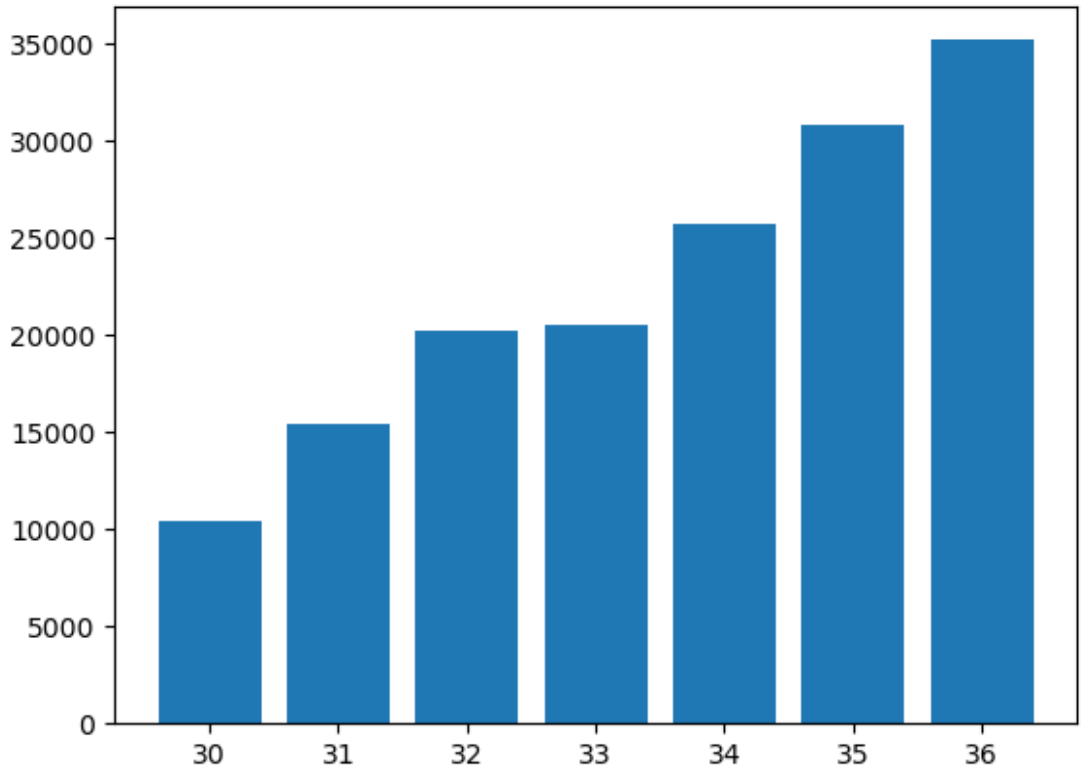
```
import matplotlib.pyplot as plt
%matplotlib inline

import numpy as np
```

Bar Plots

```
x = [30, 31, 32, 33, 34, 35, 36]
y =  [10365, 15364, 20236, 20478, 25698, 30785, 35156]
plt.bar(x, y)
```

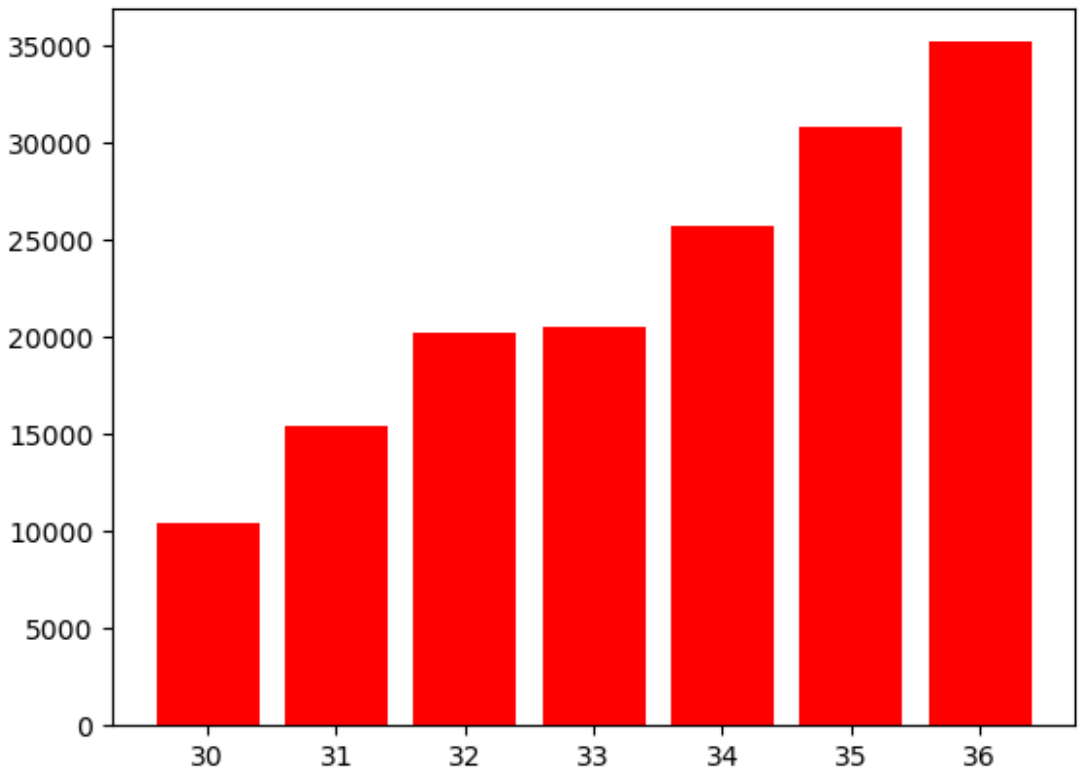
```
<BarContainer object of 7 artists>
```



Change The Color Of The Bars

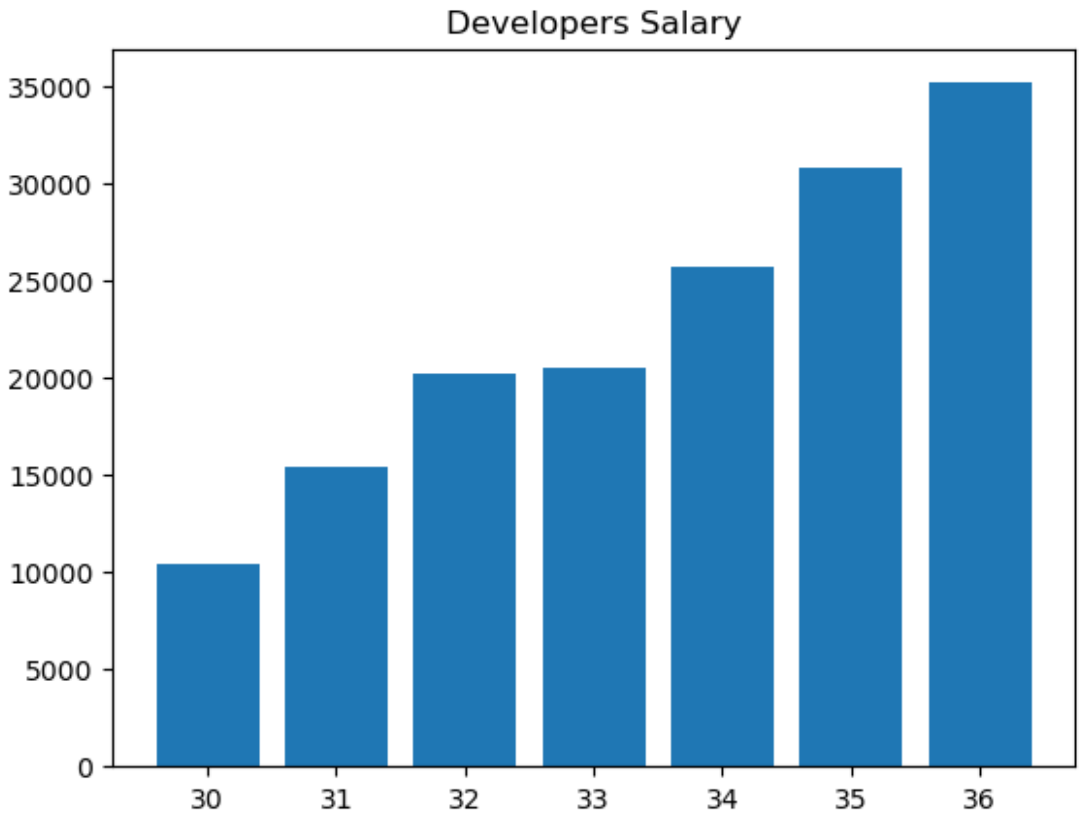
```
x = [30, 31, 32, 33, 34, 35, 36]
y =  [10365, 15364, 20236, 20478, 25698, 30785, 35156]
plt.bar(x, y, color="red")
```

```
<BarContainer object of 7 artists>
```



Write Title To The Bar

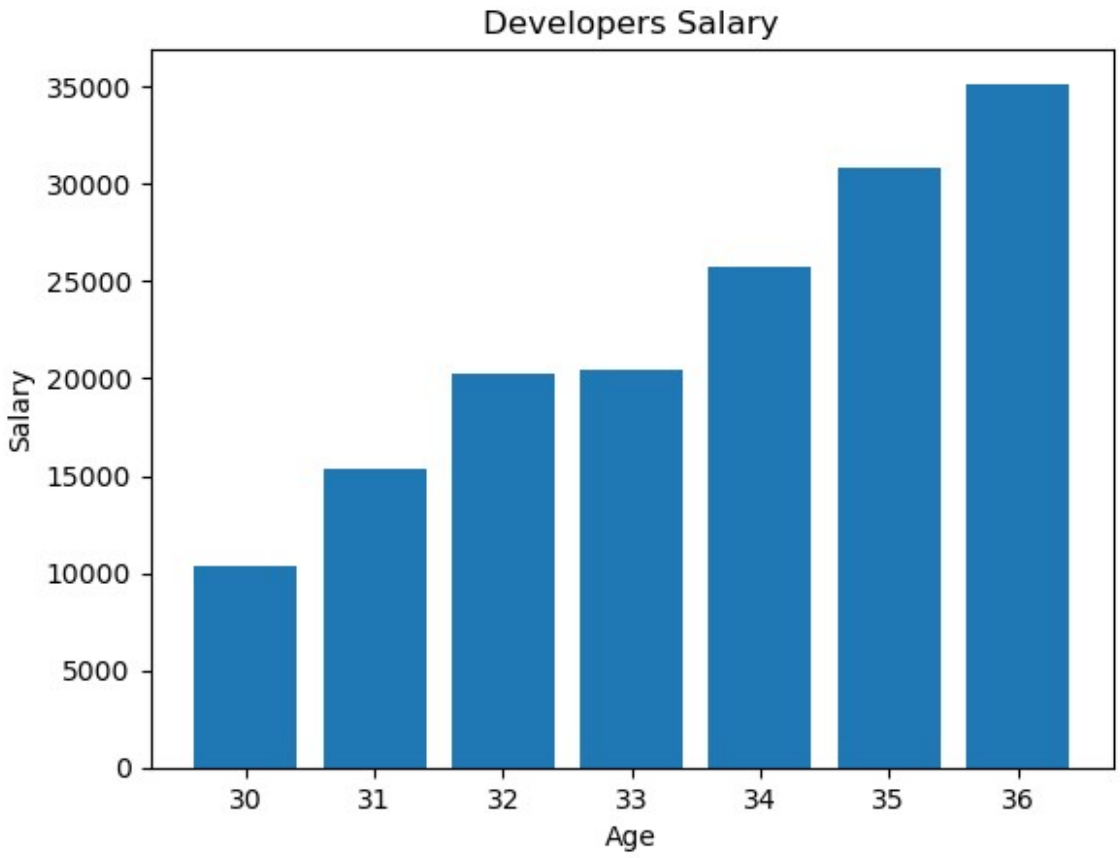
```
x = [30, 31, 32, 33, 34, 35, 36]
y =  [10365, 15364, 20236, 20478, 25698, 30785, 35156]
plt.bar(x, y)

# Title The Bars
plt.title("Developers Salary")

Text(0.5, 1.0, 'Developers Salary')
```



Title The X-axis and Y-axis

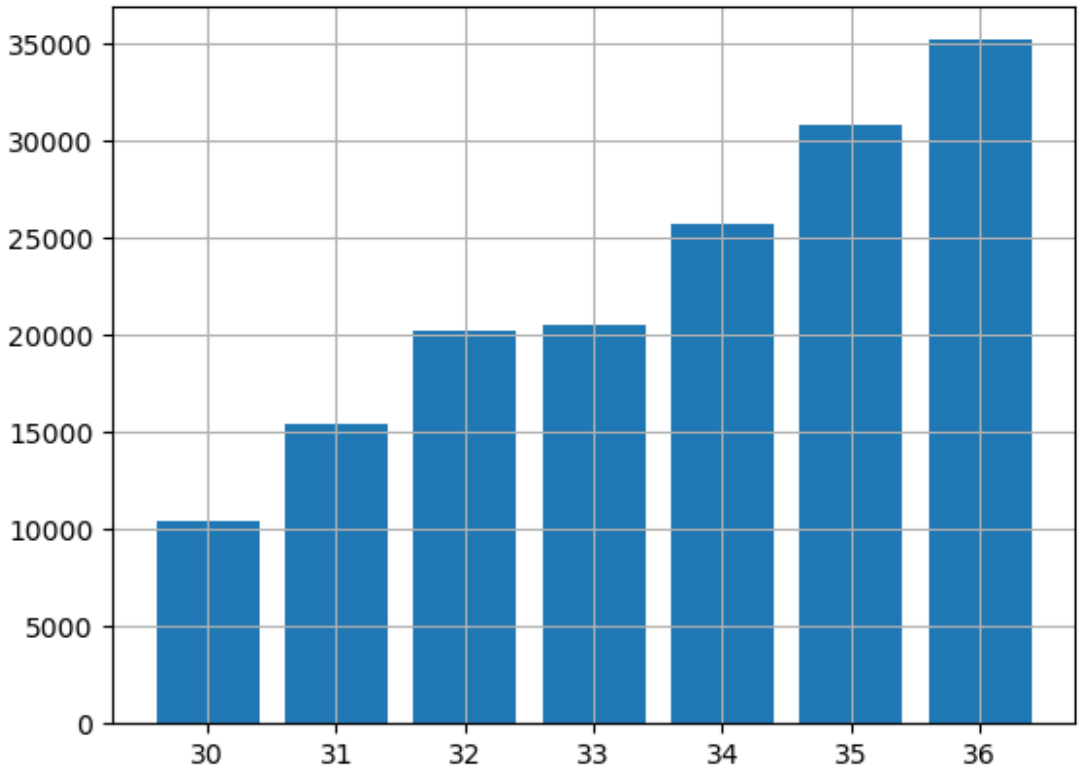
```
x = [30, 31, 32, 33, 34, 35, 36]
y =  [10365, 15364, 20236, 20478, 25698, 30785, 35156]
plt.bar(x, y)

# Name The Bars
plt.title("Developers Salary")

# Title The X-axis
plt.xlabel("Age")

# Title The Y-axis
plt.ylabel("Salary")
```

```
Text(0, 0.5, 'Salary')
```



Show The Grids

```
x = [30, 31, 32, 33, 34, 35, 36]
y =  [10365, 15364, 20236, 20478, 25698, 30785, 35156]
plt.bar(x, y)

# Show The Grids On The Graph
plt.grid()
```

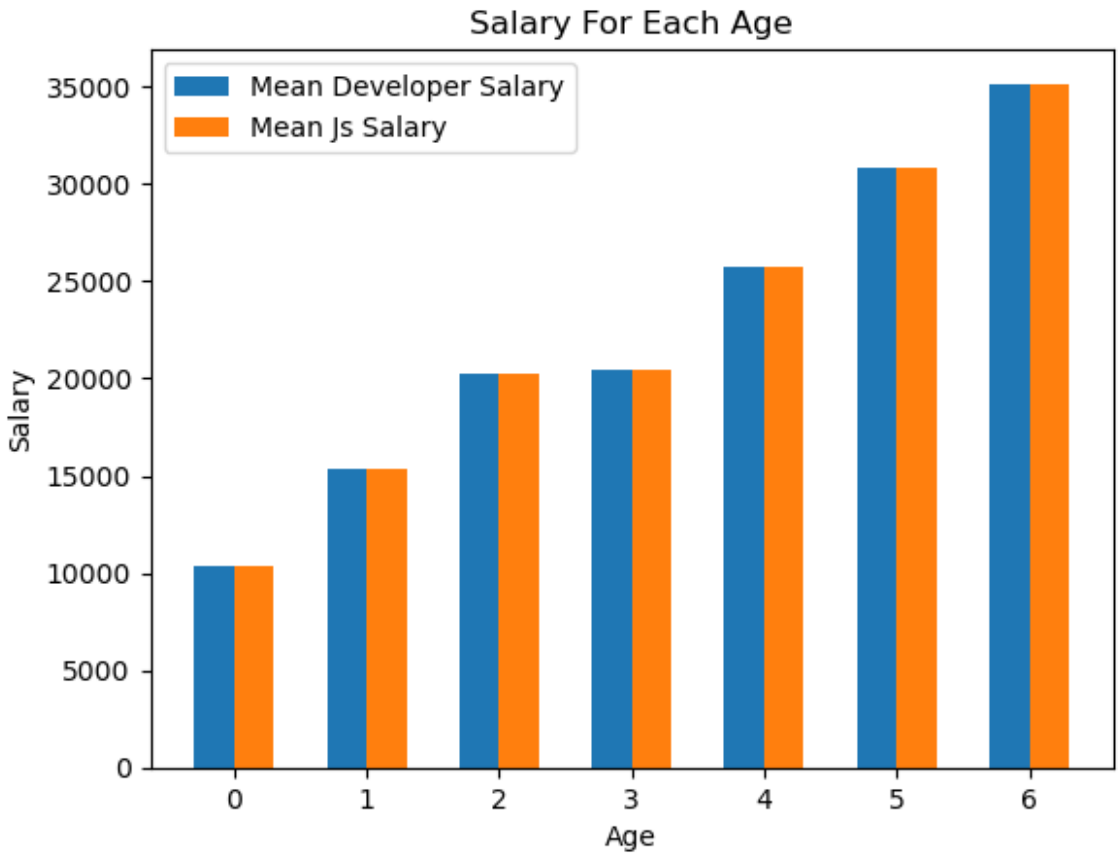


Make More Than One Bar At The Same Graph

- While Using Matplotlib, Make Override One To The Other
- Soon, Will Use "Seaborn" Library To Solve This Problem

```
dev_x = [30, 31, 32, 33, 34, 35, 36]
dev_y = [10365, 15364, 20236, 20478, 25698, 30785, 35156]

# This Will Be The X-axis for Two Bars
x_indecies = np.arange(len(dev_x))

# This Is The Width Of The Bars
width = 0.30

plt.title("Salary For Each Age")
plt.xlabel('Age')
plt.ylabel('Salary')

# Show First Bar
plt.bar(x_indecies - width / 2, dev_y, label='Mean Developer Salary',
width=width)

# Show Second Bar
js_dev_x = [30, 31, 32, 33, 34, 35, 36]
js_dev_y = [10365, 15364, 20236, 20478, 25698, 30785, 35156]
```

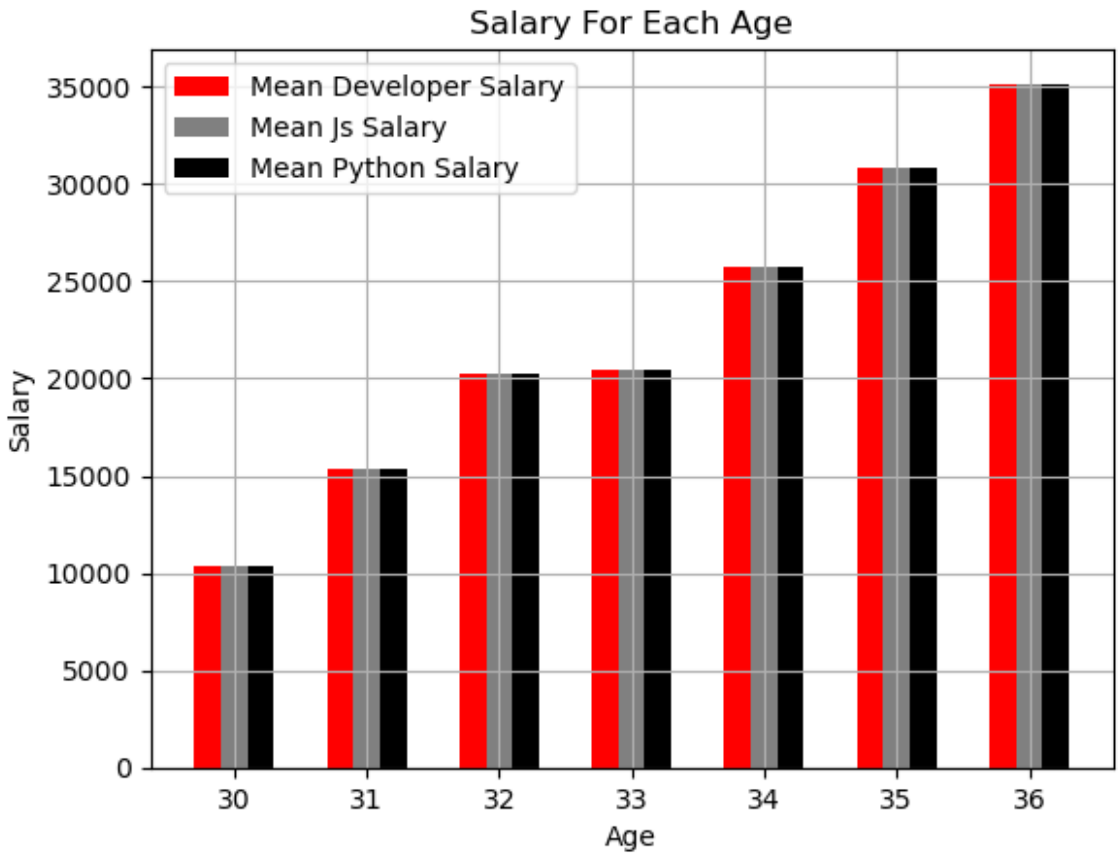
```
# Show The Second Bar
plt.bar(x_indecies + width / 2, dev_y, label='Mean Js Salary',
width=width)

# Write The Label
plt.legend()

# Show The Bars
plt.show()
```



Make 3 Bars At The Same Graph

```
# Data For First Bar
dev_x = [30, 31, 32, 33, 34, 35, 36]
dev_y = [10365, 15364, 20236, 20478, 25698, 30785, 35156]

# This Will Be The X-axis for Bars
x_indecies = np.arange(len(dev_x))

# This Is The Width Of The Bars
```

```
width = 0.2

plt.title("Salary For Each Age")
plt.xlabel('Age')
plt.ylabel('Salary')

# Show First Bar
plt.bar(x_indecies - width, dev_y, label='Mean Developer Salary',
width=width, color="red")

# Data For Second Bar
js_dev_x = [30, 31, 32, 33, 34, 35, 36]
js_dev_y = [10365, 15364, 20236, 20478, 25698, 30785, 35156]

# Show The Second Bar
plt.bar(x_indecies , dev_y, label='Mean Js Salary', width=width,
color="gray")

# Data For Third Bar
py_dev_x = [30, 31, 32, 33, 34, 35, 36]
py_dev_y = [10365, 15364, 20236, 20478, 25698, 30785, 35156]

# Show Third Bar
plt.bar(x_indecies + width , py_dev_y, label='Mean Python Salary',
width=width, color="black")

# Write The Label
plt.legend()

# ================================
# == Show Numbers On The X-axis ==
# ================================
# - The Length O The ticks Must Be Equal To Labels
# labels: x القيم اللي هبدلها مع القيم اللي موجوده في محور
# xticks: x القيم اللي موجوده في محور
plt.xticks(ticks = x_indecies, labels = dev_x)

# Show The Grid
plt.grid()

# Show The Bars
plt.show()
```


Salary For Each Age
Mean Developer Salary
Mean Js Salary
Mean Python Salary
35000
30000
25000
20000
15000
10000
5000
0
Salary
30
31
32
33
34
35
36
Age