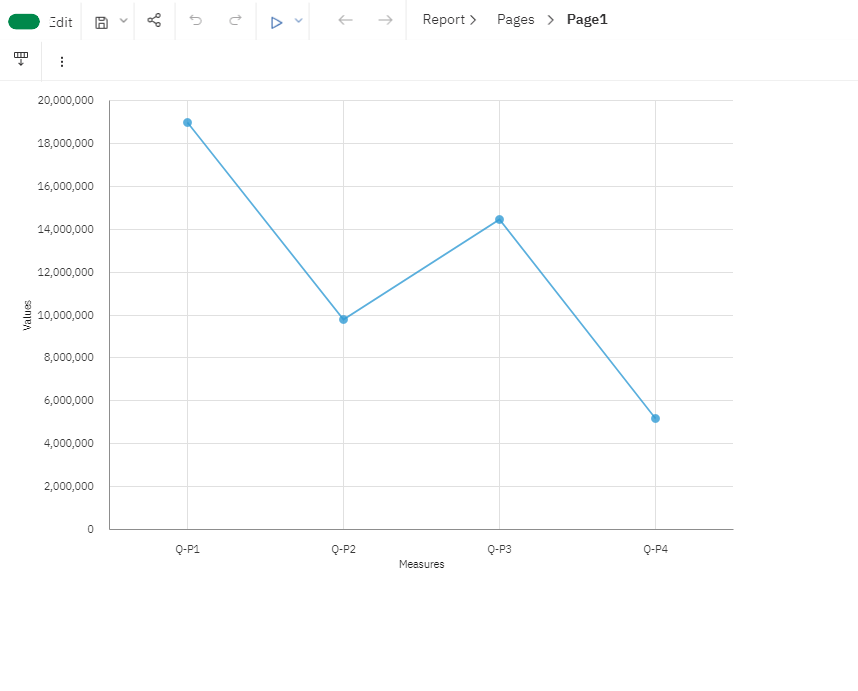
**AREA CHART**:



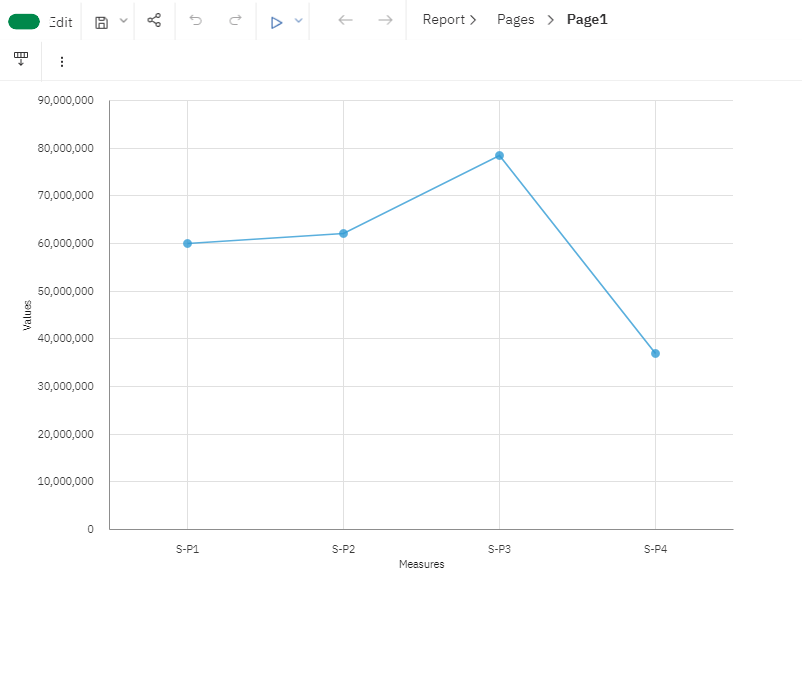


Above represents **area chart** , analysing the total unit of sales of product Q-P1,Q-P2,Q-P3 and Q-P4 and the total revenue from product S-P1,S-P2,S-P3 and S-P4.

**LINE CHART**:



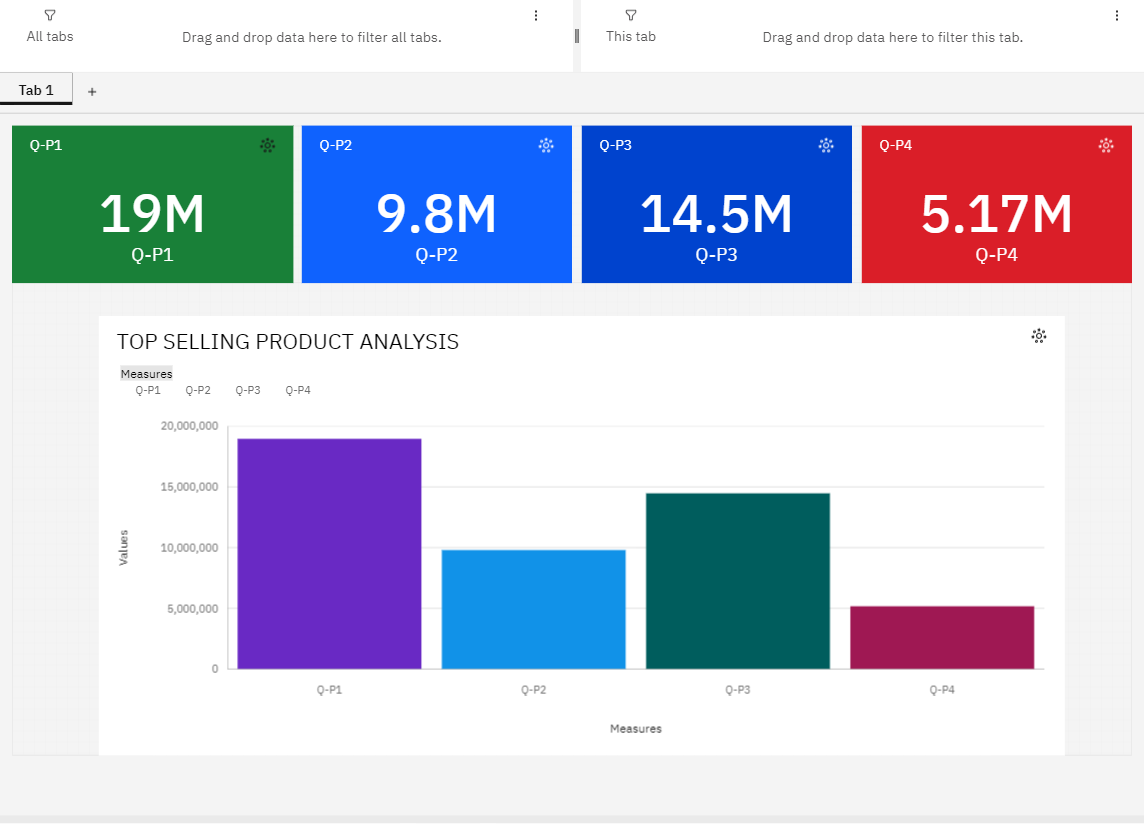
Above represents **line chart** , analysing the total unit of sales of product Q-P1,Q-P2,Q-P3 and Q-P4.



Above represents **line chart** ,analysing the total revenue from product S-P1,S-P2,S-P3 and S-P4.

Now ,lets derive the actionable insights using interactive dashboards using IBM cognos.

**TOP SELLING PRODUCT**:



The sales products are Q-P1,Q-P2,Q-P3 and Q-P4.

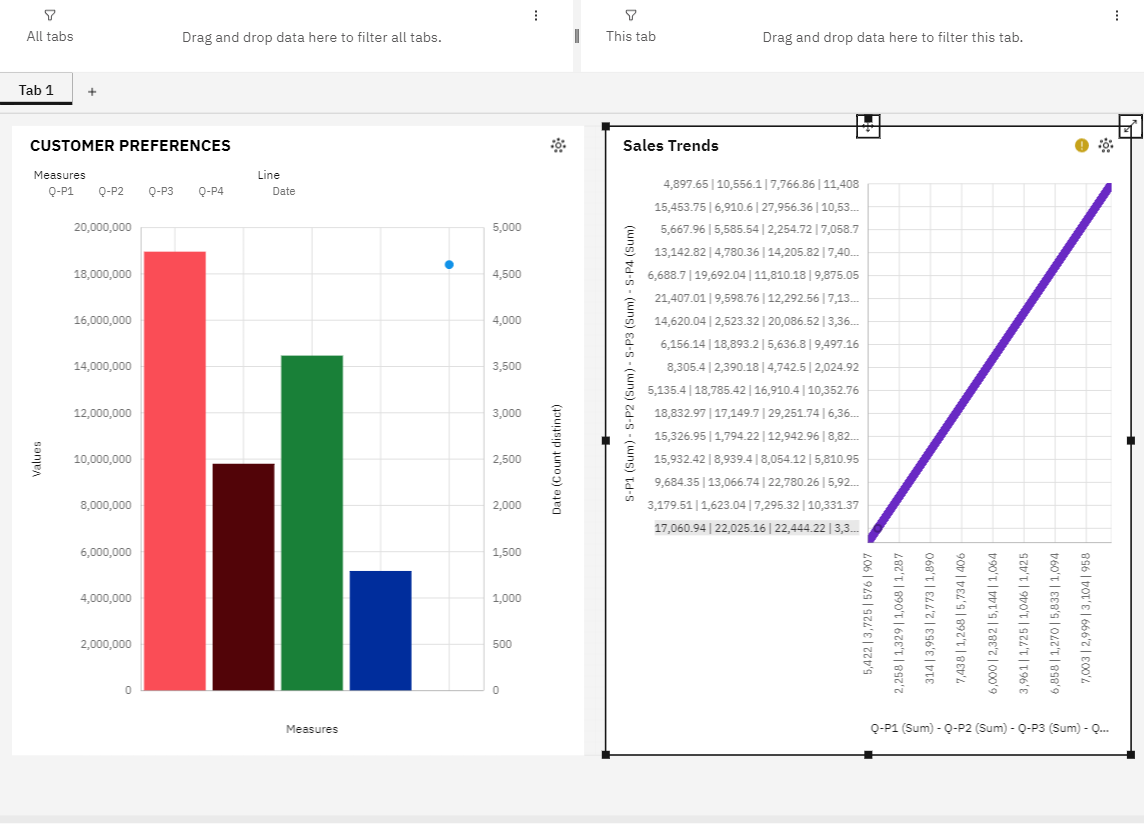
From above visualization, we come to know the product Q-P1 is a **top selling product** compare to all other products.

Hence allocate more Q-P1 product for good sales.

**CUSTOMER PREFERENCES**:

Q-P1 is a top selling product ,so customer preferences more on Q-P1 and next preference product is Q-P3.

Q-P4 product is least preference when compare to all other products by customers.

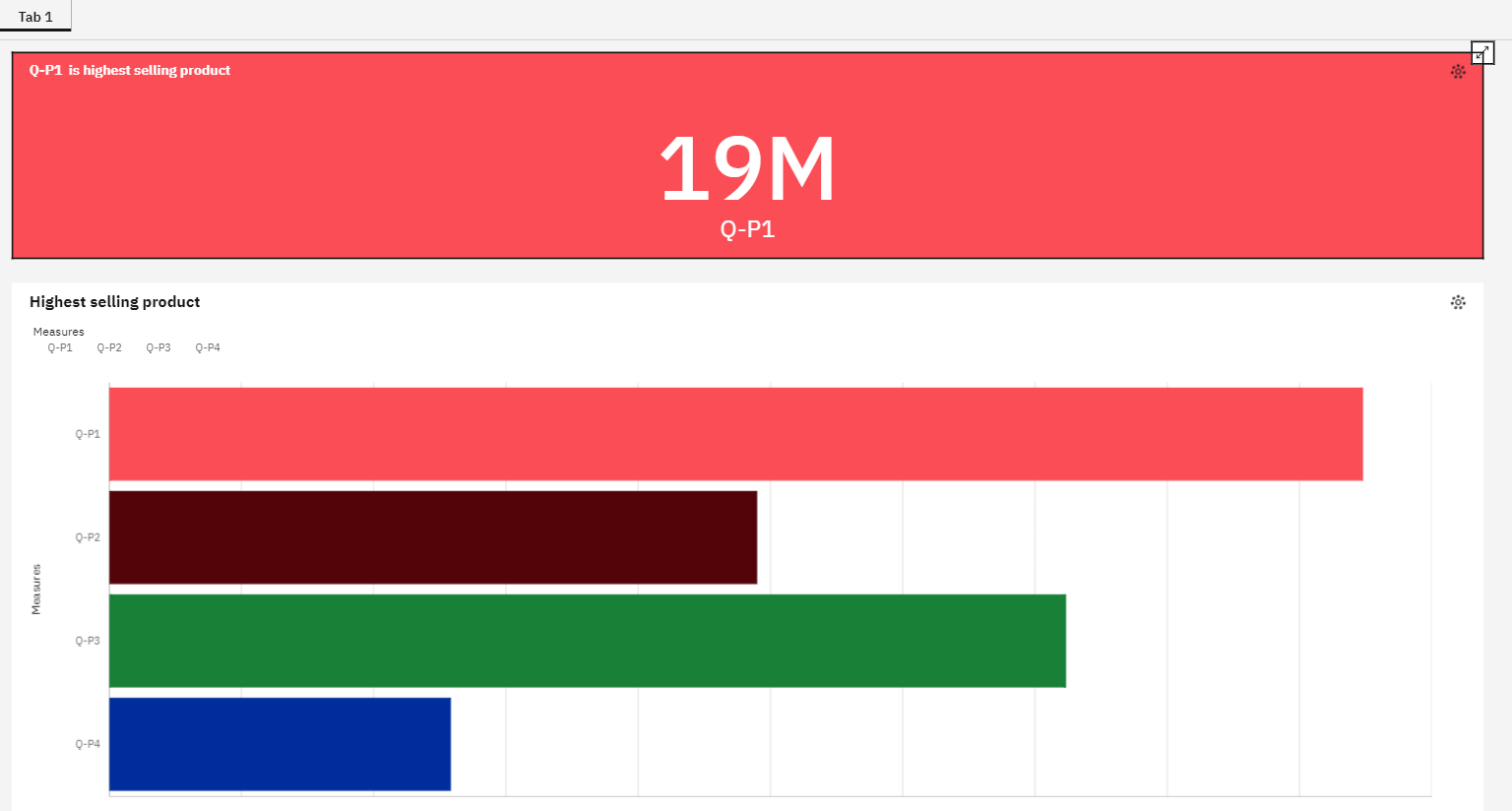


**SALES TRENDS:**

In above visualization includes the sales trends ,

which explains a “**Positive trends**” sales on the products. Hence sales trends for the products is good.

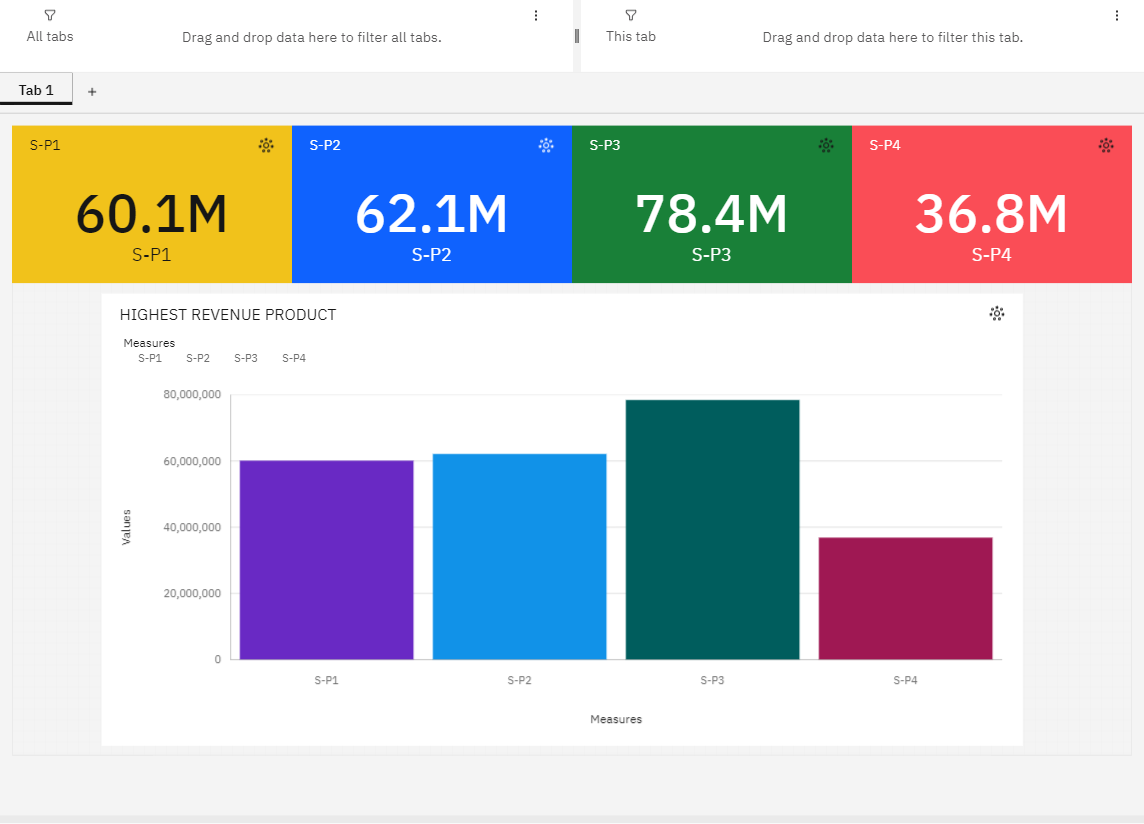
**PRODUCT WITH HIGHEST SALES:**



Where Q-P1,Q-P2,Q-P3 and Q-P4 are referred as the sales products.

The above visualization ,shows highest sales product such that Q-P1 is highest selling product and second is Q-P3 ,next to that Q-P2 and Q-P4.

**HIGHEST REVENUE PRODUCT:**

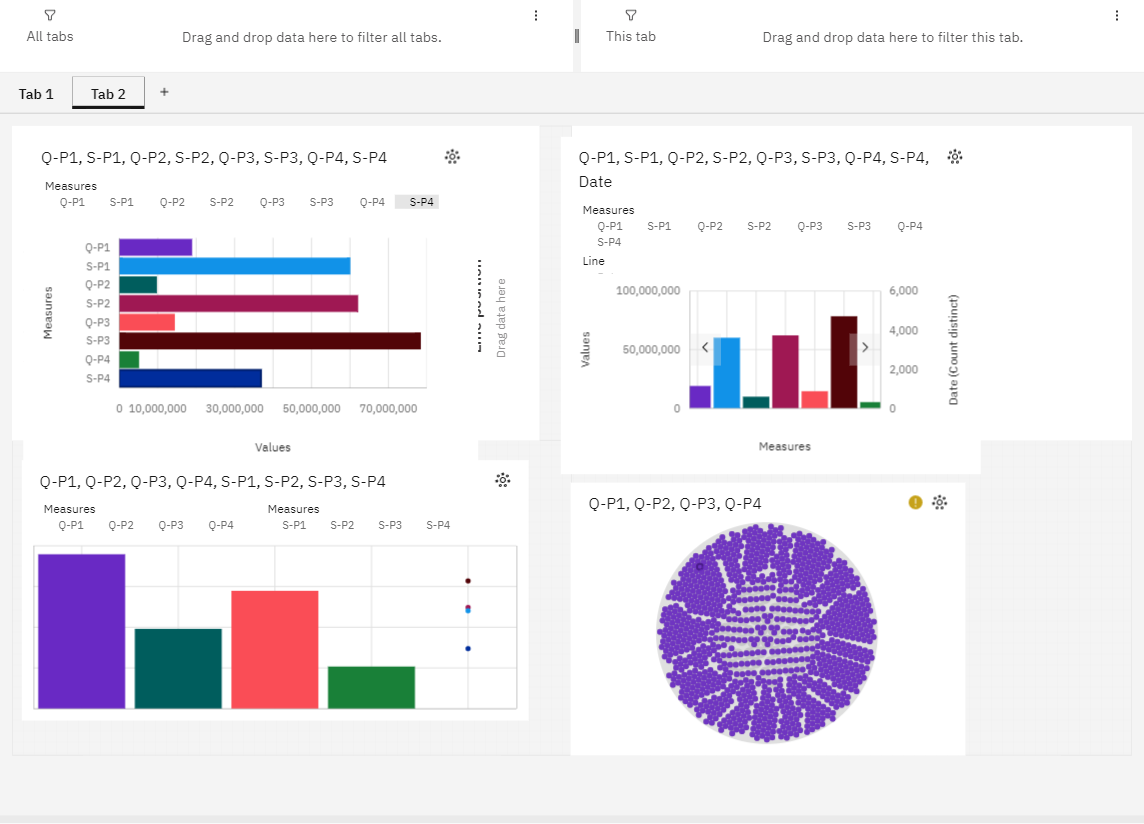


Where S-P1,S-P2,S-P3 and S-P4 denotes the Revenue of the products.

In above visualization shows the analysis of highest revenue product .As per the data ,Q-P1 is the top selling product but when comes to revenue ,

S-P3 as **highest revenue product** when compare to all other product revenue (S-P1,S-P2,S-P4).

**Different types of visualization done using single dashboard :**



Above visualization compares both product **sales**(Q-P1,Q-P2,Q-P3,Q-P4) and **revenue**(S-P1,S-P2,S-P3,S-P4).