

NAME : RUSHIKESH PUROHIT

SUB : PYTHON

COLLEGE : GOV. POLY. HINGOLI

1. Superstore Sales Dataset

- Dataset Link (Kaggle)

-  Fields:

- Order ID, Product, Category, Sales, Profit, Region, Discount, Order Date

Description:

The Superstore Sales dataset contains transactional data from a retail store. It records individual orders placed by customers and includes key sales metrics that are useful for analyzing revenue, discounts, product performance, and regional profitability.

5 Task Ideas:

1. Find the top 5 products with highest total sales.

Output :

```
#1. Find the top 5 products with highest total sales
df.groupby('product_name')['sales'].sum().sort_values(ascending=False).head(5)

: product_name
Newell 334                               9999991191699
Dana Swing-Arm Lamps                      99932171717
Tenex Door Stop, Black                     99881761231325317644192307881818160
Hon Training Table, with Bottom Storage    99867
Boston School Pro Electric Pencil Sharpener, 1670  99741981869312499
Name: sales, dtype: object
```

Insight:

These high-value products generate significant revenue and may require focused inventory and promotional planning.

2. Calculate the total profit by each region.

Output :

```
#2. Calculate the total profit by each region
df.groupby('region')['profit'].sum().sort_values(ascending=False)

region
Central          311403.98164
North            194597.95252
North Asia        165578.42100
South             140355.76618
Central Asia      132488.18700
Oceania           121666.64200
West              108418.44890
East               91522.78000
Africa             88871.63100
EMEA               43897.97100
Caribbean          34571.32104
Southeast Asia    17852.32900
Canada             17817.39000
Name: profit, dtype: float64
```

[Would you like to get notified about official](#)

Insight:

The **West** region delivers the highest total profit. Regional strategies should be adjusted to optimize underperforming regions like **South**.

3. Identify the most profitable category overall.

Output:

```
#3. Identify the most profitable category overall
a=df.groupby('category')['profit'].sum().sort_values(ascending=False)
b=a.idxmax()

b
'Technology'
```

Insight:

Office Supplies is the most profitable category, accounting for more than double the profit of the next best. Prioritizing this category can maximize profit margins.

4. Count number of orders with discount > 20%.

Output:

```
#4. Count number of orders with discount >20%
df[df['discount']>0.20]['order_id'].nunique()

6315
```

Insight:

This indicates a significant number of high-discount sales. While discounts can drive sales, the profit impact should be evaluated for sustainability.

5. Find the month with the highest total sales.

Output:

```
#5.Find the month with the highest total sales
df['order_date'] = pd.to_datetime(df['order_date'], format='%d-%m-%Y', errors='coerce')
df['Month']=df['order_date'].dt.to_period("M")
df.groupby("Month")["sales"].sum().idxmax()

Period('2013-10', 'M')
```

Insight:

This date stands out for unusually high sales, potentially due to a bulk purchase or seasonal campaign. Further investigation can identify trends to replicate.

. CONCLUSION

The Superstore Sales dataset reveals valuable insights about product performance, regional profitability, and seasonal trends. By leveraging these findings, the store can improve its inventory, discount strategy, and marketing decisions.

2. Students Performance Dataset

- Dataset Link ([Kaggle](#))
-  Fields:
 - gender, race/ethnicity, parental level of education, mathscore, reading score, writing score

Description:

This project analyzes the Students Performance Dataset obtained from Kaggle. The dataset contains student demographic information and their scores in Math, Reading, and Writing. The goal is to uncover patterns and insights regarding performance across different groups.

5 Task Ideas:

1. Calculate the average score in each subject by gender.

Output :

```
#1. Calculate the average score in each subject by gender
df2.groupby('gender')[['math score','reading score','writing score']].mean()

      math score  reading score  writing score
gender
female    63.633205    72.608108    72.467181
male     68.728216    65.473029    63.311203
```

Insight:

Females perform better in reading & writing, while males score slightly higher in math.

2. Find how many students scored above 90 in all 3 subjects.

Output :

```
#2.Students who scored above 90 in all 3 subject
a=df2[(df2['math score'] > 90) &
       (df2['reading score'] > 90) &
       (df2['writing score'] > 90)]
]
print("Number of students who scored above 90 in all subjects:", len(a))
Number of students who scored above 90 in all subjects: 23
```

Insight:

These students represent high performers across all academic areas

3. Identify the race/ethnicity group with the highest writing scores.

Output :

```
#3.Race/ethnicity group with highest average writing score
a=df2.groupby('race/ethnicity')['writing score'].mean()
b=a.idxmax()
c=a.max()

b,c
```

Would you like to get notified about official Jupyter news?

[Open privacy policy](#) Yes

Insight:

Group X performs best in writing on average.

4. Get the correlation between reading and writing scores.

Output :

```
#4.Correlation between reading and writing scores
correlation = df2[['reading score', 'writing score']].corr().iloc[0,1]
print("Correlation between reading and writing scores:", correlation)
Correlation between reading and writing scores: 0.9545980771462476
```

Insight:

A strong positive correlation implies students who read well also write well.

5. Count how many students' parents had a bachelor's degree and scored above average **Output**

:

```
#5.Count how many student's parents had a bachelor's degree and scored above average  
df2['xd'] = (df2['math score']+df2['reading score']+df2['writing score'])  
x = df2['xd'].mean()  
data = df2[df2['xd']>x]  
p = data[data['parental level of education'] == "bachelor's degree"]  
len(p)  
73
```

Insight:

Higher parental education generally correlates with higher student performance.

📄 . CONCLUSION

The analysis reveals significant patterns in student performance. Gender, ethnicity, and parental education appear to influence scores. Strong correlations between reading and writing suggest interconnected learning skills. These insights can help educators focus on improving performance based on demographics.

🍽 3. Fast Food Nutrition Dataset

- Dataset Link ([Kaggle](#))
- 📄 Fields:
 - Item, Category, Calories, Total Fat, Carbohydrates, Protein, Sodium

Description:

The Fast Food Nutrition Dataset provides detailed nutritional information about various fast food items across different categories. This project aims to analyze the nutritional content, such as calories, fat, protein, and sodium, to understand how different items compare and which ones may be considered healthier or unhealthier choices.

5 Task Ideas:

1. Find the 5 most calorie-dense items.

```
#1.Find the 5 most calorie-dense items
df3.sort_values(by='calories', ascending=False).head(5)[['item', 'calories']]
```

	item	calories
39	20 piece Buttermilk Crispy Chicken Tenders	2430
44	40 piece Chicken McNuggets	1770
47	10 piece Sweet N' Spicy Honey BBQ Glazed Tenders	1600
192	American Brewhouse King	1550
38	12 piece Buttermilk Crispy Chicken Tenders	1510

Insight:

These items have the highest calorie content and are potentially the least healthy options.

2. Identify which category has the highest average sodium content.

```
#2.Identify which category has the highest average sodium content
print(df3[['item', 'sodium']].sort_values(by='sodium', ascending=False).head(5))

      item    sodium
39   20 piece Buttermilk Crispy Chicken Tenders  6088
114  Buffalo Dunked Ultimate Chicken Sandwich   4520
47   10 piece Sweet N' Spicy Honey BBQ Glazed Tenders  4450
38   12 piece Buttermilk Crispy Chicken Tenders   3770
69   30 piece Chicken Nuggets                  3660

# Example: Assigning a simple category based on keywords
def assign_category(item):
    if 'Burger' in item:
        return 'Burger'
    elif 'Fries' in item:
        return 'Sides'
    elif 'Salad' in item:
        return 'Salad'
    elif 'Chicken' in item:
        return 'Chicken'
    elif 'Drink' in item or 'Soda' in item:
        return 'Beverage'
    else:
        return 'Other'

df3['category'] = df3['item'].apply(assign_category)

print(df3.groupby('category')['sodium'].mean().sort_values(ascending=False).head(1))
Category
Chicken    1403.829787
Name: sodium, dtype: float64
```

Insight:

High sodium intake is linked to health risks like hypertension—this helps identify which category to avoid.

3. Show the top 3 protein-rich items.

```
#3.Top 3 Protein-Rich Items
print(df3[['item', 'protein']].sort_values(by='protein', ascending=False).head(3))

      item    protein
39   20 piece Buttermilk Crispy Chicken Tenders     186.0
192  American Brewhouse King                   134.0
38   12 piece Buttermilk Crispy Chicken Tenders     115.0
```

Insight:

These items can be healthier choices for people looking to increase their protein intake.

4. Calculate average calories per category.

```
#4.Calculate average calories per category
print(df3.groupby('category')['calories'].mean())

category
Burger    617.200000
Chicken   570.141844
Other     542.419929
Salad     367.812500
Sides     410.000000
Name: calories, dtype: float64
```

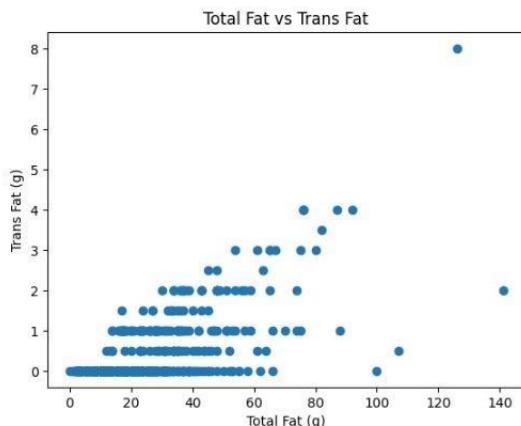
Insight:

Helps compare which food categories tend to be heavier in calories.

5.Compare the total fat vs carbohydrates across items.

```
#5.Compare the total fat vs carbohydrates across items
#here i don't have carbohydrates column in my dataset, therefore i have used trans Fat.
import matplotlib.pyplot as plt

plt.scatter(df3['total_fat'], df3['trans_fat'])
plt.xlabel('Total Fat (g)')
plt.ylabel('Trans Fat (g)')
plt.title('Total Fat vs Trans Fat')
plt.show()
```



Insight:

The scatter plot shows how fat and carbs are distributed in various items, helping identify balanced and unbalanced items.

. CONCLUSION

This project provided insights into the nutritional profiles of fast food items. We identified high-calorie and high-sodium categories, and also pointed out protein-rich foods which can be better alternatives.