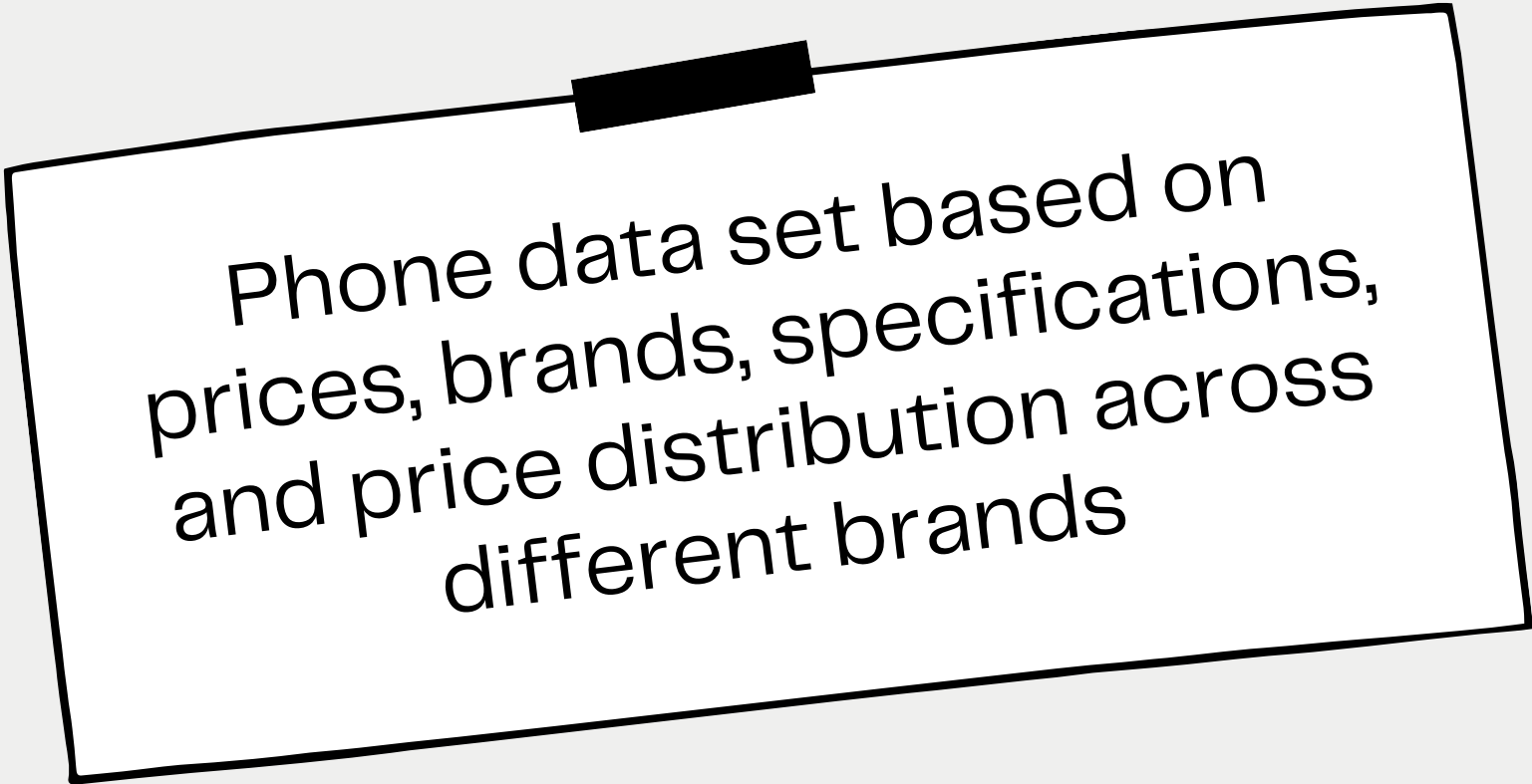


# Phone Data set



Phone data set based on  
prices, brands, specifications,  
and price distribution across  
different brands

## Q DataSet

This dataset comprises phone features and prices of popular brands, with every phone. It includes attributes such as phone name, brand, operating system, screen size and resolution, battery capacity and type, RAM, announcement date, weight, storage capacity, and various camera features like 720p, 1080p, 4K, and 8K video capabilities at different frame rates. Finally, it also provides the price of each phone in USD.



# PRICE

Adding a new column where we multiplied the amount of  
usd to the conversion of usd to php to have a column of ph  
prices

Create a new column Classification that display "price(PHP)" if  
the value of outcome is price(USD)

```
In [ ]: usd_to_php = 55  
p_data['price(PHP)'] = p_data['price(USD)'] * usd_to_php  
p_data
```

```
Out[ ]: | ...  video_4K  video_8K  video_30fps  video_60fps  video_120fps  video_240fps  video_480fps  video_960fps  price(USD)  price(PHP)
```

...	False	False	True	False	False	False	False	False	120.000	6600.00
...	False	False	True	False	False	False	False	False	100.000	5500.00



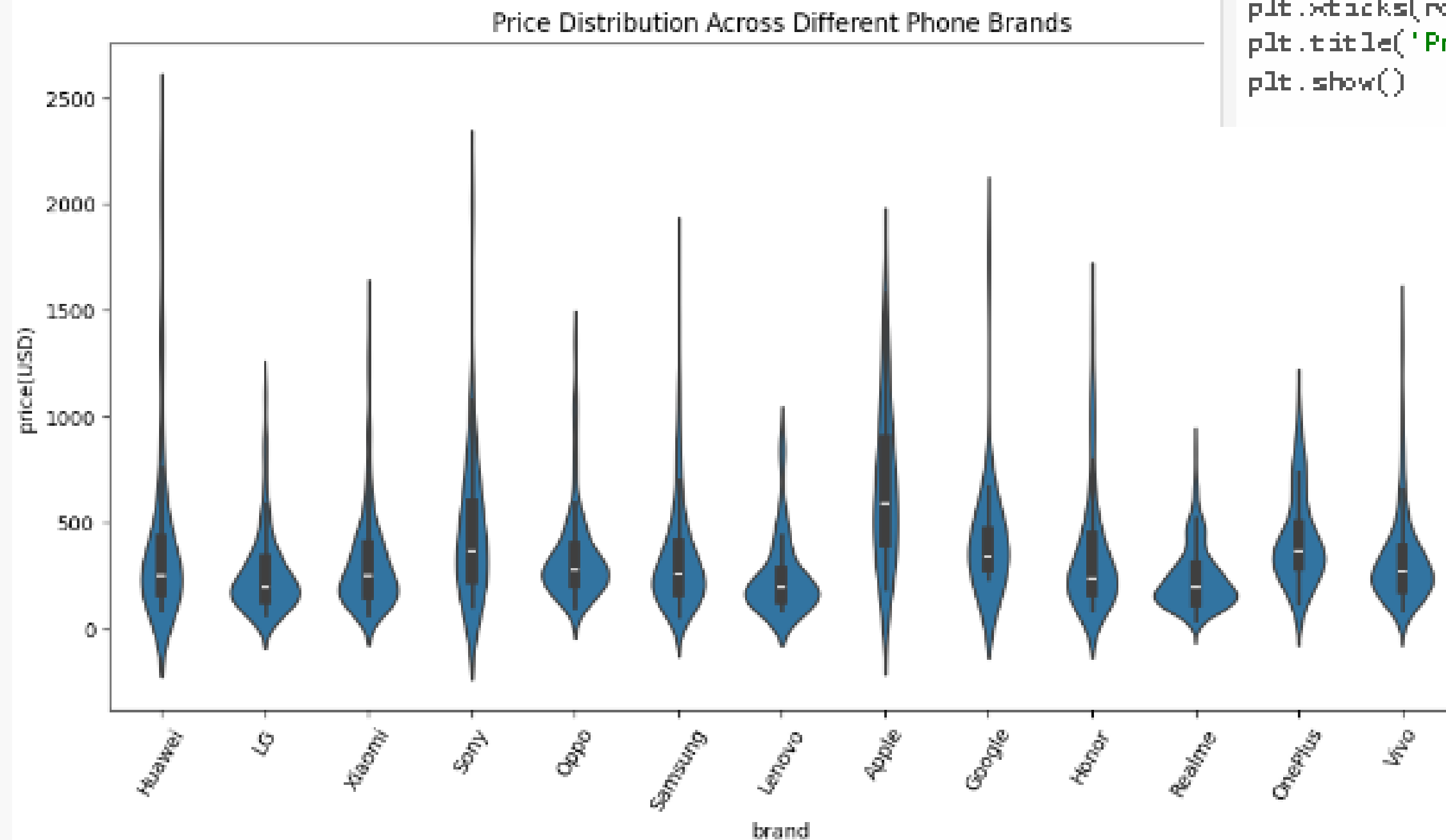
# Violin Graph

Used a violin plot graph, imported a matplotlib and also seaborn to make the violin plot work..

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

phones = pd.read_csv('cleaned_all_phones.csv')

plt.figure(figsize=(12, 6))
sns.violinplot(x='brand', y='price(USD)', data=phones)
plt.xticks(rotation=60)
plt.title('Price Distribution Across Different Phone Brands')
plt.show()
```



Price distribution across different phone brands, we will be able to analyze what brand has the most increase in its price

# DataFrame



Creating a HighRam DataFrame to be able to specify which phones have ram that exceeds or equal to 15

Create a new dataframe "HighRam" that gathers data with ram(GB) greater than or equal to 15

```
In [ ]: p_data = pd.DataFrame(p_data)
HighRam = p_data[p_data['ram(GB)'] >= 15].copy()

HighRam
```

```
Out[ ]:
```

	phone_name	brand	os	inches	resolution	mAh	battery_type	ram(GB)	announcement_date	weight(g)	...	video_
468	Galaxy S20 Ultra	Samsung	Android 10	6.90	1440x3200	5000	Li-Ion	16	2020-02-11	220.0	...	Tr
470	Galaxy S20 Ultra 5G	Samsung	Android 10	6.90	1440x3200	5000	Li-Ion	16	2020-02-11	222.0	...	Tr
412	Galaxy S20	Samsung	Android	6.7	1080x2340	5000	Li-Ion	16	2020-02-11	200.0	...	Tr



## Display the total records of High ram phones

```
[ ]: print("total records of High ram(GB)phone: ", len(HighRam))
```

```
total records of High ram(GB)phone: 35
```

## Display all the mid range phones available in the data set in php

```
[ ]: p_data = pd.DataFrame(p_data)

midrange = 25000
f = p_data[p_data['price(PHP)'] <= midrange]
print("price(PHP)\tphone_name")
for i, row in f.iterrows():
    print("P{:.0f}\t\t{}".format(row['price(PHP)'], row['phone_name']))
```

price(PHP)	phone_name
P6600	Y6II Compact
P5500	K20 plus
P23100	P8 Lite (2017)

Displaying the total high ram and mid range phones prices.

# Mean



Use numpy to get the average of inch and prices of phones

```
In [ ]: m_usd = np.mean(p_data['price(USD)'])  
m_php = np.mean(p_data['price(PHP)'])  
m_inch = np.mean(p_data['inches'])  
  
print("Average prices of phone in USD: ${:.2f}".format(m_usd))  
print("Average prices of phone in PHP: ₱{:.2f}".format(m_php))  
print("Average inch of phone in the dataset: {:.2f}".format(m_inch))
```

```
Average prices of phone in USD: $337.85  
Average prices of phone in PHP: ₱18581.59  
Average inch of phone in the dataset: 6.42
```

# Median



Use numpy to get the Median inches and prices of phone.

```
[ ]: m_usd = np.median(p_data['price(USD)'])  
      m_php = np.median(p_data['price(PHP)'])  
      m_inch = np.median(p_data['inches'])  
  
      print("Median prices of phone in USD: ${:.2f}".format(m_usd))  
      print("Median prices of phone in PHP: ₱{:.2f}".format(m_php))  
      print("Median inch of phone in the dataset: {:.2f}".format(m_inch))
```

```
Median prices of phone in USD: $260.00  
Median prices of phone in PHP: ₱14300.00  
Median inch of phone in the dataset: 6.50
```





# STANDARD DEVIATION

Use numpy to get the standard deviation of the phone prices.

```
[ ]: std_usd = np.std(p_data['price(USD)'])  
std_php = np.std(p_data['price(PHP)'])  
  
print("standard deviation prices of phone in USD: ${:.2f}".format(std_usd))  
print("standard deviation prices of phone in PHP: ₱{:.2f}".format(std_php))
```

```
standard deviation prices of phone in USD: $266.65  
standard deviation prices of phone in PHP: ₱14665.89
```