

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

ModelA_16 = pd.read_excel("Model A.xlsx",sheet_name="16-17")
ModelA_17 = pd.read_excel("Model A.xlsx",sheet_name="17-18")
ModelA_18 = pd.read_excel("Model A.xlsx",sheet_name="18-19")

ModelB_16 = pd.read_excel("Model B.xlsx",sheet_name="16-17")
ModelB_17 = pd.read_excel("Model B.xlsx",sheet_name="17-18")
ModelB_18 = pd.read_excel("Model B.xlsx",sheet_name="18-19")

ModelC_16 = pd.read_excel("Model C.xlsx",sheet_name="16-17")
ModelC_17 = pd.read_excel("Model C.xlsx",sheet_name="17-18")
ModelC_18 = pd.read_excel("Model C.xlsx",sheet_name="18-19")

ModelD_16 = pd.read_excel("Model D.xlsx",sheet_name="16-17")
ModelD_17 = pd.read_excel("Model D.xlsx",sheet_name="17-18")
ModelD_18 = pd.read_excel("Model D.xlsx",sheet_name="18-19")

# # Convert into Numpy

# X and Y axis

head1=ModelA_18.iloc[1,2:9]
head2=head1.copy()

head1.to_numpy()
head2.to_numpy()

# MODEL-A

ModelA_16=ModelA_16.iloc[3:10, 2:9]
ModelA_17=ModelA_17.iloc[4:11, 2:9]
ModelA_18=ModelA_18.iloc[4:11, 2:9]

```

```
ModelA_16=ModelA_16.to_numpy()  
ModelA_17=ModelA_17.to_numpy()  
ModelA_18=ModelA_18.to_numpy()
```

```
# MODEL-B
```

```
ModelB_16=ModelB_16.iloc[3:10, 2:9]  
ModelB_17=ModelB_17.iloc[4:11, 2:9]  
ModelB_18=ModelB_18.iloc[4:11, 2:9]
```

```
ModelB_16=ModelB_16.to_numpy()  
ModelB_17=ModelB_17.to_numpy()  
ModelB_18=ModelB_18.to_numpy()
```

```
# MODEL-C
```

```
ModelC_16=ModelC_16.iloc[3:10, 2:9]  
ModelC_17=ModelC_17.iloc[4:11, 2:9]  
ModelC_18=ModelC_18.iloc[4:11, 2:9]
```

```
ModelC_16=ModelC_16.to_numpy()  
ModelC_17=ModelC_17.to_numpy()  
ModelC_18=ModelC_18.to_numpy()
```

```
# MODEL-D
```

```
ModelD_16=ModelD_16.iloc[3:10, 2:9]  
ModelD_17=ModelD_17.iloc[4:11, 2:9]  
ModelD_18=ModelD_18.iloc[4:11, 2:9]
```

```
ModelD_16=ModelD_16.to_numpy()  
ModelD_17=ModelD_17.to_numpy()  
ModelD_18=ModelD_18.to_numpy()
```

```
# # Storing All Values in 1 array
```

```
new = np.array([])
```

```
for i in range(7):
```

```
    for j in range(7):
```

```
        new = np.append(new, ModelA_16[i][j])
```

```
        new = np.append(new, ModelB_16[i][j])
```

```
        new = np.append(new, ModelC_16[i][j])
```

```
        new = np.append(new, ModelD_16[i][j])
```

```
        new = np.append(new, ModelA_17[i][j])
```

```
        new = np.append(new, ModelB_17[i][j])
```

```
        new = np.append(new, ModelC_17[i][j])
```

```
        new = np.append(new, ModelD_17[i][j])
```

```
        new = np.append(new, ModelA_18[i][j])
```

```
        new = np.append(new, ModelB_18[i][j])
```

```
        new = np.append(new, ModelC_18[i][j])
```

```
        new = np.append(new, ModelD_18[i][j])
```

```
Final_Data = np.reshape(new,(147,4))
```

```
print(Final_Data)
```

```
# # Plotting On Matplotlib
```

```
Count = 0
```

```
for value1 in head1:# value1 is for X axis
```

```
    for value2 in head2:# value2 is for y axis
```

```
        x = ["Model A \n(Product x Product)","Model B \n(Product x Product)","Model C \n(Industry x Industry)","Model D \n(Industry x Industry)"]
```

```
        y1 = Final_Data[Count]
```

```
        y2 = Final_Data[Count+1]
```

```

y3 = Final_Data[Count+2]
Count=Count+3 # Count is used to change values Final_Data

# Inserting Bar Plots
plt.rcParams['font.size']='5'
plt.rcParams['figure.dpi']=600
plt.rcParams['savefig.dpi']=600
bar_width = 0.08 # width of bars
width = 0.3 # space given between bars
p = np.arange(len(x))
p1 = [j+width for j in p]
p2 = [j+width for j in p1]
fig, ax = plt.subplots()
y1 = ax.bar(p, y1, bar_width,color="royalblue",label="2016-17")
y2 = ax.bar(p1, y2, bar_width,color="seagreen",label="2017-18")
y3 = ax.bar(p2, y3, bar_width,color="indianred",label="2018-19")

# inserting x axis label
ax.set_xticks(p+width) #Location
ax.set_xticklabels(x)#Name

# inserting Labels
plt.xlabel(value1+"\n\n (Input)", fontsize=6)
plt.ylabel(value2+"\n\n (Output)", fontsize=6)
plt.title("(Values in ₹ Crore)")

# inserting legend(specify the location for the bar label)
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left', borderaxespad=0.)

# inserting Values on bars
def insert_data_labels(bars):
    for bar in bars:
        bar_height = bar.get_height()
        if bar_height < 0:

```

```

        #bar_height = bar.get_height()
        ax.annotate('{0:.0f}'.format(bar.get_height()),
                    xy=(bar.get_x() + bar.get_width() / 2, bar_height),
                    xytext=(0,-60),
                    textcoords='offset pixels',
                    ha='center',
                    va='bottom'
                    )
    else:
        #bar_height = bar.get_height()
        ax.annotate('{0:.0f}'.format(bar.get_height()),
                    xy=(bar.get_x() + bar.get_width() / 2, bar_height),
                    xytext=(0,2),
                    textcoords='offset pixels',
                    ha='center',
                    va='bottom'
                    )

insert_data_labels(y1)
insert_data_labels(y2)
insert_data_labels(y3)

plt.show()

```