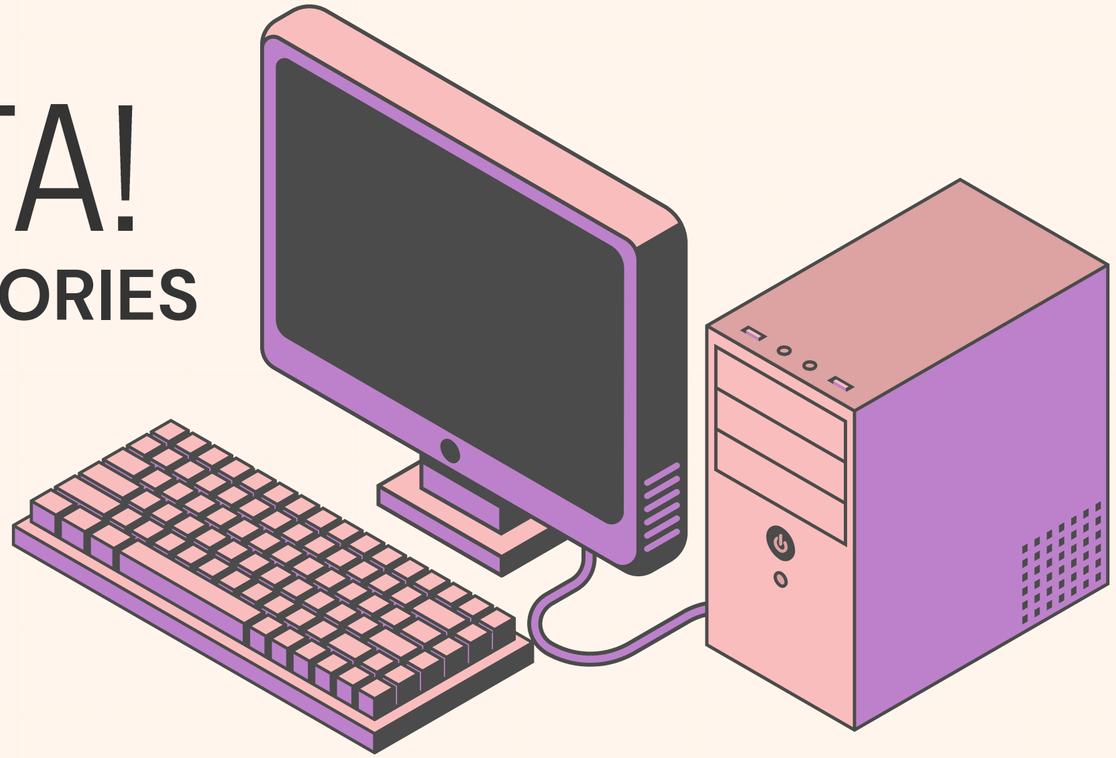
EVERYTHING DATA!

LET'S TALK PLOTS AND REPOSITORIES



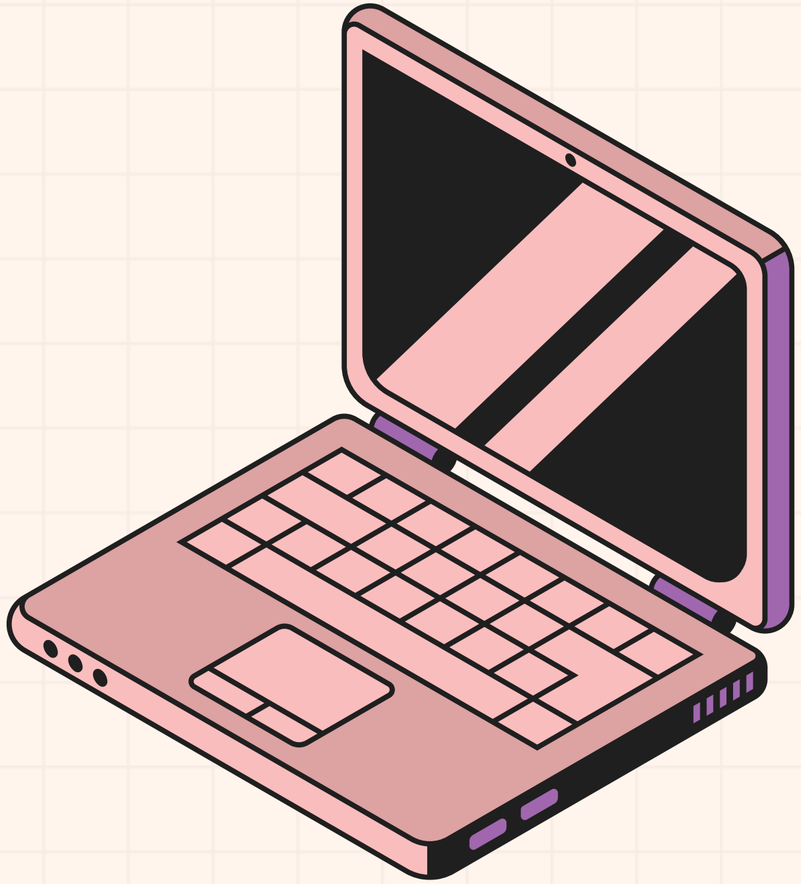
Today, the group will happily share a moment speaking summaries and in-depth descriptions of CTEC 128 and CTEC 298 tasks.

by: Aaronea Wiggins + Lojain Idris

INTRODUCTION

Hi everyone,

I’m Aaronea Wiggins and Lojain Idris.
We are both active participants within course CTEC 298 session 2.



SUMMARIES OF CTEC 128

PREVIOUSLY IN CTEC

I worked with a group of four and together we gathered data.

GATHERED DATA

what data you asked?
my group was able to retrieve stats directly from the FBI website in relation to the number of crimes that occurred in 2020, grouped by race.

OVERALL STUDY

our overall discovery was that Pacific Islanders and White Americans have committed the most crimes.

SUMMARIES OF CTEC 128

PREVIOUSLY IN CTEC

Top Data Breaches (2004-2021)

TOPIC

•Focuses on the the growing global threat of cyber-attacks to businesses

•Identifies common types of cyber-attacks.
•Analyzes the average number of attacks each year(2004-2021)

TOPIC

•Pivot table to show how different organization types were affected over time

•provide useful insights for developing better cybersecurity strategies.

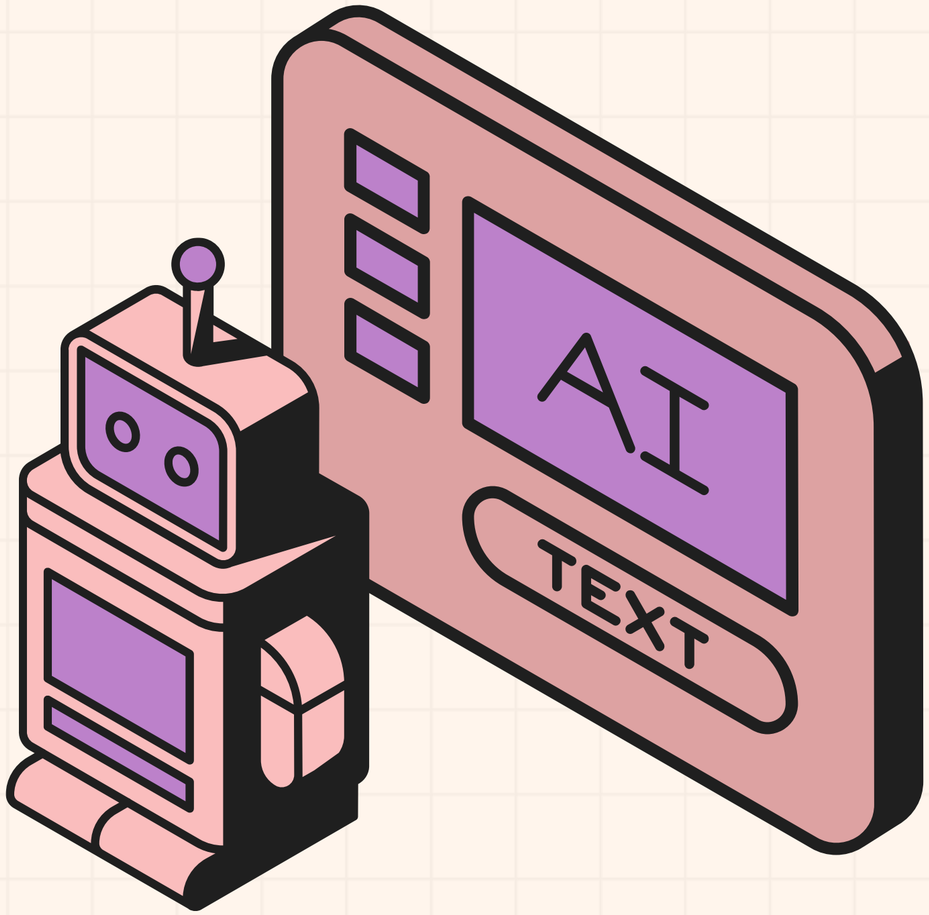
DESCRIPTION CTEC 298 MATERIAL

During my time within this course

- I have been introduced to many data related tutorials and tasks.
- CTEC 298 has exposed the class to data software and repositories such Matplotlib, Jupyter Notebook, Tableau, and GitHub.

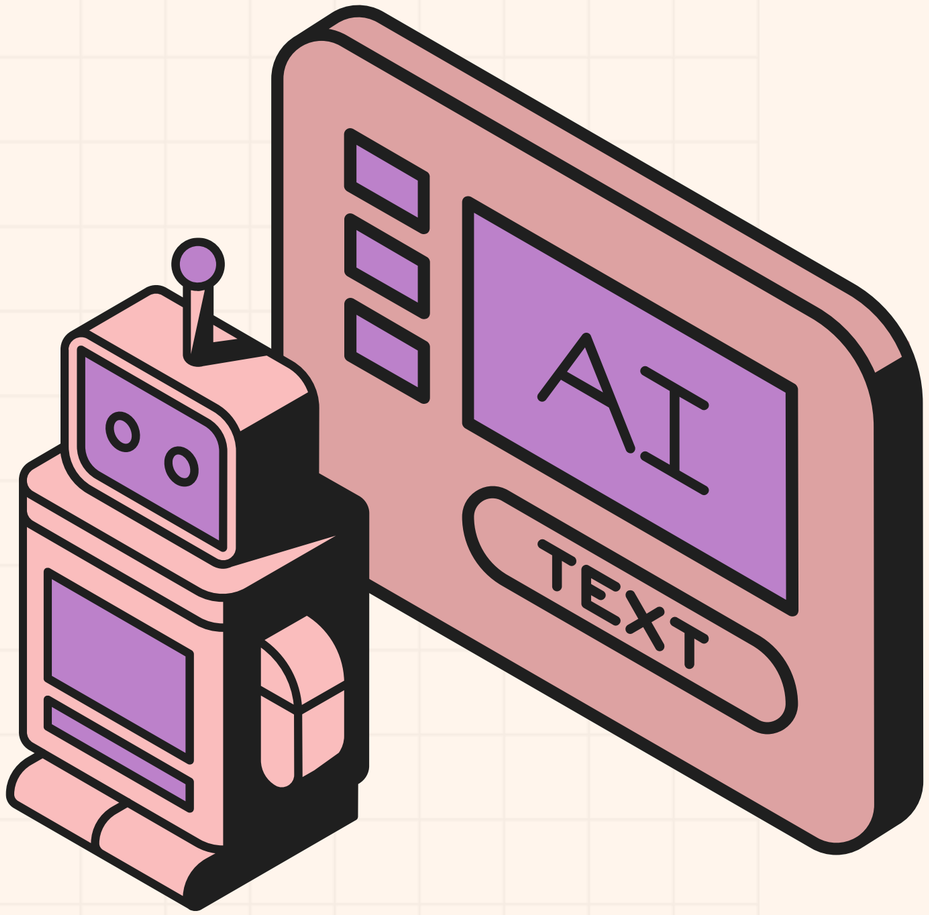
DEFINITIONS

- ·Jupyter Notebook is an interactive notebook which integrates code, visualizations, equations, and more. File extension .ipynb
- Matplotlib: A comprehensive library for creating static, animated, and interactive visualizations in Python
- Tableau: software for data input, visual analytical tool to help understand data
- GitHub: considerable repository to manage and share code.
- I have imported and uploaded several files within my repositories utilizing CTEC 128's data.



DESCRIPTION
CTEC 298 MATERIAL

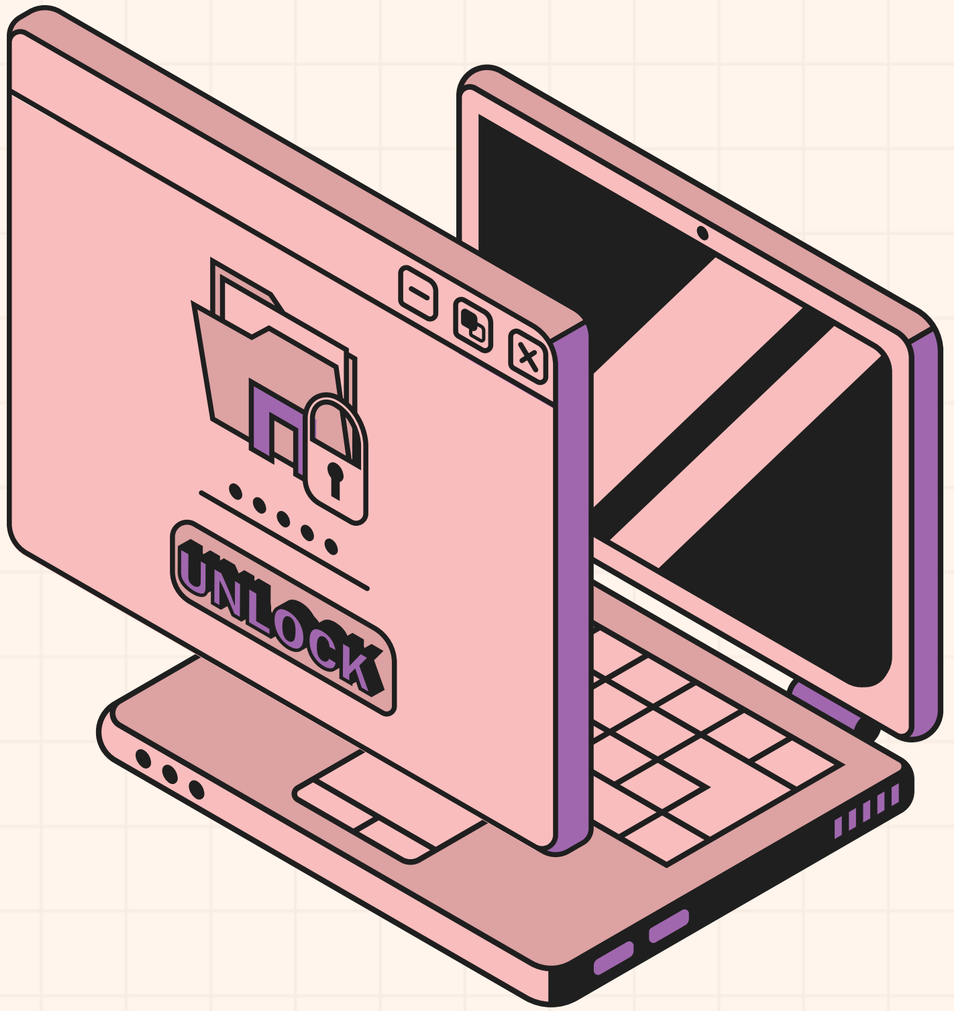
- LearnPython Tutorials Part One.
- LearnPython Tutorials Part Two
- Anaconda Installation
- GItHub Account Set Up
- Matplotlib Tutorials
- Matplotlib Python Plotting
- Pandas Tutorials
- Import numPy
- Tableau Tutorials
- Jupyter Notebook Python Code Completion



DESCRIPTION OF CTEC 298 PLOT(S)

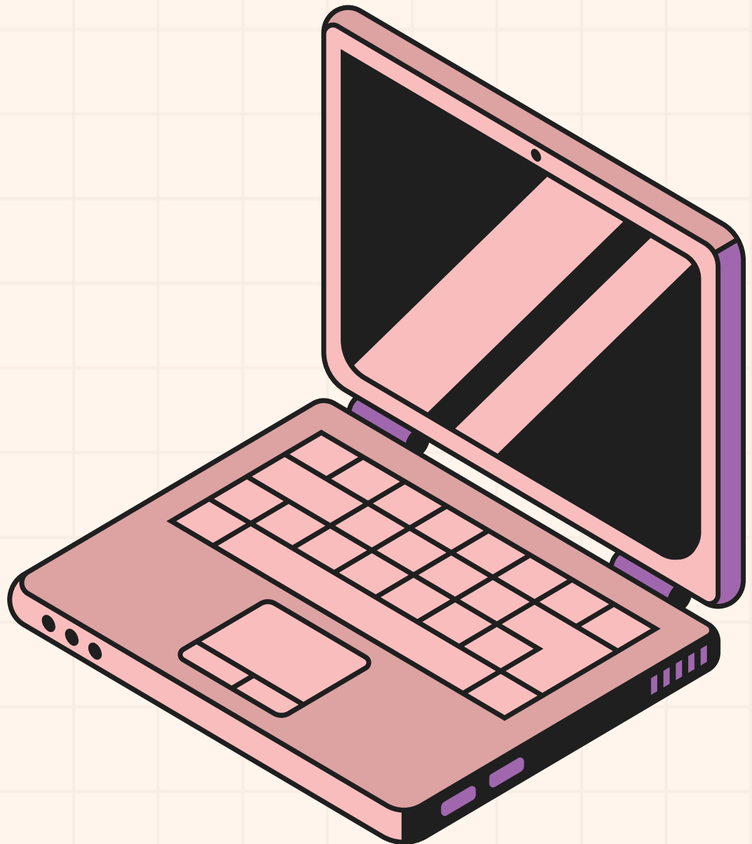
Of the six plots that we had to code, I'd easily say the bar graph was less tedious to complete within my Jupyter notebook.

During our individual plot assignment, I was assigned the Histogram that is a great plot for stats!



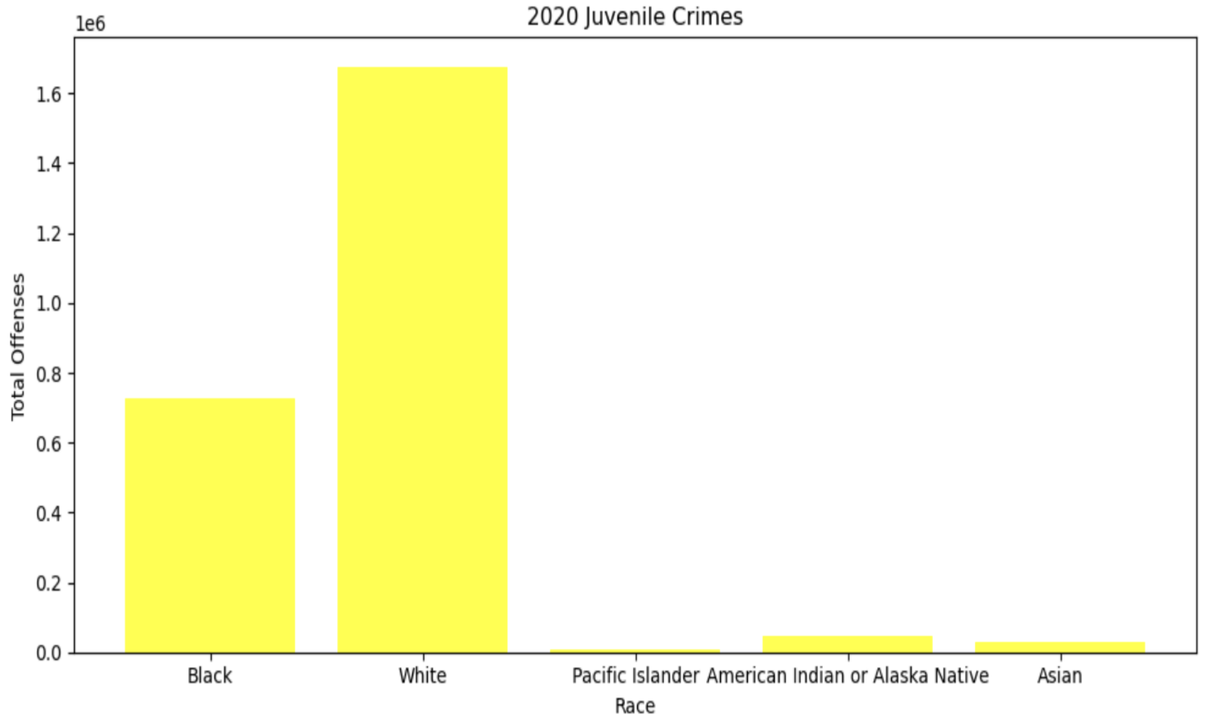
AW

VISUALS – JUPYTER NB OF CTEC 298 PLOT(S)



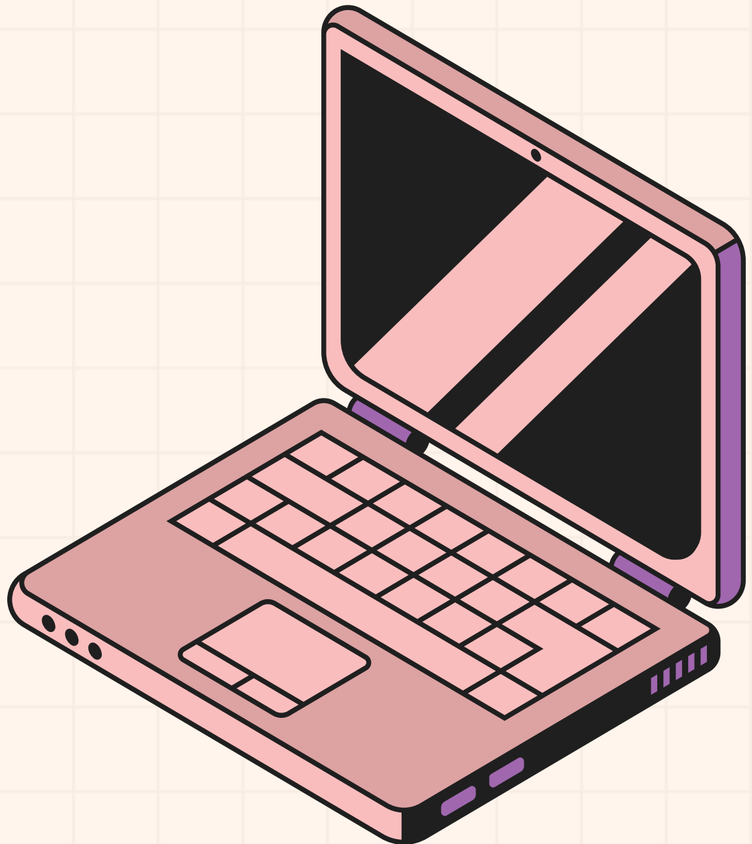
Bar Plot

[25]:
```
import pandas as pd
import matplotlib.pyplot as plt

# Load the excel file
df = pd.read_excel("Final CTEC 128 Stats.xlsx")

#dataset
df.columns = df.columns.str.strip()
df.rename(columns={'Unnamed: 0': 'Crime'}, inplace=True)

#Bar graph (plot)
totals = [df[race].sum() for race in races]
plt.figure(figsize=(10, 5))
plt.bar(races, totals, color='yellow')
plt.title('2020 Juvenile Crimes')
plt.xlabel('Race')
plt.ylabel('Total Offenses')
plt.tight_layout()
plt.show()
```



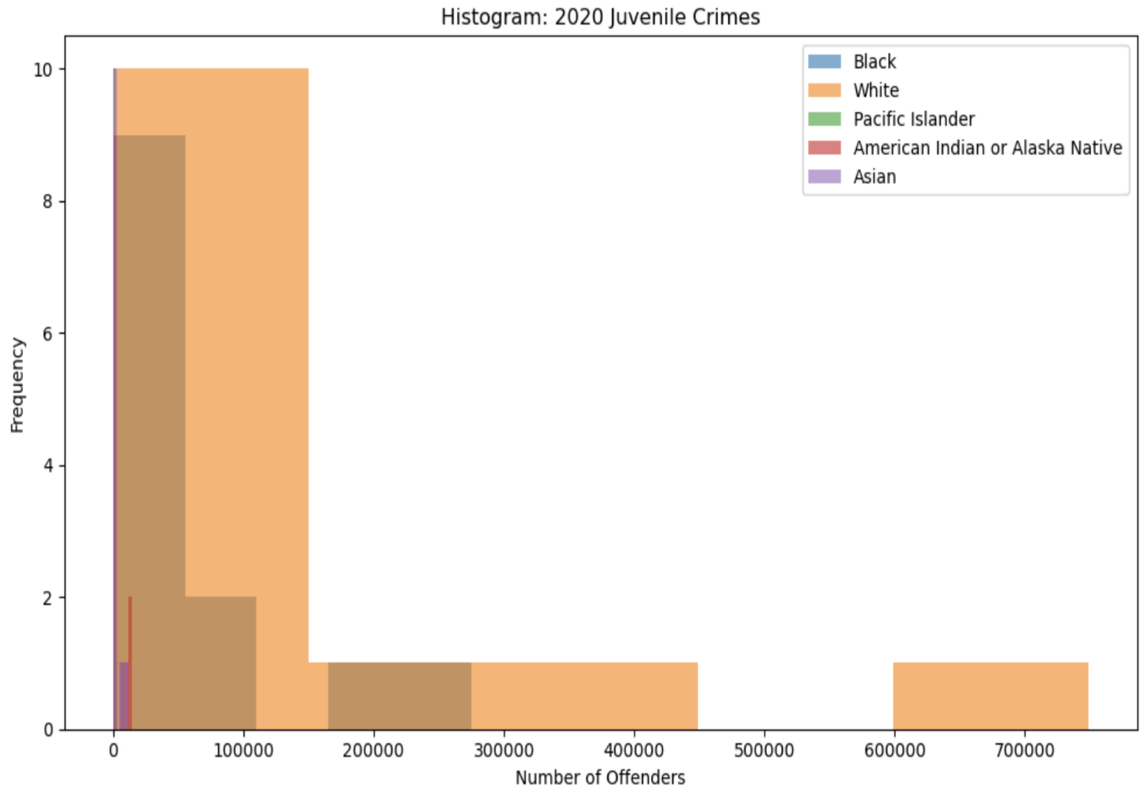
AW

VISUALS – JUPYTER NB OF CTEC 298 PLOT(S)



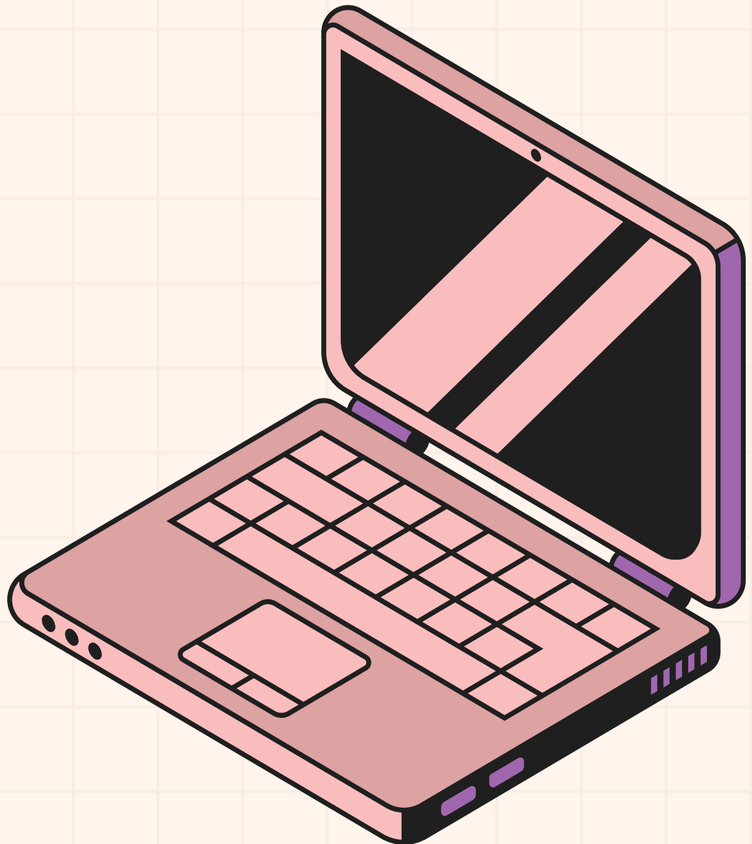
Histogram plot

```
[23]: import pandas as pd
      import matplotlib.pyplot as plt

      # Load the excel file
      df = pd.read_excel("Final CTEC 128 Stats.xlsx")

      #Dataset
      df.columns = df.columns.str.strip()
      df.rename(columns={'Unnamed: 0': 'Crime'}, inplace=True)

      races = ['Black', 'White', 'Pacific Islander', 'American Indian or Alaska Native', 'Asian']

      # Histogram: 2020 Juvenile Crimes
      plt.figure(figsize=(10, 6))
      for race in races:
          plt.hist(df[race], bins=5, alpha=0.6, label=race)

      plt.title('Histogram: 2020 Juvenile Crimes')
      plt.xlabel('Number of Offenders')
      plt.ylabel('Frequency')
      plt.legend()
      plt.tight_layout()
      plt.show()
```



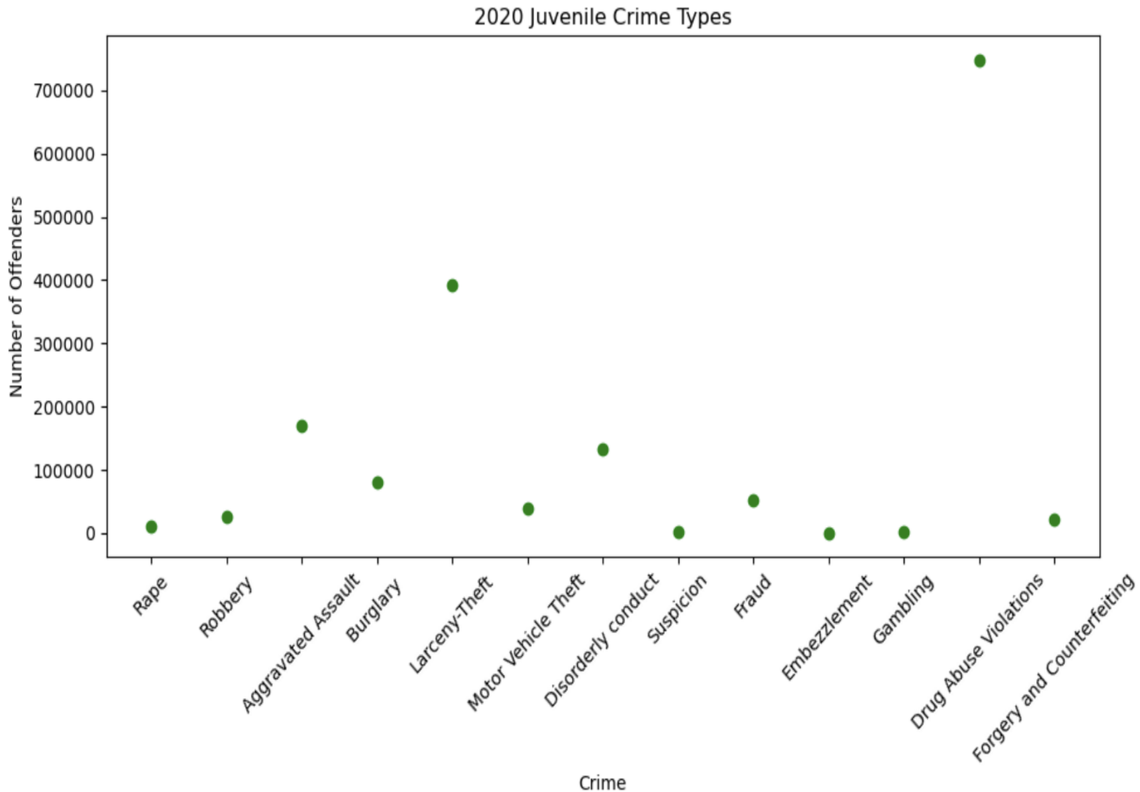
AW

VISUALS – JUPYTER NB OF CTEC 298 PLOT(S)



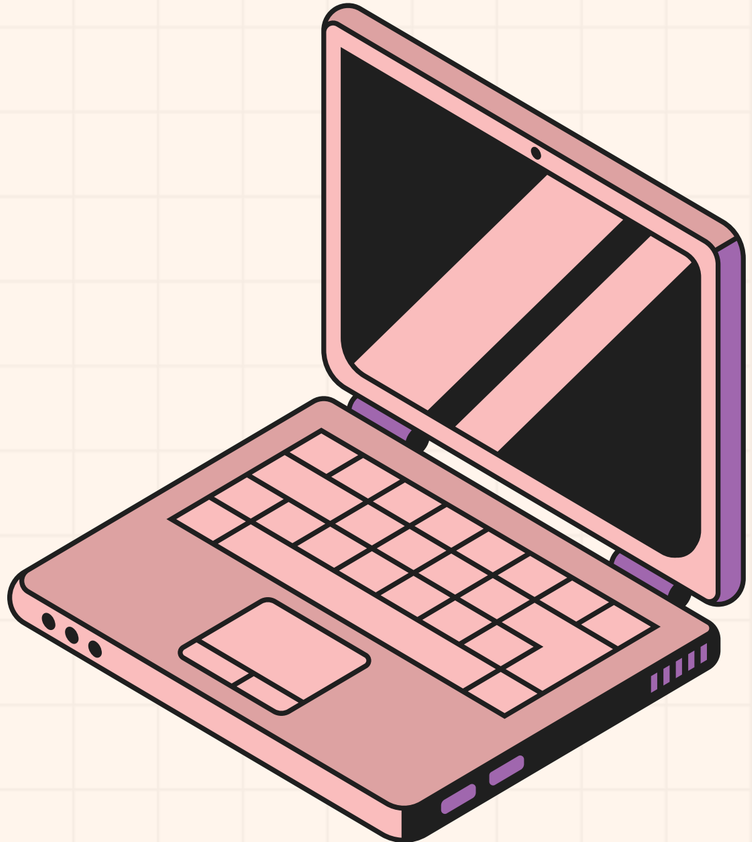
Scatter plot

```
[18]: import pandas as pd
      import matplotlib.pyplot as plt

      # Load the excel file
      df = pd.read_excel("Final CTEC 128 Stats.xlsx")

      #Dataset
      df.columns = df.columns.str.strip()
      df.rename(columns={'Unnamed: 0': 'Crime'}, inplace=True)

      races = ['Black', 'White', 'Pacific Islander', 'American Indian or Alaska Native', 'Asian']


      plt.figure(figsize=(10, 6))
      plt.scatter(df['Crime'], df['White'], color='green')
      plt.title('2020 Juvenile Crime Types')
      plt.xlabel('Crime')
      plt.ylabel('Number of Offenders')
      plt.xticks(rotation=45)
      plt.tight_layout()
      plt.show()
```

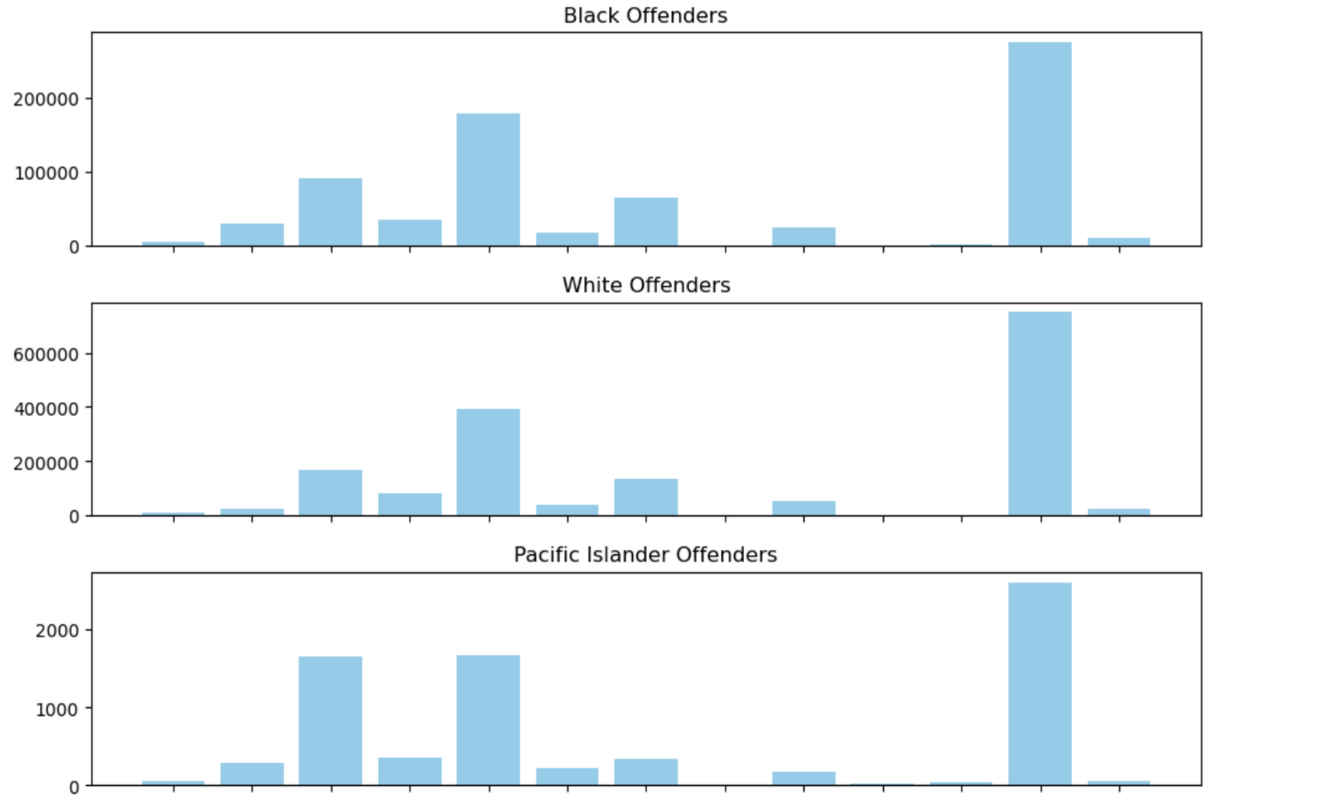


AW

VISUALS – JUPYTER NB OF CTEC 298 PLOT(S)



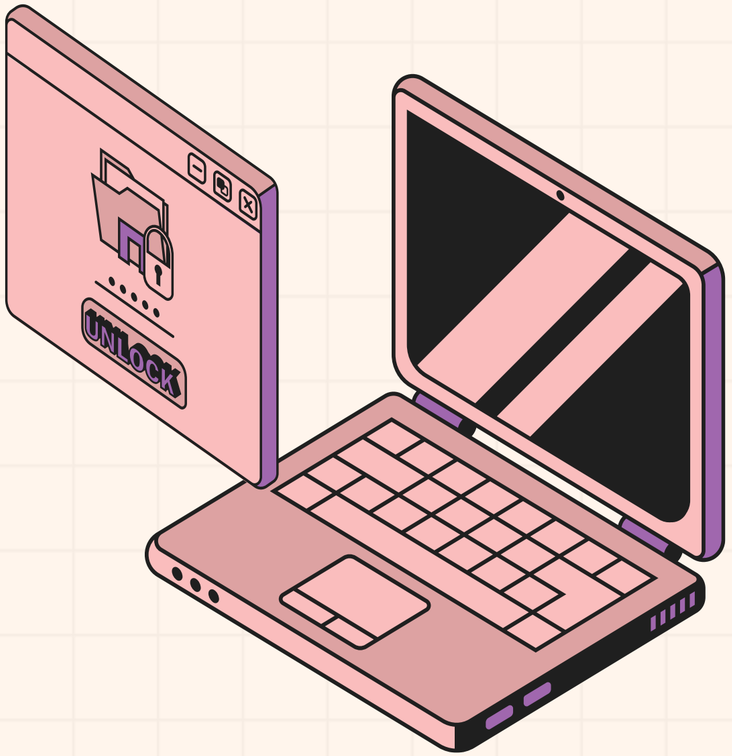
```
# Load the excel file
df = pd.read_excel("Final CTEC 128 Stats.xlsx")

#Dataset
df.columns = df.columns.str.strip()
df.rename(columns={'Unnamed: 0': 'Crime'}, inplace=True)

races = ['Black', 'White', 'Pacific Islander', 'American Indian or Alaska Native', 'Asian']

#Chart reqs
fig, axs = plt.subplots(len(races), 1, figsize=(10, 12), sharex=True)
for i, race in enumerate(races):
    axs[i].bar(df['Crime'], df[race], color='skyblue')
    axs[i].set_title(f'{race} Offenders')
    axs[i].tick_params(axis='x', rotation=45)
plt.tight_layout()
plt.show()
```



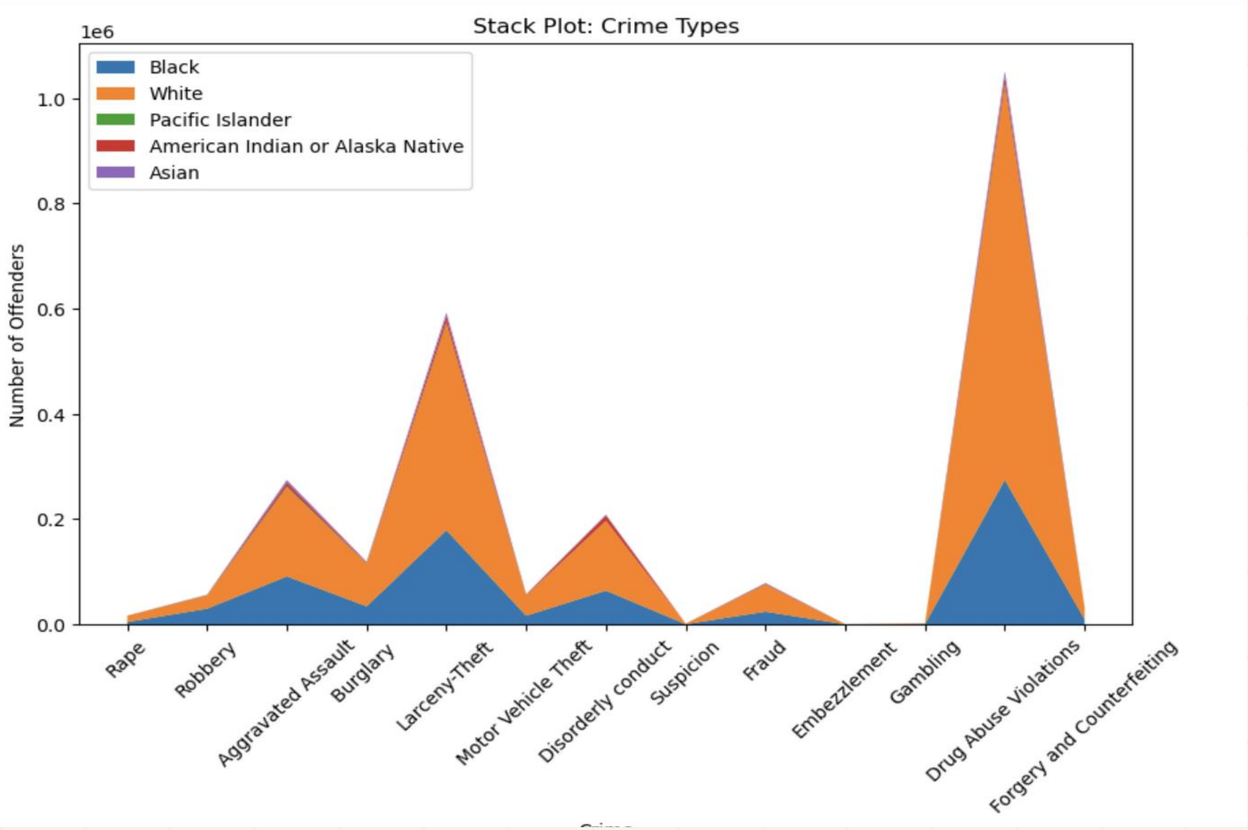
Multi-Plot

AW

VISUALS – JUPYTER NB OF CTEC 298 PLOT(S)



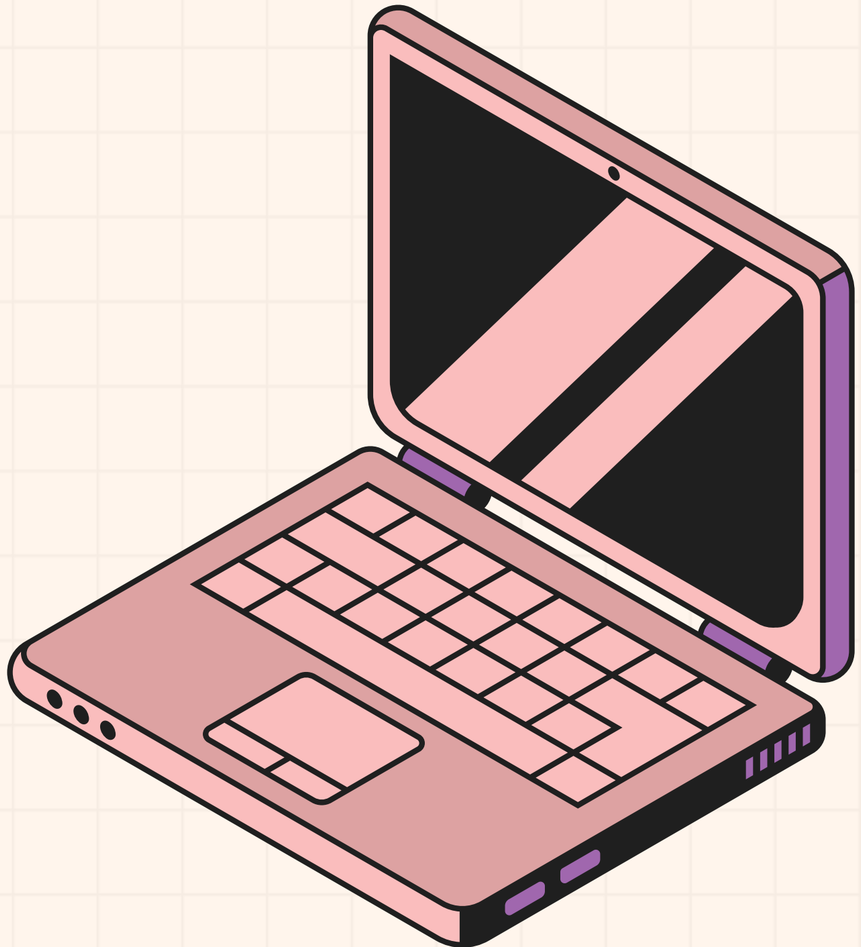

```
7]: import pandas as pd
    import matplotlib.pyplot as plt

    # Load the excel file
    df = pd.read_excel("Final CTEC 128 Stats.xlsx")

    # Clean and prepare data
    df.columns = df.columns.str.strip()
    df.rename(columns={'Unnamed: 0': 'Crime'}, inplace=True)

    races = ['Black', 'White', 'Pacific Islander', 'American Indian or Alaska Native', 'Asian']
    plt.figure(figsize=(10, 6))
    plt.stackplot(df['Crime'], [df[race] for race in races], labels=races)
    plt.title('Stack Plot: Crime Types')
    plt.xlabel('Crime')
    plt.ylabel('Number of Offenders')
    plt.legend(loc='upper left')
    plt.xticks(rotation=45)
    plt.show()
```



Stacked Plot

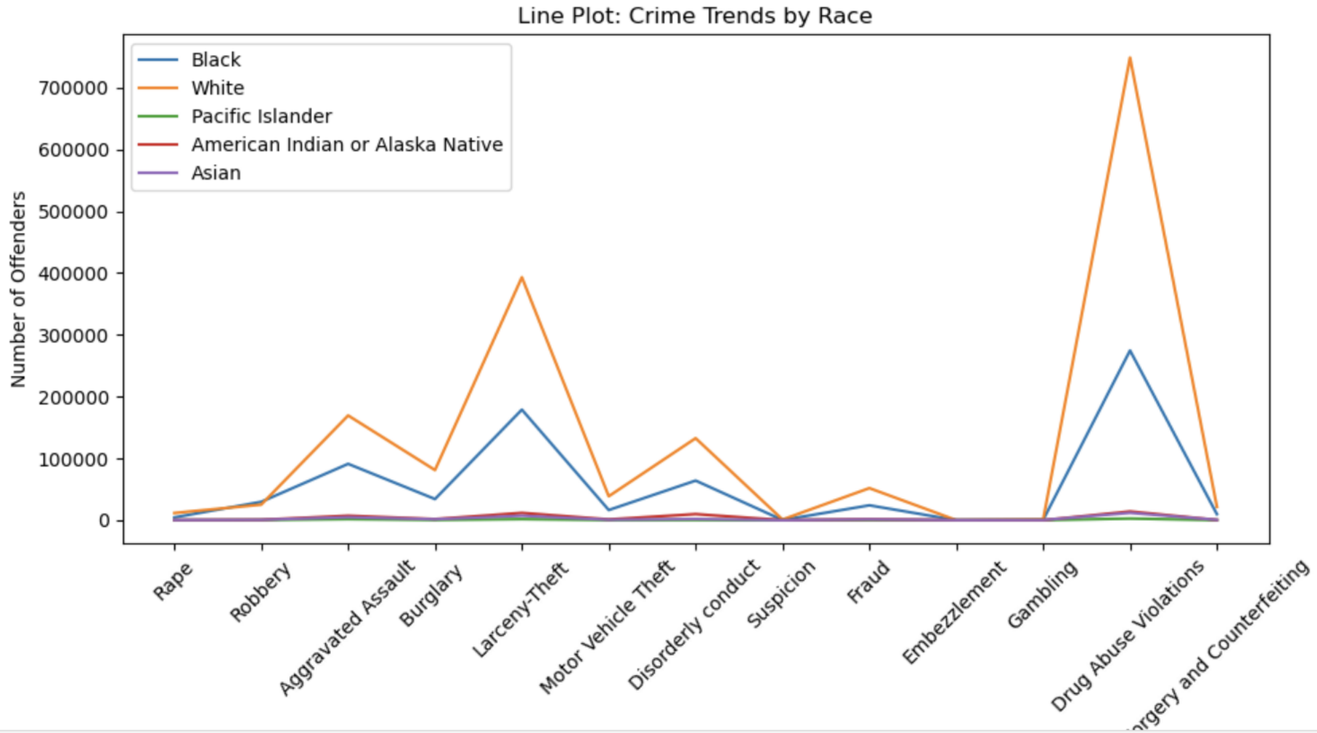
AW

VISUALS – JUPYTER NB OF CTEC 298 PLOT(S)



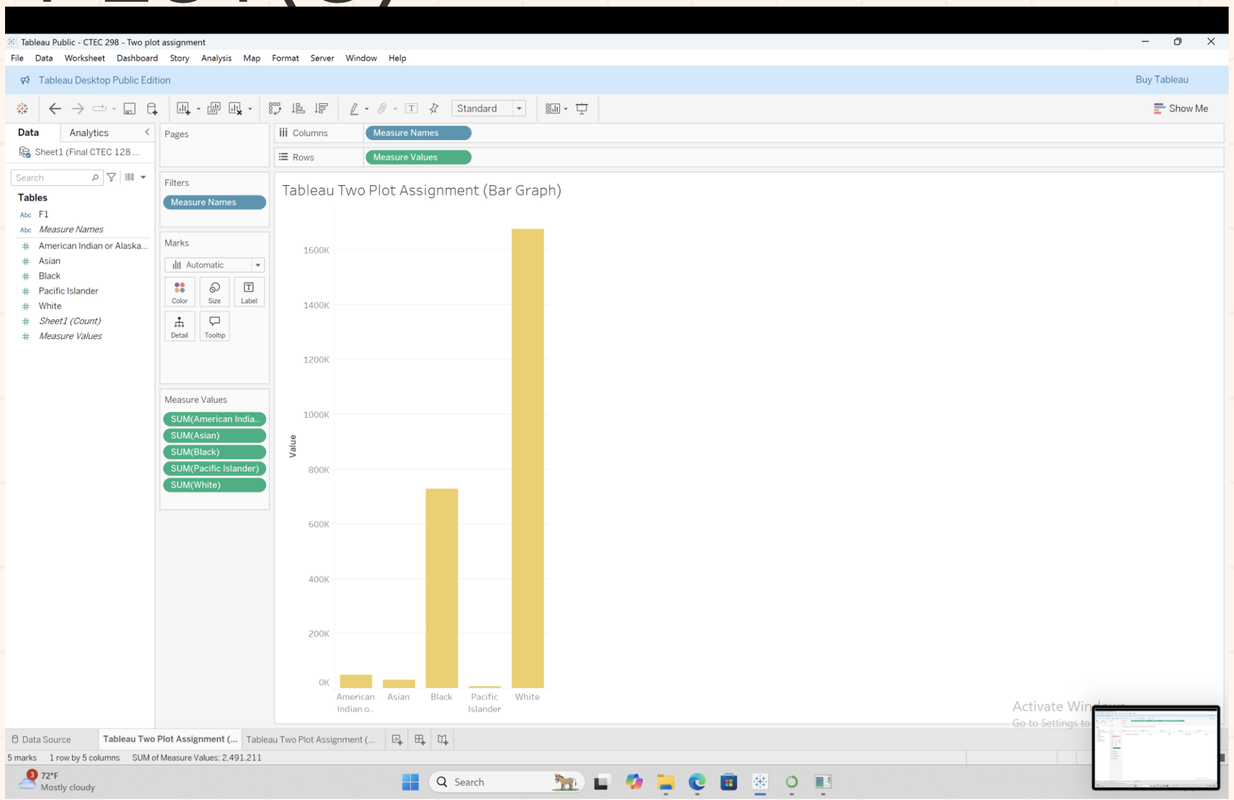
Line Plot

```
[19]: import pandas as pd
      import matplotlib.pyplot as plt

      # Load the excel file
      df = pd.read_excel("Final CTEC 128 Stats.xlsx")

      # Dataset
      df.columns = df.columns.str.strip()
      df.rename(columns={'Unnamed: 0': 'Crime'}, inplace=True)

      races = ['Black', 'White', 'Pacific Islander', 'American Indian or Alaska Native', 'Asian']

      plt.figure(figsize=(10, 6))
      for race in races:
          plt.plot(df['Crime'], df[race], label=race)
      plt.title('Line Plot: Crime Trends by Race')
      plt.xlabel('Crime')
      plt.ylabel('Number of Offenders')
      plt.xticks(rotation=45)
      plt.legend()
      plt.tight_layout()
      plt.show()
```

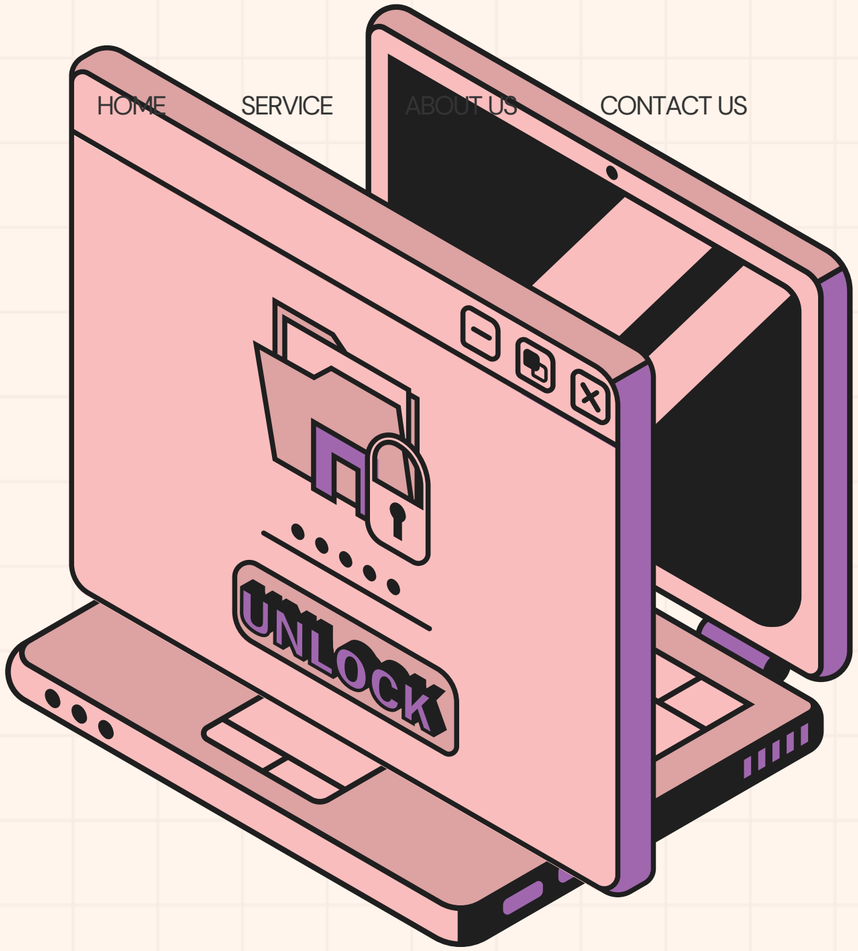


AW

VISUALS – TABLEAU PLOT(S)

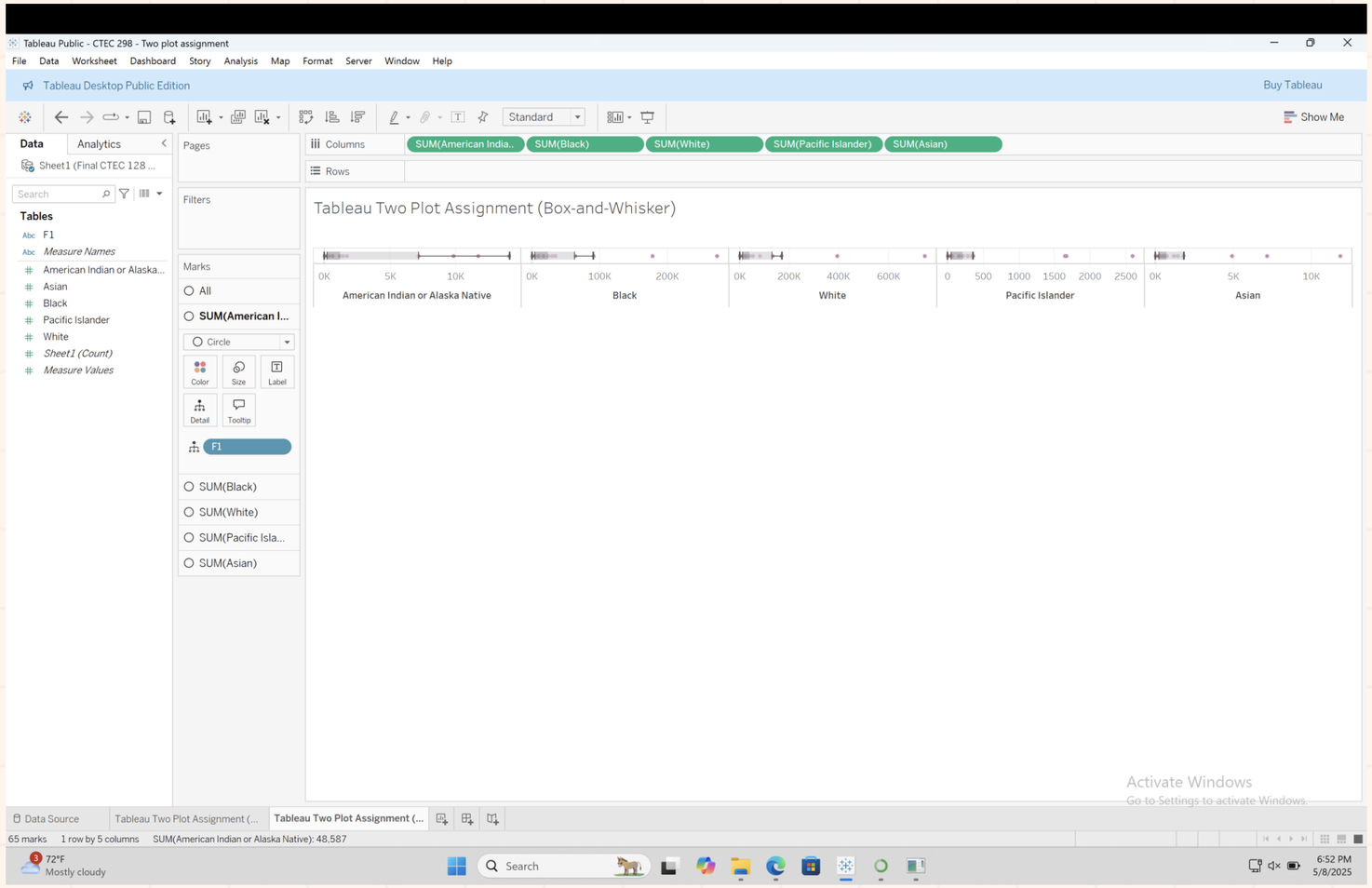


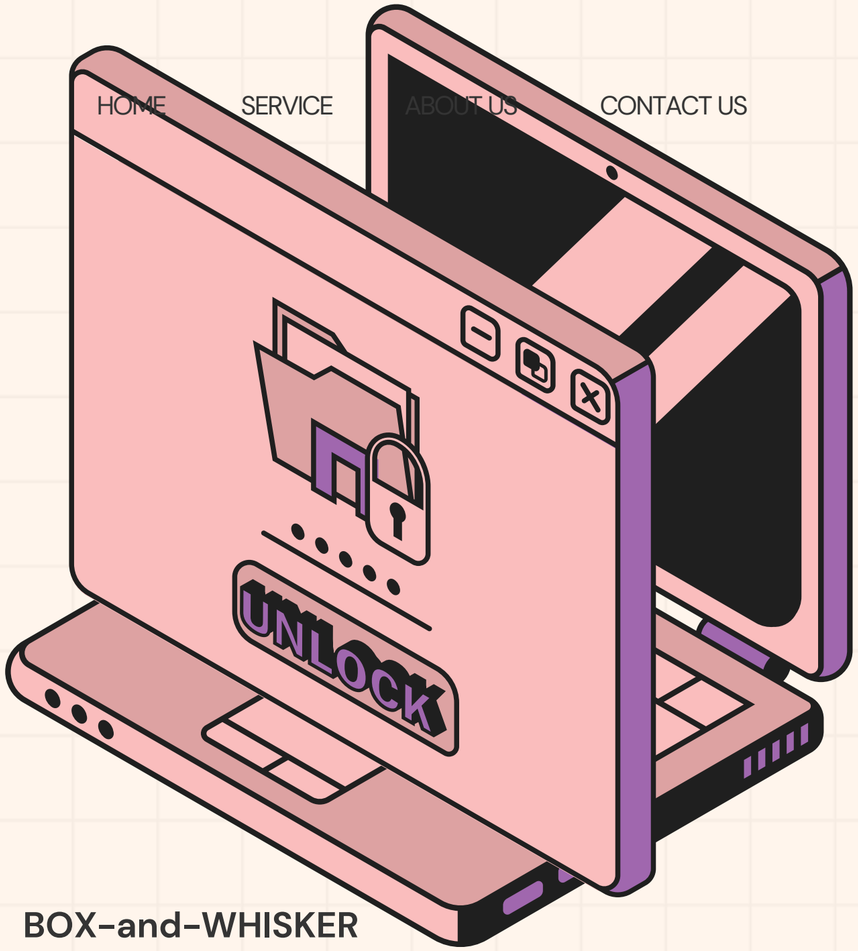
Bar Plot



AW

VISUALS – TABLEAU PLOT(S)





BOX-and-WHISKER

DESCRIPTION OF CTEC 298 PLOT(S)

Matplotlib Pie Chart
- Shows top 5 breached organization types
- Each slice shows % of total breaches

Matplotlib Histogram
- Shows how many records were breached in General
- Bars are orange with black edges

Matplotlib Scatter Plot
- Shows breaches from 2004 to 2021
- Each yellow dot shows one breach

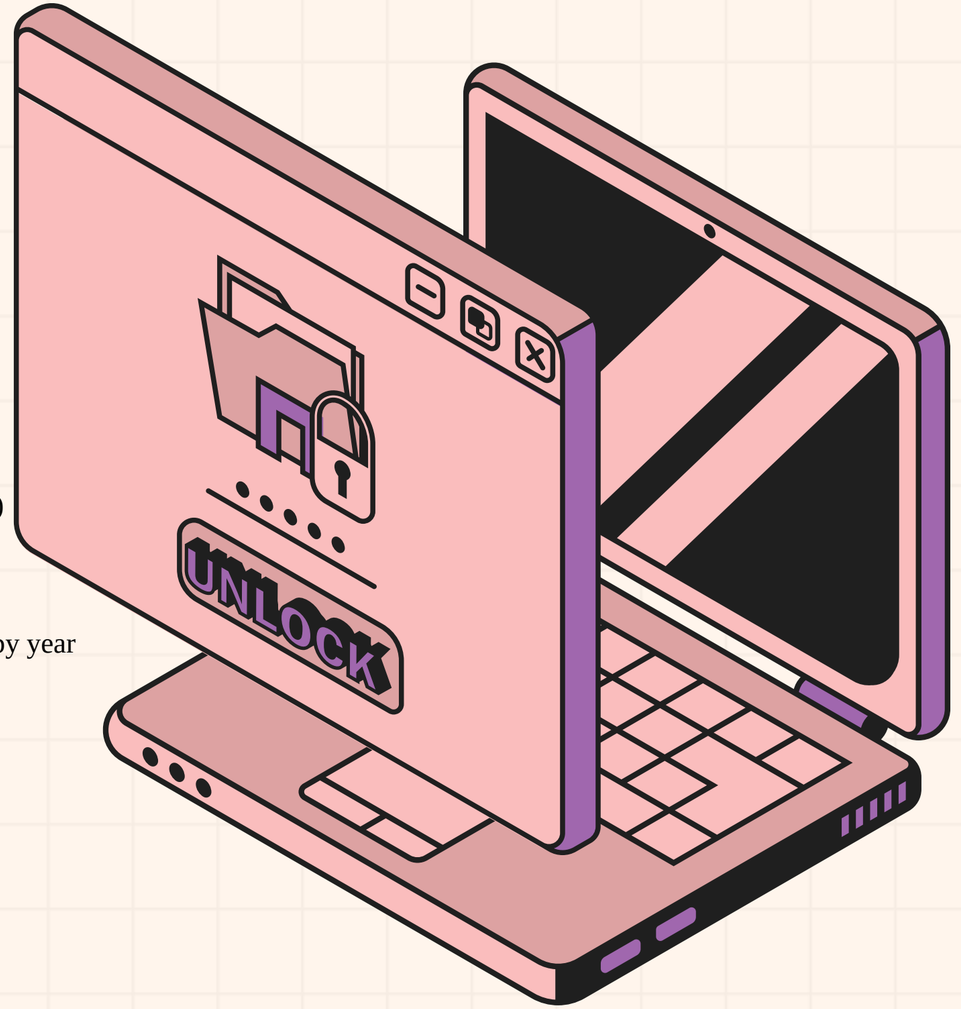
Matplotlib Bar Graph
- Shows top 10 companies or groups with biggest breaches
- Red bars show number of records breached

Matplotlib Area Plot
- Shows total breaches each year (2004–2021)
- Shows how breaches increased or decreased

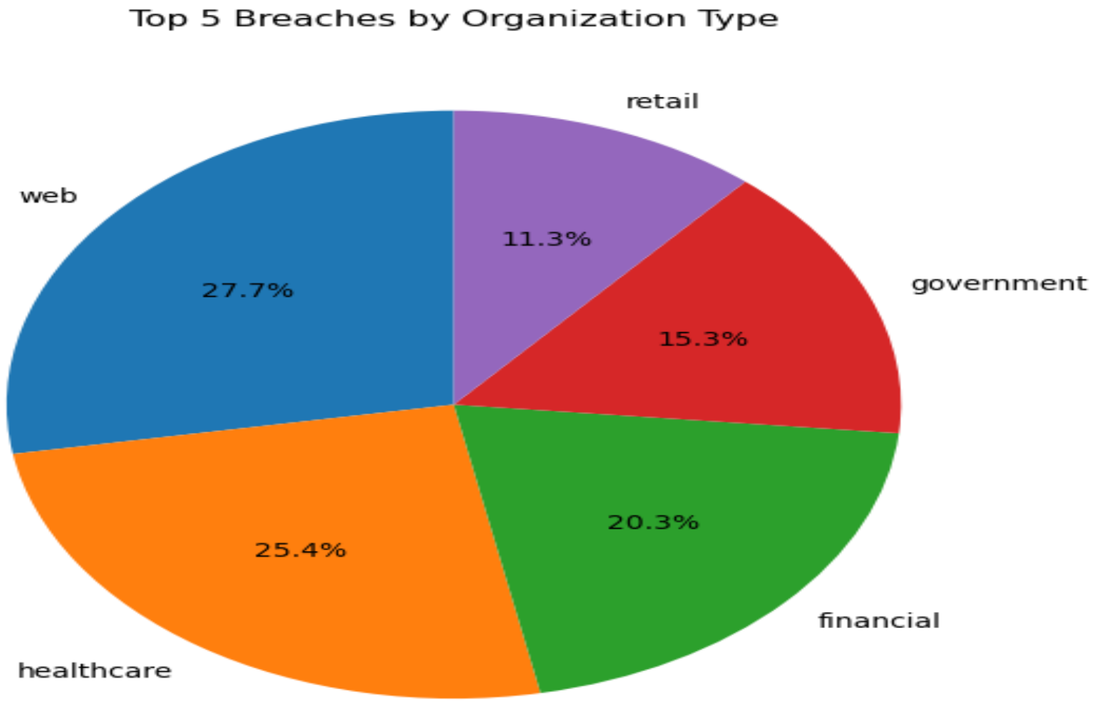
Matplotlib Hexagonal Bin Plot
- Shows where breaches were most common by year and record count
- Fixed data by changing years to numbers

Tableau Packed Bubble Chart
- Shows breach methods using bubbles
- Bigger bubbles = more records affected

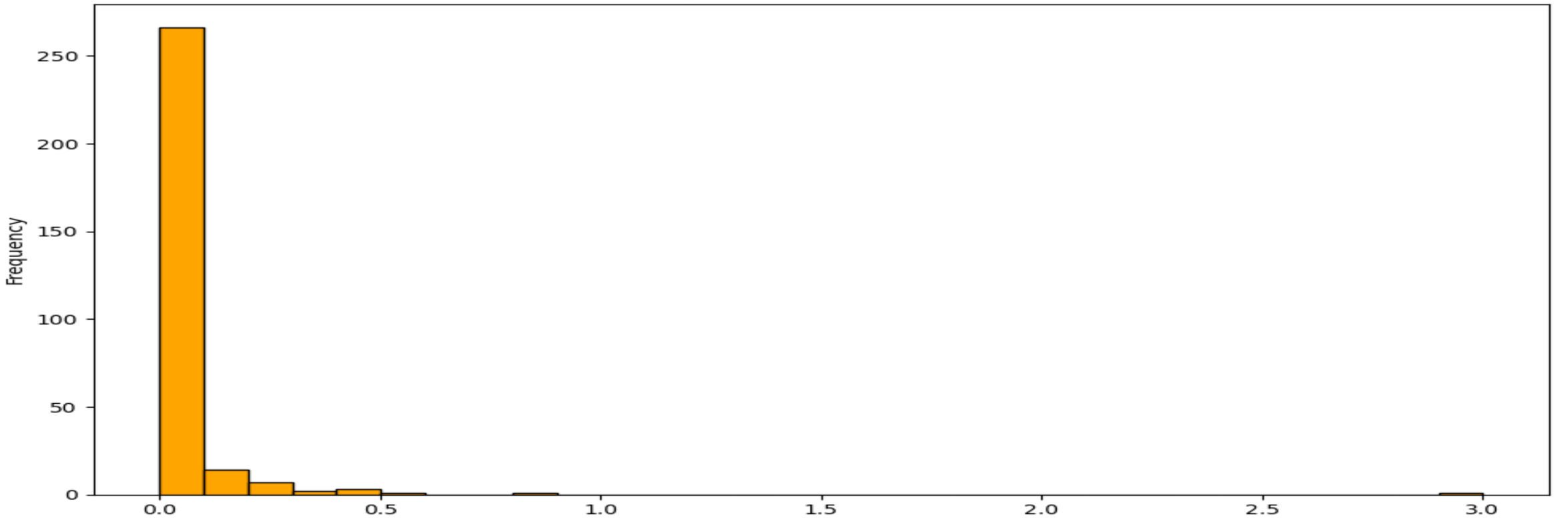
Tableau Horizontal Bar Chart
- Shows total records breached by method
- Longer bars = more records lost



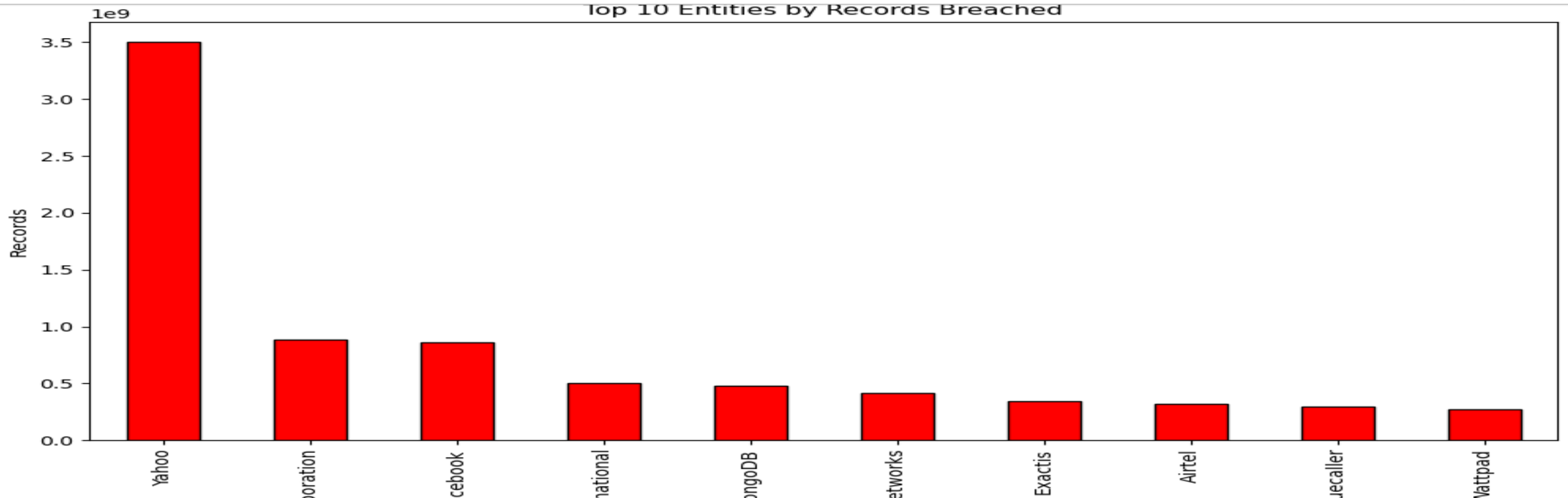
[8]:
```
#creating fig and axes
fig, axes = plt.subplots(figsize=(6, 6))

#creating or plotting the Pie chart
org_counts = df['Organization type'].value_counts().nlargest(5)# top five
axes.pie(org_counts, labels=org_counts.index, autopct='%1.1f%%', startangle=90)
axes.set_title("Top 5 Breaches by Organization Type")
plt.tight_layout()
plt.show()
```



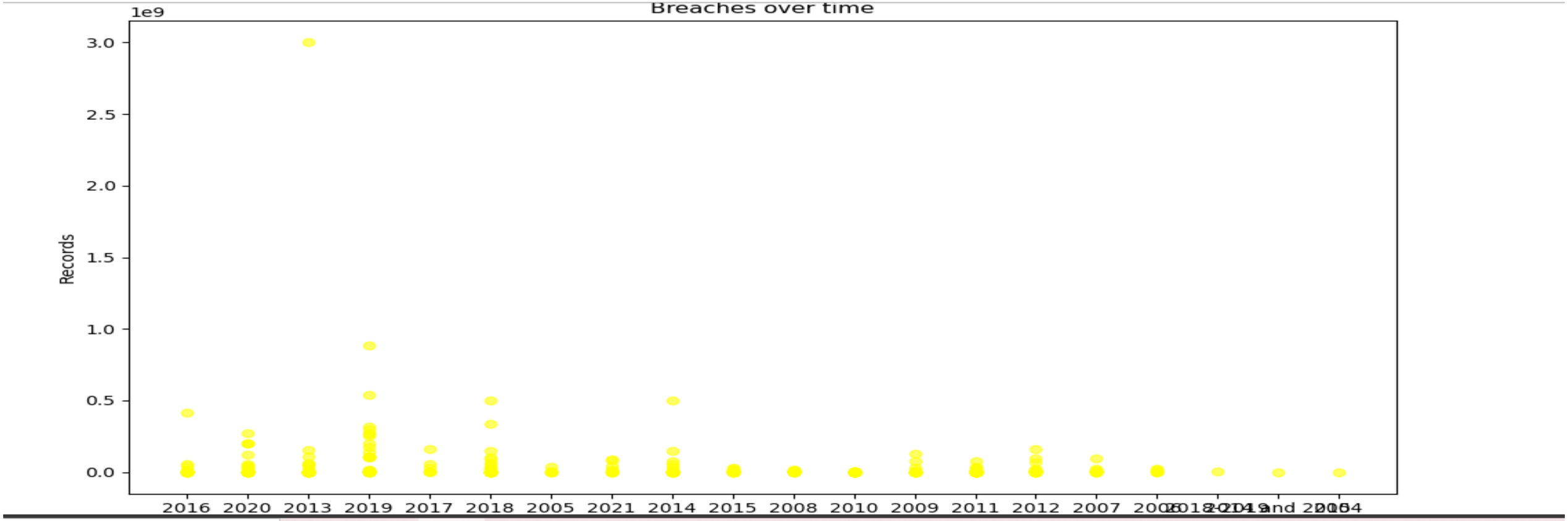
[15]:
```
#histograms plot
plt.figure(figsize=(10,6))
plt.hist(df['Records'], bins=30, color='orange', edgecolor='black')
#naming the lablels
plt.title("Histogram of Records Breached")
plt.xlabel("Records")
plt.ylabel("Frequency")
#displaying
plt.tight_layout()
plt.show
```



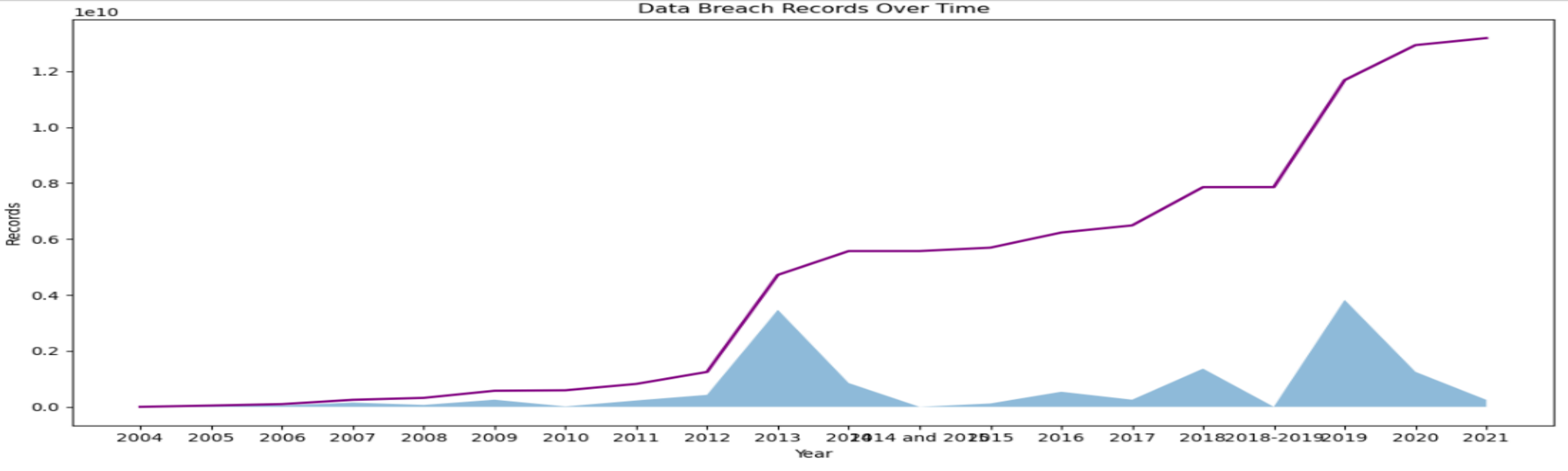
[18]:
```
#bar graph
top_entities = df.groupby('Entity')['Records'].sum().sort_values(ascending=False).head(10)
plt.figure(figsize=(12,6))
top_entities.plot(kind='bar', color='red', edgecolor='black')
#naming the labels
plt.title("Top 10 Entities by Records Breached")
plt.xlabel("entity")
plt.ylabel("Records")
plt.xtick(rotation=45, ha='right')#rotate x-axis and aligning the lebels to right

plt.tigh_layout()
plt.show()
```



[21]:
```
#scatter plot
plt.figure(figsize=(10, 6))
plt.scatter(df['Year'], df['Records'], color='yellow', alpha=0.6)

#naming the lebels
plt.title("Breaches over time")
plt.xlabel("Year")
plt.ylabel("Records")
#diplay
plt.tight_layout()
plt.show()
```



[24]:
```
#area plot
#calculate cumulative breach size over time and total records
records_per_year = df.groupby('Year')['Records'].sum().sort_index()
cumulative_records = records_per_year.cumsum()
#plotting
plt.figure(figsize=(12,6))
plt.fill_between(records_per_year.index, records_per_year.values, alpha=0.5, label='Total Records (per year)')
plt.plot(cumulative_records.index, cumulative_records.values, color='purple', label='Cumulative Records', linewidth=2)

# naming Labels
plt.title("Data Breach Records Over Time")
plt.xlabel("Year")
plt.ylabel("Records")
# Display the plot
plt.tight_layout()
plt.show()
```



36]:
```
#hexbin plot / hexagonal Bin
#plotting
#convert any string in years to numeric value
df['Year'] = pd.to_numeric(df['Year'], errors='coerce')
plt.figure(figsize=(10, 6))
plt.hexbin(df['Year'], df['Records'], gridsize=30, cmap='Blues', bins='log')
# naming labels
plt.title("Breaches Records over time")
plt.xlabel("year")
plt.ylabel("Records")

#display
plt.tight_layout()
plt.show()
```

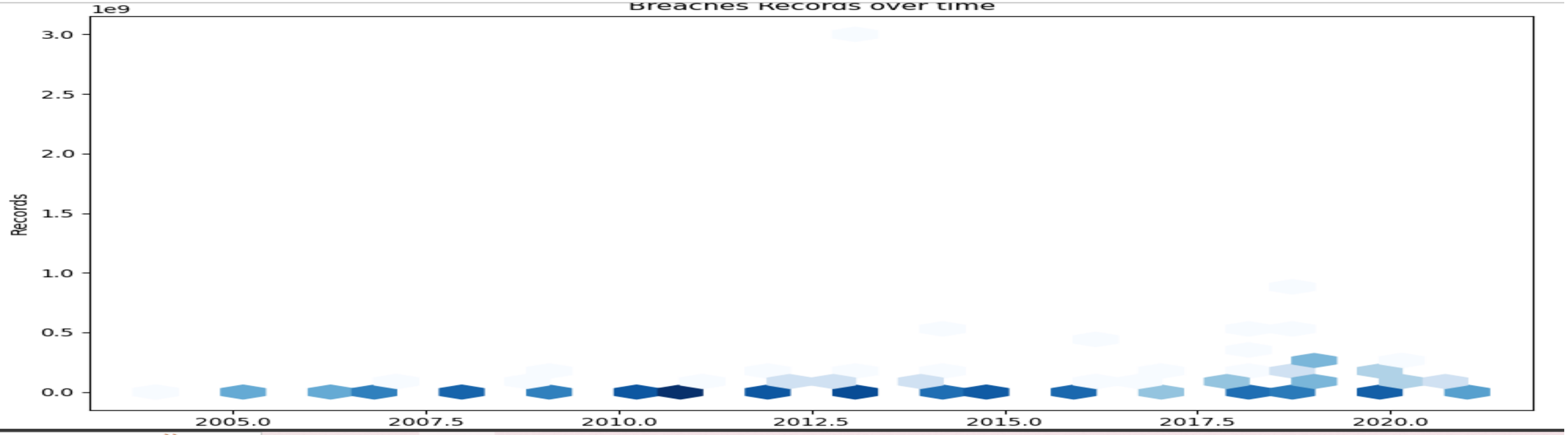
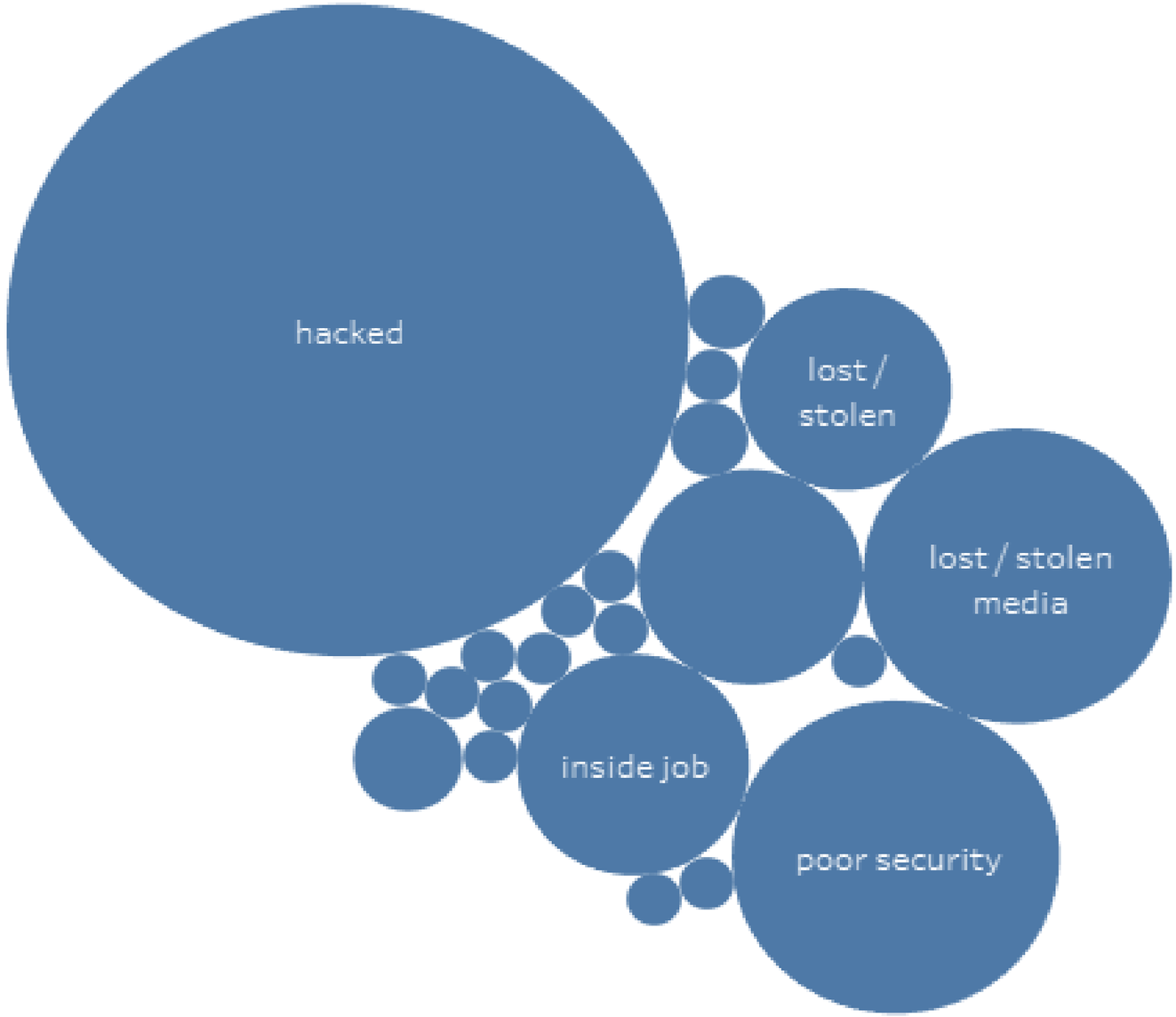
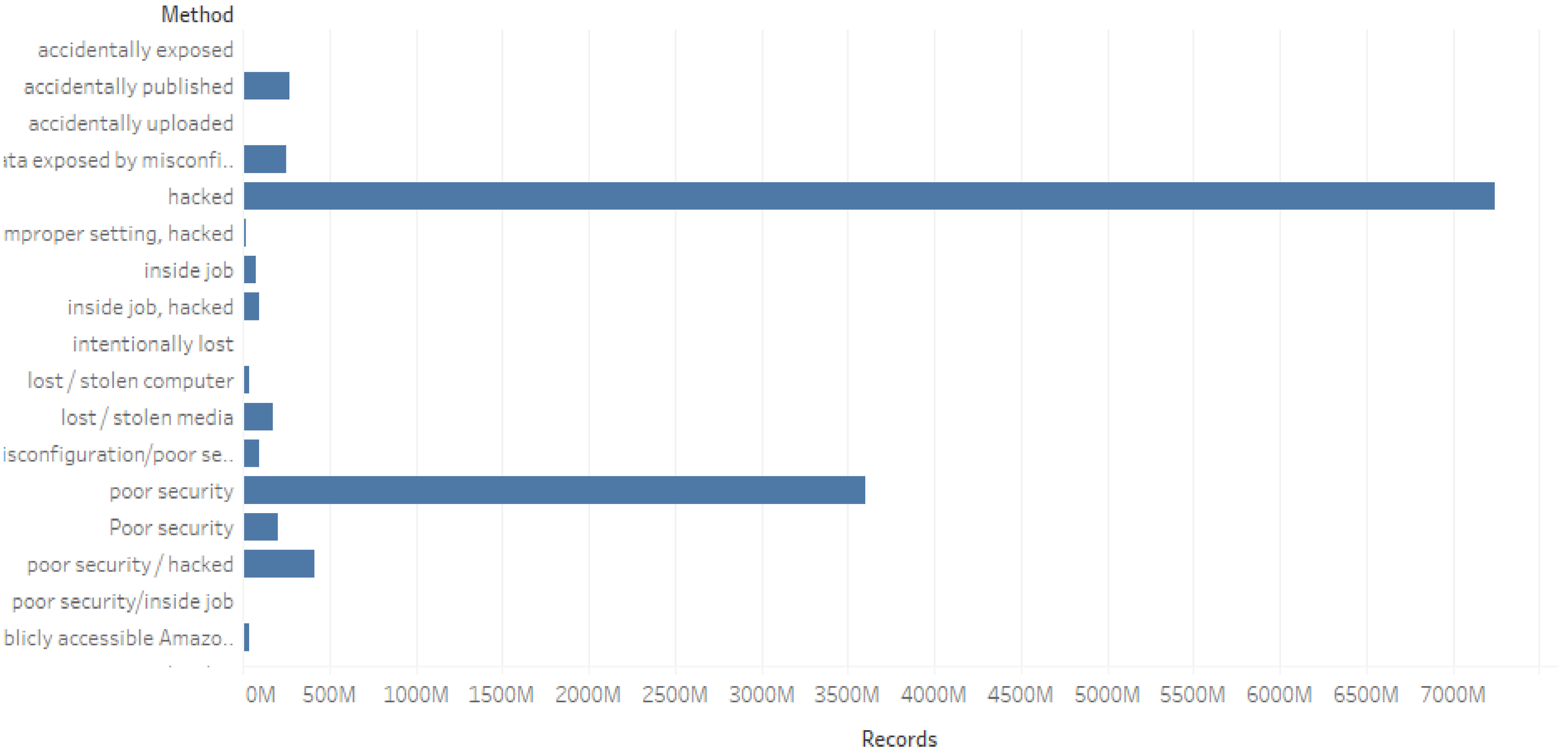


Tableau Plots
Packed Bubble



Horizontal Bar



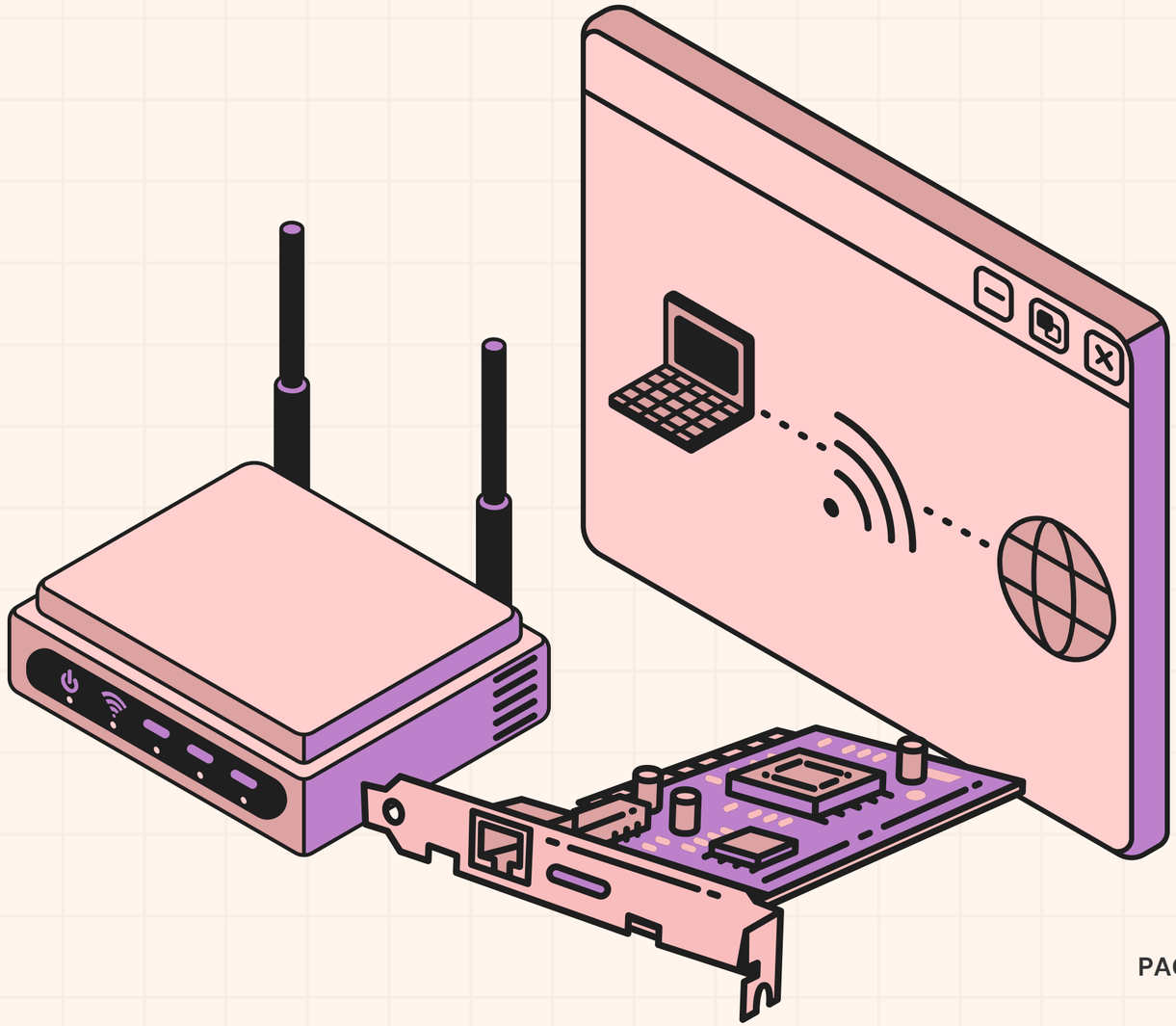
CONCLUSION:

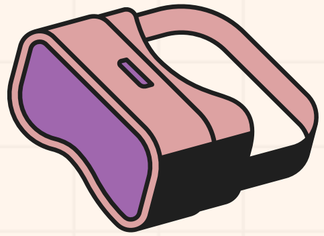
This project has been a series of planned tasks with an objective, often involving multiple steps. It was be carried out by multiple individuals that formed or a team and was designed to achieve the specified outcome within a set timeframe (today). This class has taught us punctuality, how to data—visually, via code, and with teamwork

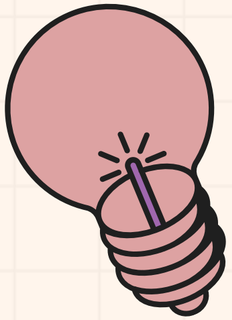
REFERENCES

Arrests by offense, age, and race | Office of Juvenile Justice and Delinquency Prevention. (n.d.). https://ojjdp.ojp.gov/statistical-briefing-book/crime/faqs/ucr_table_2 – filtered by age/year

Top Data Breaches (2004-2021)
https://www.kaggle.com/datasets/hishaamarmghan/list-of-top-data-breaches-2004-2021







THANK YOU

Bowie State University

Aaronea Wiggins
Lojain Idris