CoGrammar

Welcome to this session

Skills Bootcamp:

Tutorial

The session will start shortly...

Questions? Drop them in the chat.
We'll have dedicated moderators
answering questions.

Safeguarding & Welfare

We are committed to all our students and staff feeling safe and happy; we want to make sure there is always someone you can turn to if you are worried about anything.

If you are feeling upset or unsafe, are worried about a friend, student or family member, or you feel like something isn't right, speak to our safeguarding team:

Ian Wyles
Designated Safeguarding Lead

Simone Botes

Nurhaan Snyman

Rafiq Manan

Ronald Munodawafa

Tevin Pitts

Scan to report a safeguarding concern

or email the Designated Safeguarding Lead:
Ian Wyles
safeguarding@hyperiondev.com

CoGrammar HyperionDev

Skills Bootcamp Full Stack Web Development

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(Fundamental British Values: Mutual Respect and Tolerance)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. We will be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: **Questions**

CoGrammar

Skills Bootcamp Cloud Web Development

- For all **non-academic questions**, please submit a query: ***www.hyperiondev.com/support***
- **Report a safeguarding incident: *www.hyperiondev.com/safeguardreporting***
- We would love your feedback on lectures: ***Feedback on Lectures.***
- Find all the lecture **content** in your **Lecture Backpack** on GitHub.
- If you are hearing impaired, kindly use your computer's function through Google chrome to enable captions.

CoGrammar

Skills Bootcamp Progression Overview

✅ Criterion 1 - Initial Requirements

Specific achievements within the first two weeks of the program.

To meet this criterion, students need to, by no later than **01 December 2024 (C11)** or **22 December 2024 (C12):**

- **Guided Learning Hours** (GLH): Attend a minimum of 7-8 GLH per week (lectures, workshops, or mentor calls) for a total minimum of **15 GLH**.
- **Task Completion:** Successfully complete the **first 4 of the assigned tasks**.

✅ Criterion 2 - Mid-Course Progress

Progress through the successful completion of tasks within the first half of the program.

To meet this criterion, students should, by no later than **12 January 2025 (C11)** or **02 February 2025 (C12):**

- **Guided Learning Hours** (GLH): Complete at least **60 GLH**.
- **Task Completion :** Successfully complete the **first 13 of the assigned tasks**.

Skills Bootcamp Progression Overview

✅ Criterion 3 – End-Course Progress

Showcasing students' progress nearing the completion of the course.

To meet this criterion, students should:

- **Guided Learning Hours** (GLH)**:** Complete the **total minimum required GLH,** by the **support end date.**
- **Task Completion : Complete all mandatory tasks**, including any necessary resubmissions, by the end of the bootcamp, **09 March 2025 (C11)** or **30 March 2025 (C12).**

✅ Criterion 4 - Employability

Demonstrating progress to find employment.

To meet this criterion, students should:

- **Record an Interview Invite:** Students are required to record proof of invitation to an interview by **30 March 2025 (C11)** or **04 May 2025 (C12)**.
  - **South Holland Students** are required to proof and interview by **17 March 2025.**
- **Record a Final Job Outcome :** Within 12 weeks post-graduation, students are required to record a job outcome.

CoGrammar

Learning Outcomes

❖ Use the Matplotlib and Seaborn to create common data visualisations

❖ Analyse and interpret visualisations

❖ Identify best practices for effective data visualization

CoGrammar

Lecture Overview

- → Basic Visualisation
- → Break
- → Advanced Visualisation
- → Q&A

CoGrammar

Data Visualisation

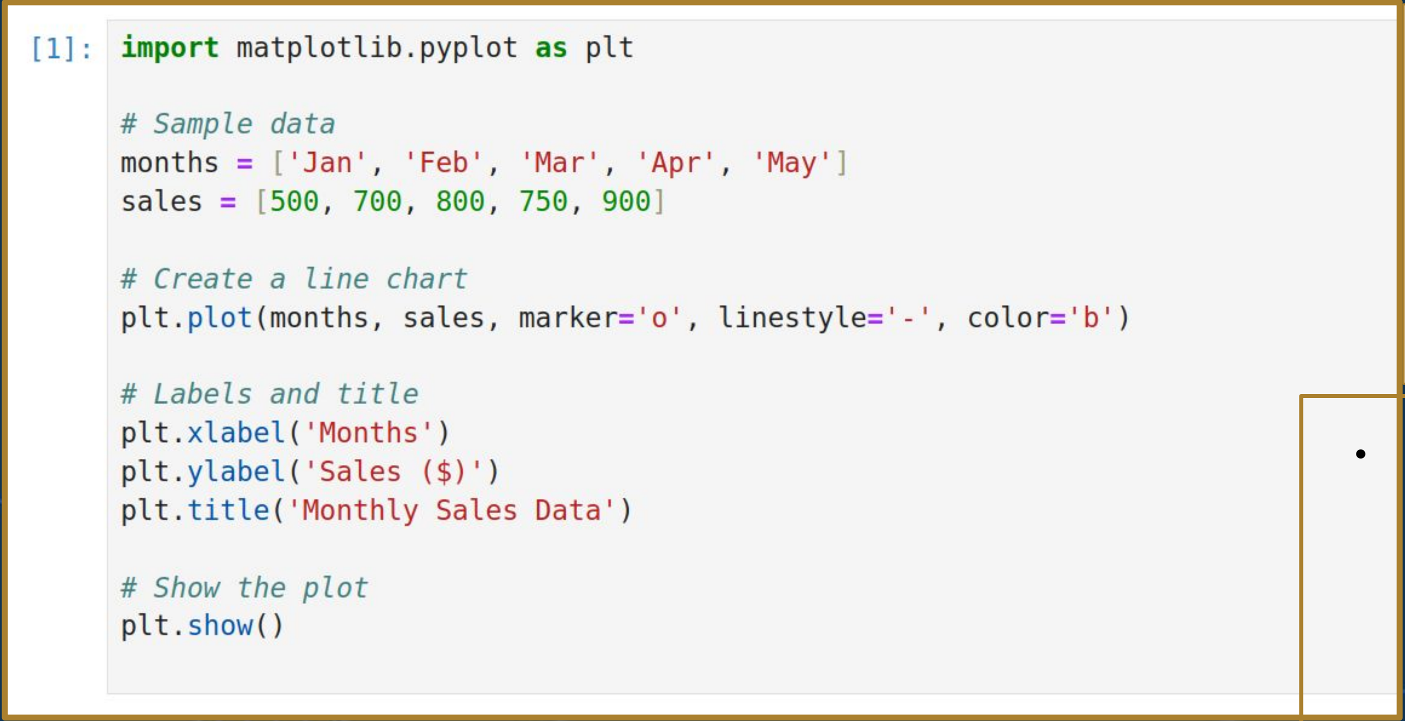
CoGrammar

📝 Exercise

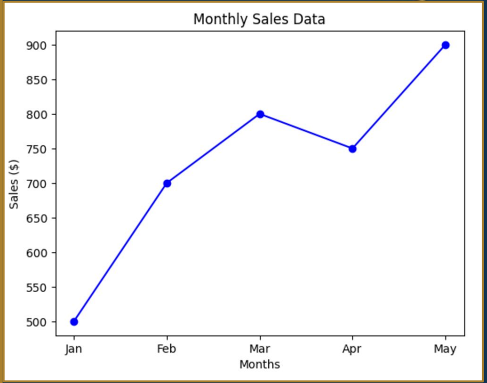
❖ "For the following datasets, which visualisation technique would you use?"

- ➢ Monthly sales data for a company.
- ➢ Exam scores of students in a class.
- ➢ The relationship between ad spend and revenue.
- ➢ Market share of smartphone brands.

Basic Plotting with Matplotlib



```
[1]: import matplotlib.pyplot as plt

     # Sample data
     months = ['Jan', 'Feb', 'Mar', 'Apr', 'May']
     sales = [500, 700, 800, 750, 900]

     # Create a line chart
     plt.plot(months, sales, marker='o', linestyle='-', color='b')

     # Labels and title
     plt.xlabel('Months')
     plt.ylabel('Sales ($)')
     plt.title('Monthly Sales Data')

     # Show the plot
     plt.show()
```



- Why is a line chart a good choice here?

CoGrammar

📊 Boxplots: Identifying Outliers

❖ **What is a Boxplot?**

➢ A boxplot (also called a box-and-whisker plot) is a standardized way of displaying data distribution based on a five-number summary:

- Minimum – The smallest data point (excluding outliers).
- First quartile (Q1) – The 25th percentile (lower quartile).
- Median (Q2) – The 50th percentile (middle value).
- Third quartile (Q3) – The 75th percentile (upper quartile).
- Maximum – The largest data point (excluding outliers).

❖ Outliers are data points that fall far outside the normal range of the dataset. They are usually plotted as individual points beyond the whiskers.

CoGrammar

How to Identify Outliers Using a Boxplot?

❖ Whiskers extend to 1.5 × IQR (Interquartile Range) beyond Q1 and Q3.

❖ Outliers are any data points that fall:
  - ➢ Below Q1 - 1.5 × IQR
  - ➢ Above Q3 + 1.5 × IQR

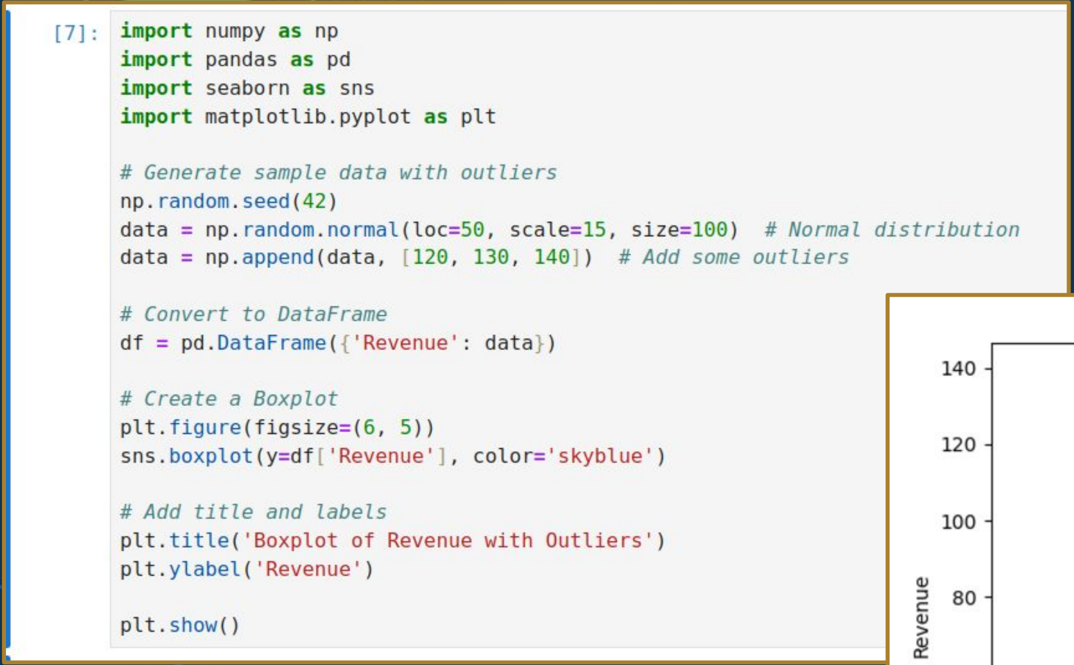
❖ Formula for IQR (Interquartile Range):

$$IQR = Q3 - Q1$$

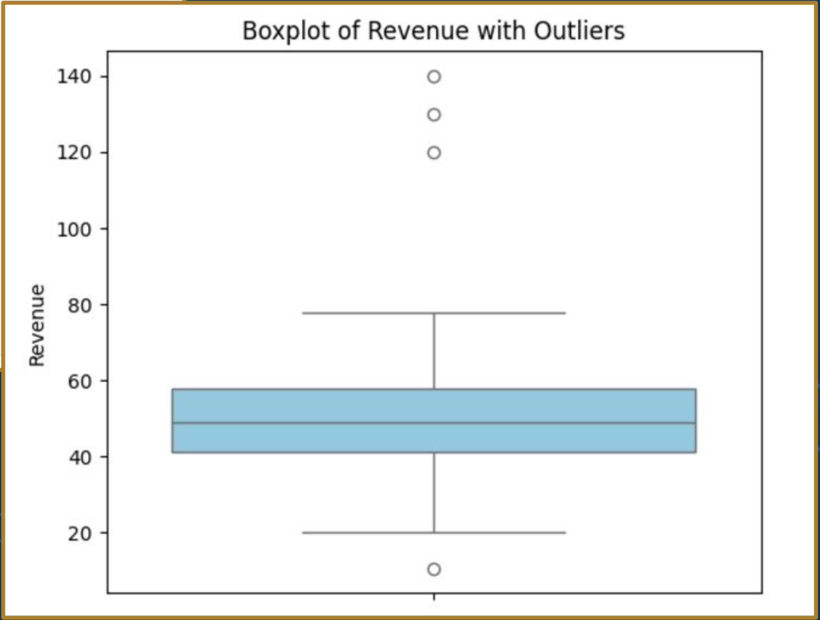
$$\text{Lower Bound} = Q1 - 1.5 \times IQR$$

$$\text{Upper Bound} = Q3 + 1.5 \times IQR$$

Let's visualize a dataset and identify outliers using Seaborn and Matplotlib.



```
[7]: import numpy as np
     import pandas as pd
     import seaborn as sns
     import matplotlib.pyplot as plt

     # Generate sample data with outliers
     np.random.seed(42)
     data = np.random.normal(loc=50, scale=15, size=100)  # Normal distribution
     data = np.append(data, [120, 130, 140])  # Add some outliers

     # Convert to DataFrame
     df = pd.DataFrame({'Revenue': data})

     # Create a Boxplot
     plt.figure(figsize=(6, 5))
     sns.boxplot(y=df['Revenue'], color='skyblue')

     # Add title and labels
     plt.title('Boxplot of Revenue with Outliers')
     plt.ylabel('Revenue')

     plt.show()
```



CoGrammar

What Do You See?

- ❖ The box represents the middle 50% of data (IQR).
- ❖ The horizontal line inside the box is the median (middle value).
- ❖ The whiskers extend to the smallest and largest values within 1.5 × IQR.
- ❖ The dots beyond the whiskers are outliers.

CoGrammar

How to Handle Outliers?

- ❖ Remove Outliers – If they result from errors or do not contribute meaningful insights.
- ❖ Transform Data – Use log transformation or normalization to reduce their effect.
- ❖ Cap Outliers – Replace extreme values with upper and lower bounds.
- ❖ Use Robust Models – Some models, like decision trees, are less sensitive to outliers.

CoGrammar

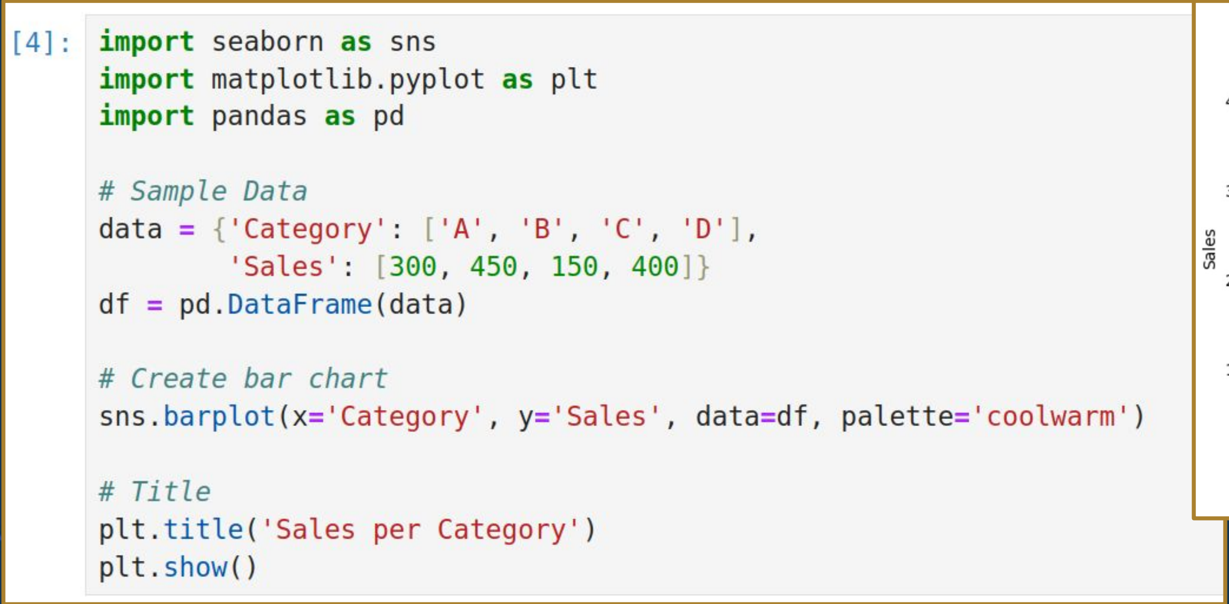
Advanced Visualisations

CoGrammar

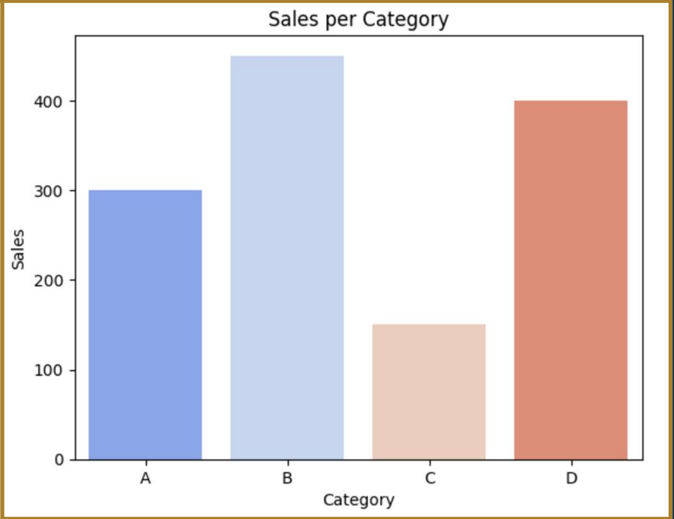
Let's take a break

CoGrammar

Using Seaborn for Advanced Visualizations



```
[4]: import seaborn as sns
     import matplotlib.pyplot as plt
     import pandas as pd

     # Sample Data
     data = {'Category': ['A', 'B', 'C', 'D'],
             'Sales': [300, 450, 150, 400]}
     df = pd.DataFrame(data)

     # Create bar chart
     sns.barplot(x='Category', y='Sales', data=df, palette='coolwarm')

     # Title
     plt.title('Sales per Category')
     plt.show()
```



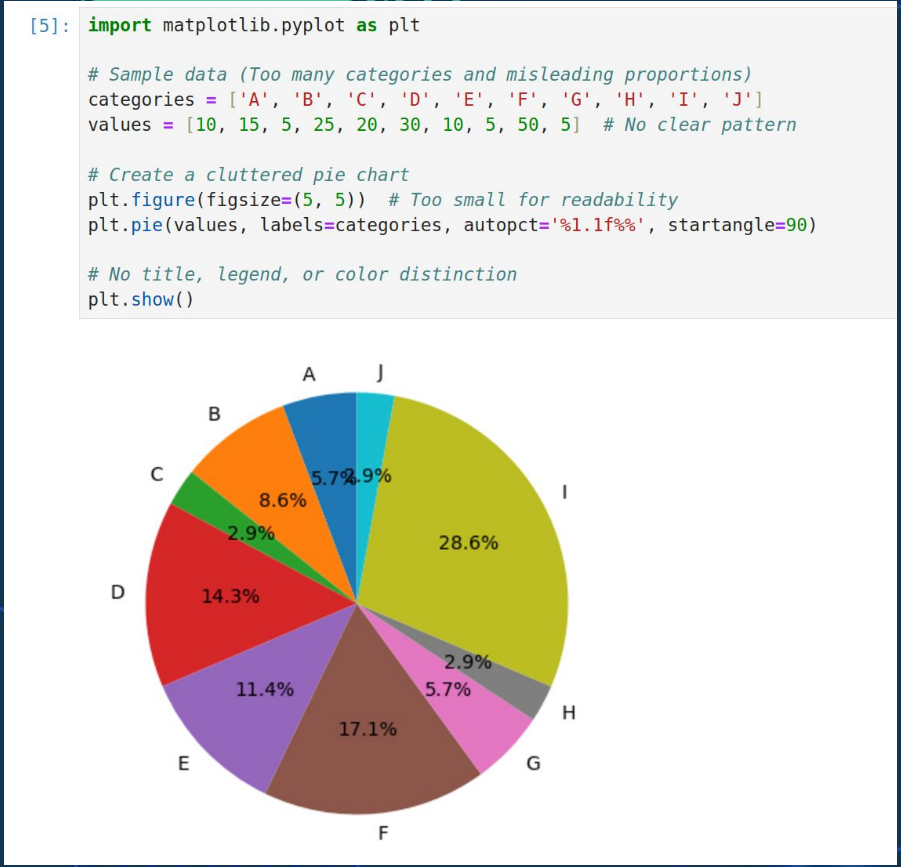
- When should we use bar charts instead of pie charts?

CoGrammar

📝 Exercise

CoGrammar

Good Visualisation?



```
[5]: import matplotlib.pyplot as plt

     # Sample data (Too many categories and misleading proportions)
     categories = ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J']
     values = [10, 15, 5, 25, 20, 30, 10, 5, 50, 5]  # No clear pattern

     # Create a cluttered pie chart
     plt.figure(figsize=(5, 5))  # Too small for readability
     plt.pie(values, labels=categories, autopct='%1.1f%%', startangle=90)

     # No title, legend, or color distinction
     plt.show()
```

Questions

- ❖ What's wrong with this chart?
- ❖ How can we improve it?

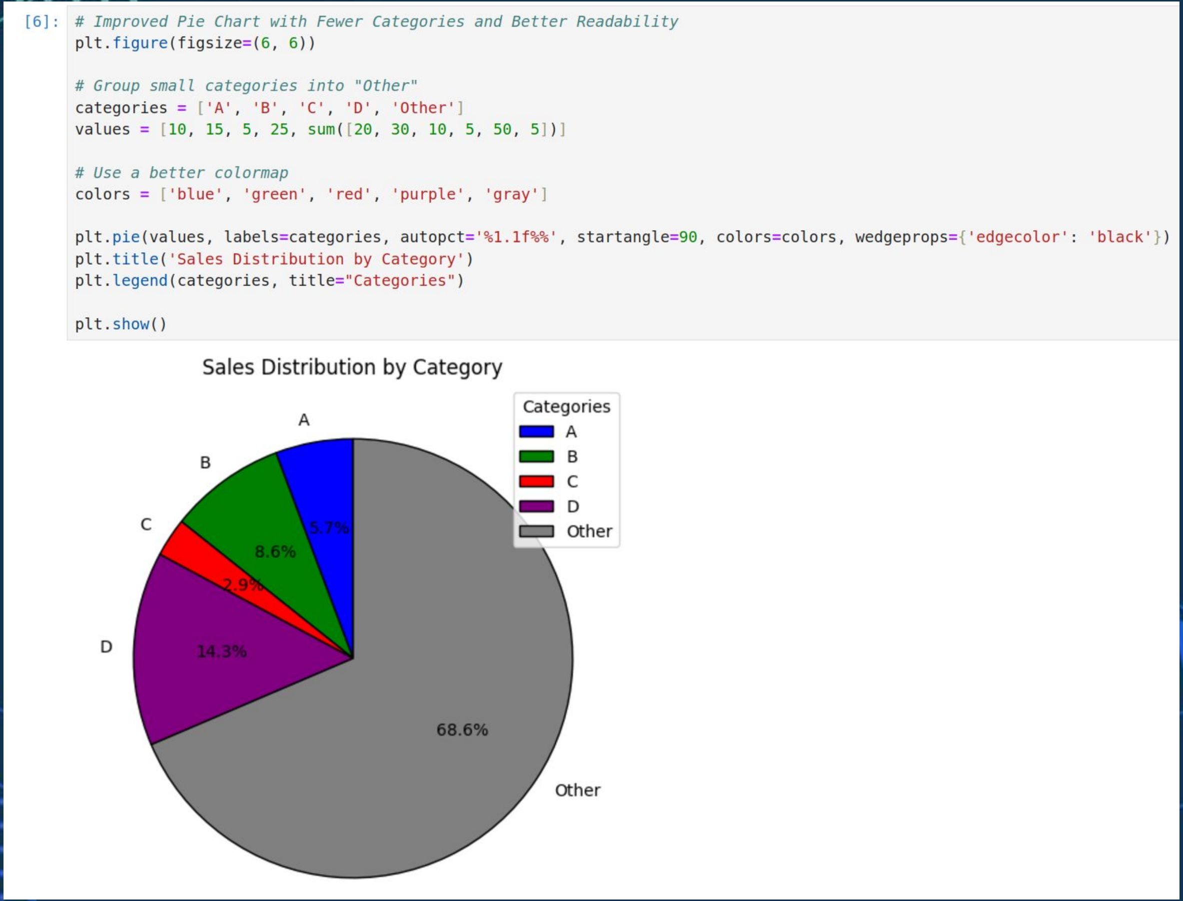
CoGrammar

Seaborn

- ❖ **What's wrong with this chart?**
  - ➢ 🚫 Too many categories, making it cluttered.
  - ➢ 🚫 Missing title and proper labels for clarity.
  - ➢ 🚫 Some sections are too small to be readable.
  - ➢ 🚫 The scale is misleading (percentages add confusion).
- ❖ **How can we improve it?**
  - ➢ ✅ Use a bar chart instead if categories are too many.
  - ➢ ✅ Add a title and proper labels.
  - ➢ ✅ Reduce the number of categories or group smaller ones under "Other."
  - ➢ ✅ Use distinct colors and a better layout.

CoGrammar

Improved Version



```
[6]: # Improved Pie Chart with Fewer Categories and Better Readability
     plt.figure(figsize=(6, 6))

     # Group small categories into "Other"
     categories = ['A', 'B', 'C', 'D', 'Other']
     values = [10, 15, 5, 25, sum([20, 30, 10, 5, 50, 5])]

     # Use a better colormap
     colors = ['blue', 'green', 'red', 'purple', 'gray']

     plt.pie(values, labels=categories, autopct='%1.1f%%', startangle=90, colors=colors, wedgeprops={'edgecolor': 'black'})
     plt.title('Sales Distribution by Category')
     plt.legend(categories, title="Categories")

     plt.show()
```

CoGrammar

Summary

- ❖ A bad chart makes data harder to understand.
- ❖ Simplify when needed (group small categories together).
- ❖ Choose the right chart type (Pie vs. Bar).
- ❖ Always add labels, a title, and proper formatting for clarity.

CoGrammar

Resources

- Matplotlib:
  https://matplotlib.org/stable/gallery/color/named_colors.html
  https://matplotlib.org/stable/api/pyplot_summary.html

- Seaborn examples:
  https://seaborn.pydata.org/examples/index.html

- If SSL error while getting Seaborn built-in data, download the data csv file from here - https://github.com/mwaskom/seaborn-data

- Resources to be installed for tutorial: **Python, NumPy, Pandas, Jupyter Notebook, Matplotlib, Seaborn**

CoGrammar

Questions and Answers

CoGrammar

Thank you for attending

CoGrammar

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education