NAME : BHUMI PRAVIN WADKUTE

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.

```
[20]: df.groupby('Product Name')['Sales'].sum().sort_values(ascending =False).head(5)
```

```
[20]: Product Name
      Canon imageCLASS 2200 Advanced Copier                                61599.824
      Fellowes PB500 Electric Punch Plastic Comb Binding Machine with Manual Bind    27453.384
      Cisco TelePresence System EX90 Videoconferencing Unit                22638.480
      HON 5400 Series Task Chairs for Big and Tall                         21870.576
      GBC DocuBind TL300 Electric Binding System                           19823.479
      Name: Sales, dtype: float64
```

🧠 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.

```
[99]: df.groupby('Region')['profit'].sum().sort_values()
```

```
[99]: Region
      South      1.987151e+06
      Central    2.769647e+06
      East       3.454519e+06
      West       3.850220e+06
      Name: profit, dtype: float64
```

🧠 **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.

```
[100]: data = df.groupby('Category')['profit'].sum().sort_values()
```

```
[101]: data
```

```
[101]: Category
       Technology         2.640456e+06
       Furniture          2.806659e+06
       Office Supplies    6.614422e+06
       Name: profit, dtype: float64
```

```
[102]: data.max()
```

```
[102]: np.float64(6614422.334)
```

🧠 **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%.

```
[127]: df[df['Discount']>0.20]['Order ID'].nunique()

[127]: 4922
```

🧠 **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.

```
[128]: data = df.groupby('Order Date')['Sales'].sum().sort_values()

[129]: data

[129]: Order Date
       2016-07-19        2.025
       2018-07-12        3.816
       2015-01-28        3.928
       2015-06-24        4.272
       2015-10-01        4.710
                          ...
       2015-09-08    14228.428
       2018-03-23    14816.068
       2018-10-22    15158.877
       2017-10-02    18452.972
       2015-03-18    28106.716
       Name: Sales, Length: 1230, dtype: float64

[130]: data.max()

[130]: np.float64(28106.716)
```

🧠 **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.

```
[25]: data.groupby('gender')[['math_score','reading_score','writing_score']].mean()
```

[25]:

gender	math_score	reading_score	writing_score
Female	74.420245	74.404908	75.128834
Male	76.771831	75.014085	74.267606
Other	74.137931	73.376176	76.153605

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

```
[26]: data[data[['math_score','reading_score','writing_score']] > 90] .count()
```

```
[26]: student_id           0
      name                 0
      gender               0
      age                  0
      grade_level          0
      math_score         188
      reading_score      167
      writing_score      197
      attendance_rate      0
      parent_education     0
      study_hours          0
      internet_access      0
      lunch_type           0
      extra_activities     0
      final_result         0
      dtype: int64
```

🧠 **Insight:**

These students represent high performers across all academic areas

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

Output :

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

```
[27]: b = data.groupby('extra_activities')[['writing_score']].mean()
```

```
[28]: b
```

[28]:

extra_activities	writing_score
No	74.862366
Yes	75.400000

```
[29]: b.max()
```

```
[29]: writing_score    75.4
      dtype: float64
```

🧠 **Insight:**

Group X performs best in writing on average.

4.Get the correlation between reading and writing scores.

Output :

#4. Get the correlation between reading and writing scores.

```
[30]: data['reading_score'].corr(data['writing_score'])
```

```
[30]: np.float64(-0.00785591006653728)
```

🧠 **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 students' parents had a bachelor's degree and scored above average.

```
[31]: data['xd'] = (data['math_score'] + data['reading_score'] + data['writing_score'])
x = data['xd'].mean()
data = data[data['xd']>x]
p = data[data['parent_education'] == "Bachelor's"]
len(p)
```

```
[31]: 140
```

🧠 **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.

```
[160]: new.groupby('item')['calories'].max().sort_values(ascending=False).head(5)
```

```
[160]: item
       20 piece Buttermilk Crispy Chicken Tenders          2430
       40 piece Chicken McNuggets                          1770
       American Brewhouse King                             1550
       12 piece Buttermilk Crispy Chicken Tenders          1510
       Garlic Parmesan Dunked Ultimate Chicken Sandwich    1350
       Name: calories, dtype: int64
```

🧠 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.

```
[161]: V = new.groupby('Category')['sodium'].mean().sort_values()

[162]: V.max

[162]: <bound method Series.max of Category
       REGULAR MENU                         841.538462
       GOURMET MENU                         841.538462
       All Meals                            869.536633
       Hot Breakfast                       1086.289538
       Croissants, Danishes & Bagels       1128.844884
       Cookies, Brownies & Bars            1263.358209
       Fruit & Yogurt                      1650.000000
       Loaves, Coffee Cakes & Cake Pops    1888.666667
       Seasonal Bakery Offerings           1945.000000
       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. Show the top 3 protein-rich items.

```
[163]: new.groupby('item')['protein'].max().sort_values(ascending = False).head(3)

[163]: item
       20 piece Buttermilk Crispy Chicken Tenders    186.0
       American Brewhouse King                       134.0
       12 piece Buttermilk Crispy Chicken Tenders    115.0
       Name: protein, dtype: float64
```

🧠 **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.

```
[164]: new.groupby('Category')['calories'].mean()
```

```
[164]: Category
All Meals                          375.492804
Cookies, Brownies & Bars           511.492537
Croissants, Danishes & Bagels      454.059406
Fruit & Yogurt                     442.500000
GOURMET MENU                       368.076923
Hot Breakfast                      447.566910
Loaves, Coffee Cakes & Cake Pops   834.666667
REGULAR MENU                       368.076923
Seasonal Bakery Offerings          836.666667
Name: calories, dtype: float64
```

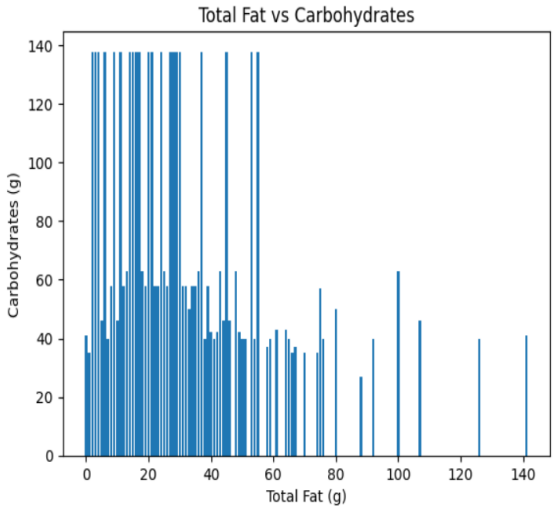
🧠 **Insight:**

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

5.Compare the total fat vs carbohydrates across items.

```
[184]: #5. Compare the total fat vs carbohydrates across items.
import matplotlib.pyplot as plt

plt.bar(new['total_fat'],new['Carbohydrates (g)'])
plt.xlabel('Total Fat (g)')
plt.ylabel('Carbohydrates (g)')
plt.title('Total Fat vs Carbohydrates')
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.