Student ID: 19522348

Full name: LÊ ĐỨC TÍN

```
import numpy as np
x = np.linspace(0, 5, 11)
y = x**2
```

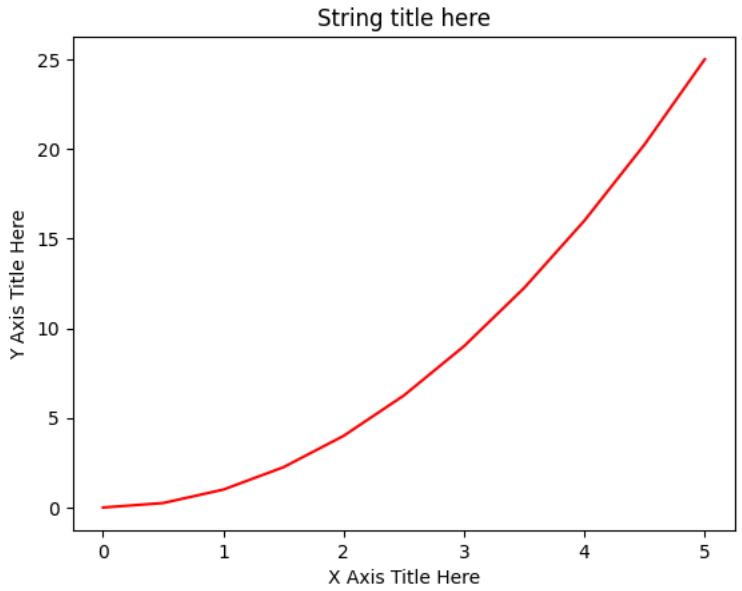
```
x
```

```
array([0. , 0.5, 1. , 1.5, 2. , 2.5, 3. , 3.5, 4. , 4.5, 5. ])
```

```
y
```

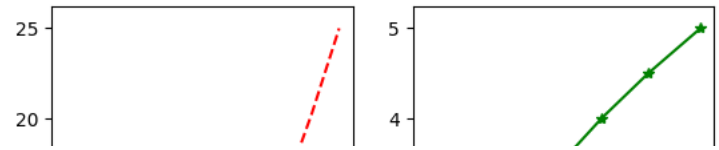
```
array([ 0.  ,  0.25,  1.  ,  2.25,  4.  ,  6.25,  9.  , 12.25, 16.  ,
       20.25, 25.  ])
```

```
import matplotlib.pyplot as plt
plt.plot(x, y, 'r')
plt.xlabel('X Axis Title Here')
plt.ylabel('Y Axis Title Here')
plt.title('String title here')
plt.show()
```



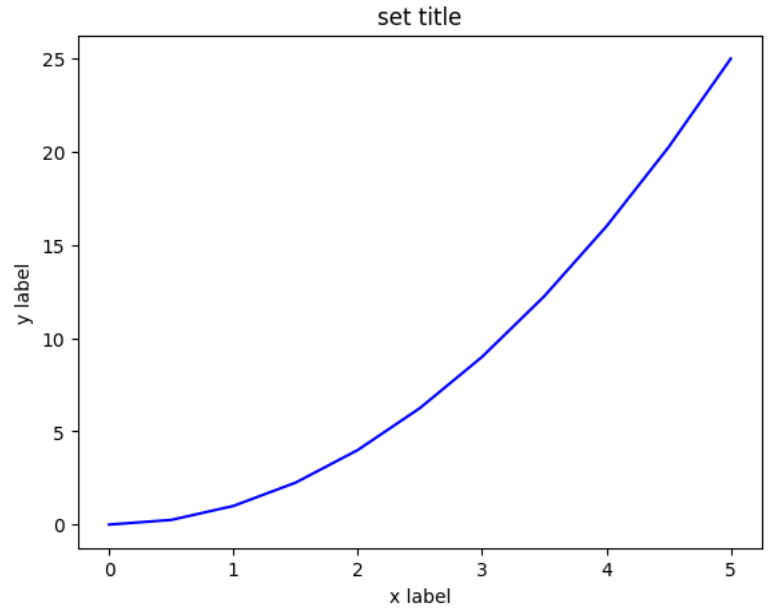

```
plt.subplot(1, 2, 1)
plt.plot(x, y, 'r--')
plt.subplot(1, 2, 2)
plt.plot(y, x, 'g*-')
```

```
[<matplotlib.lines.Line2D at 0x7efed88ca970>]
```




```
fig = plt.figure()

axes = fig.add_axes([0.1, 0.1, 0.8, 0.8])

axes.plot(x, y, 'b')
axes.set_xlabel('x label')
axes.set_ylabel('y label')
axes.set_title('set title')
```

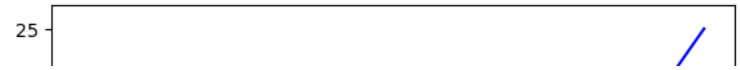
```
Text(0.5, 1.0, 'set title')
```



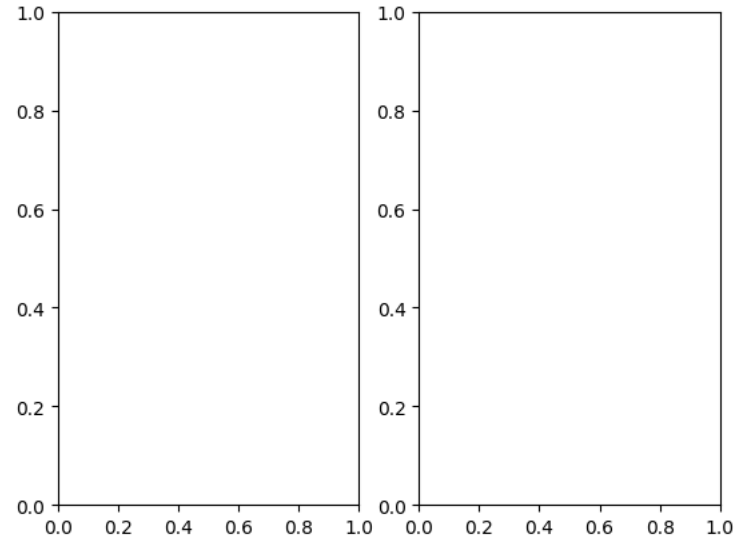
6. Subplot()

```
fig = plt.figure()

axes1 = fig.add_axes([0.1, 0.1, 0.8, 0.8])
axes2 = fig.add_axes([0.2, 0.5, 0.4, 0.3])

axes1.plot(x, y, 'b')

axes2.plot(y, x, 'r')
```

```
[<matplotlib.lines.Line2D at 0x7efed862ff40>]
```



```
fig, axes = plt.subplots(nrows=1, ncols=2)
```

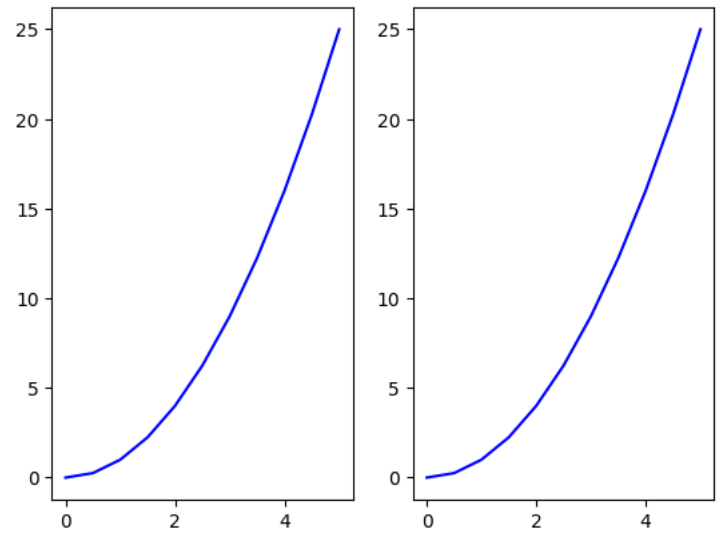


```
axes
```

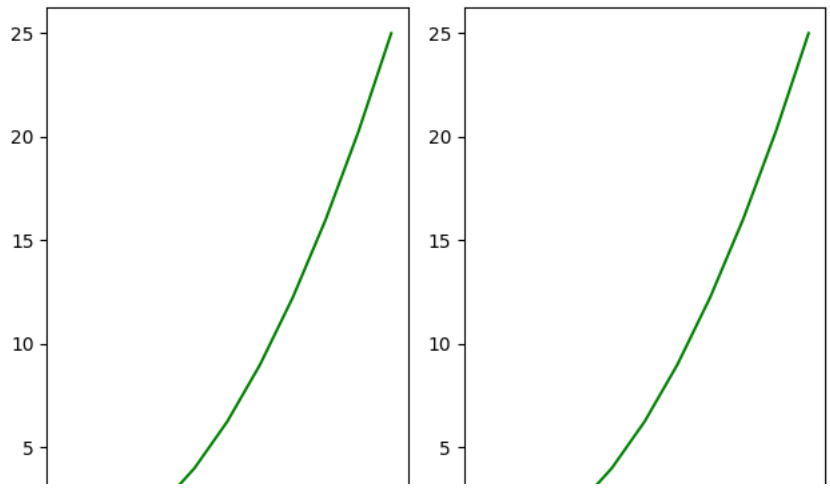
```
array([<Axes: >, <Axes: >], dtype=object)
```

```
for ax in axes:
  ax.plot(x, y, 'b')
```

```
fig
```



```
fig, axes = plt.subplots(nrows=1, ncols=2)

for ax in axes:
  ax.plot(x, y, 'g')

fig
plt.tight_layout()
```



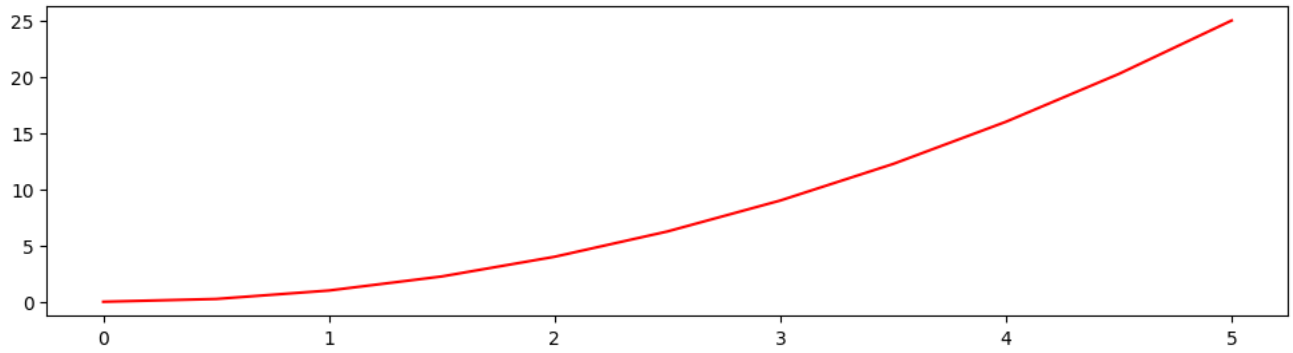
7. Figure size, aspect ratio and DPI

```
fig = plt.figure(figsize=(8,4), dpi=100)
```

```
<Figure size 800x400 with 0 Axes>
```

```
fig, axes = plt.subplots(figsize=(12, 3))

axes.plot(x, y, 'r')
```

```
[<matplotlib.lines.Line2D at 0x7efed3fe4580>]
```




```
fig.savefig('filename.png')
```

```
fig.savefig("filename.png", dpi=200)
```

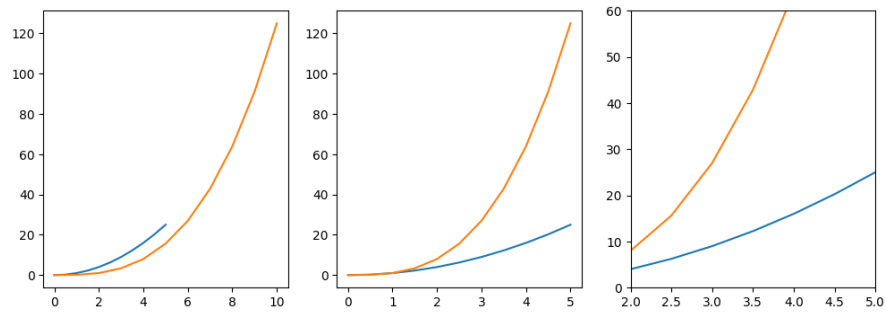
```
fig = plt.figure()

ax = fig.add_axes([0, 0, 1, 1])

ax.plot(x, x**2, label="x**2")
ax.plot(x, x**3, label="x**3")
ax.legend()
```

```
<matplotlib.legend.Legend at 0x7efed8109250>
```

12. Plot range

```
fig, axes = plt.subplots(1, 3, figsize=(12, 4))

axes[0].plot(x, x**2, x**3)

axes[1].plot(x, x**2, x, x**3)
axes[1].axis('tight')

axes[2].plot(x, x**2, x, x**3)
axes[2].set_ylim([0, 60])
axes[2].set_xlim([2, 5])
```

```
(2.0, 5.0)
```



II. Seaborn

1. Load testing dataset

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

%matplotlib inline
sns.get_dataset_names()
```

```
['anagrams',
 'anscombe',
 'attention',
 'brain_networks',
 'car_crashes',
 'diamonds',
 'dots',
 'dowjones',
 'exercise',
```

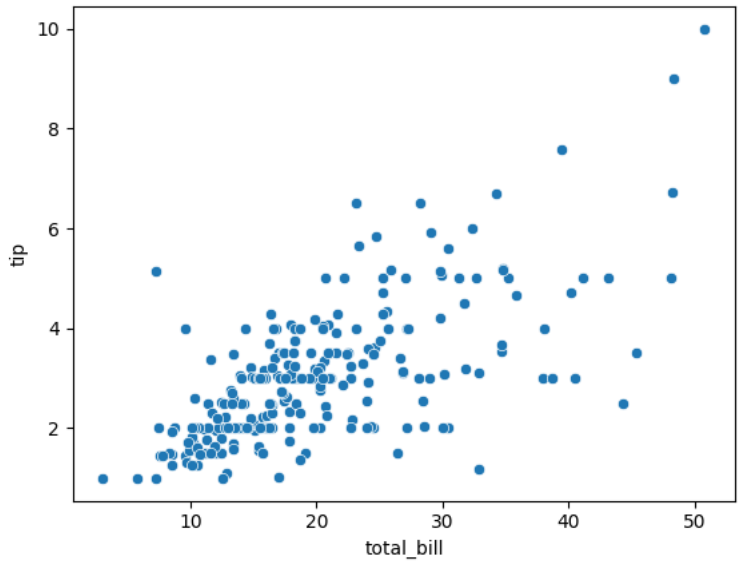
```
'flights',
'fmri',
'geyser',
'glue',
'healthexp',
'iris',
'mpg',
'penguins',
'planets',
'seaice',
'taxis',
'tips',
'titanic']
```

```
tips = sns.load_dataset("tips")
tips.head()
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

2. Scatter plot

```
ax = sns.scatterplot(x="total_bill", y="tip", data=tips)
```



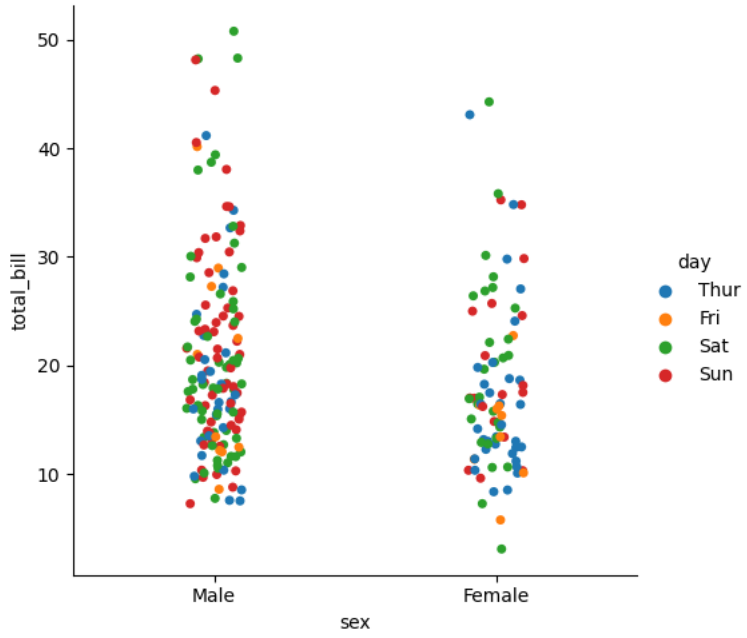

```
sns.relplot(x="total_bill", y="tip", data=tips, kind="scatter", hue="sex", size="size")
```

```
<seaborn.axisgrid.FacetGrid at 0x7efec2aba3d0>
```

3. Categorical functions

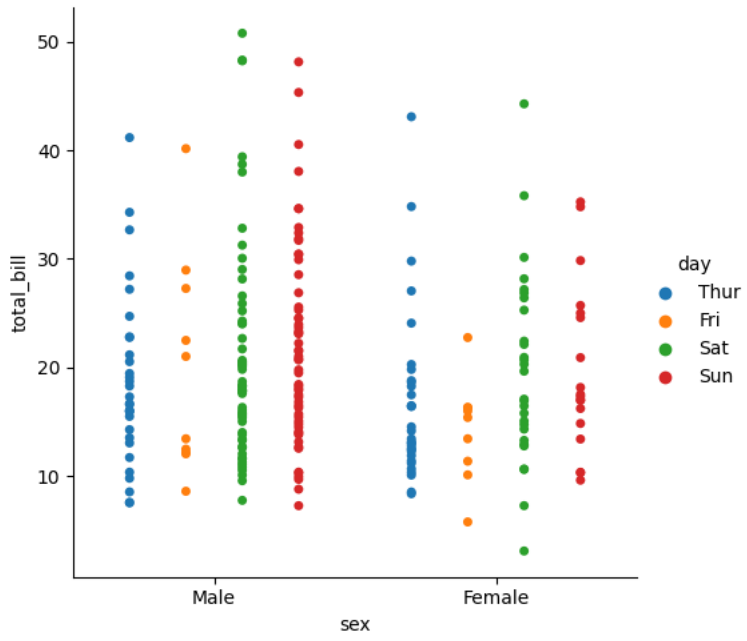
```
sns.catplot(x="sex", y="total_bill", hue="day", data=tips, kind="strip")
```

```
<seaborn.axisgrid.FacetGrid at 0x7efeca0d5460>
```



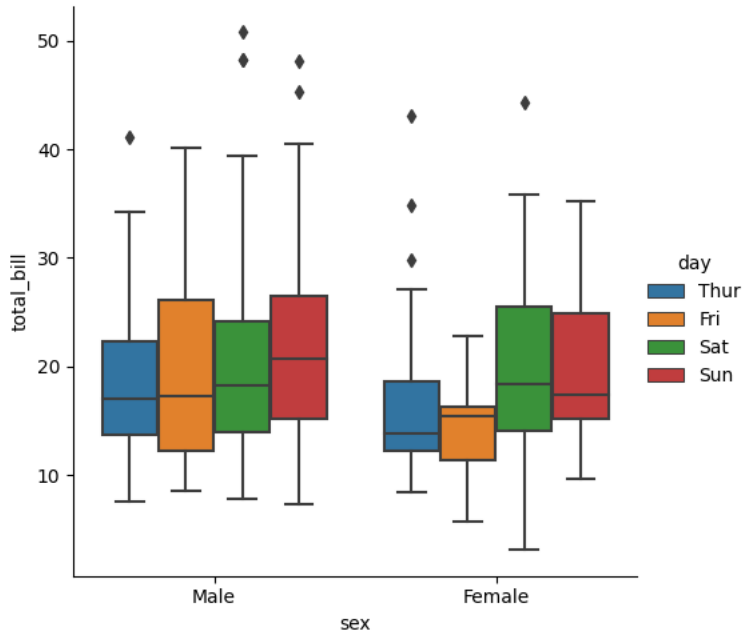

```
sns.catplot(x="sex", y="total_bill", hue="day", data=tips, kind="strip", jitter=False, dodge=True)
```

```
<seaborn.axisgrid.FacetGrid at 0x7efec27778e0>
```



```
sns.catplot(x="sex", y="total_bill", hue="day", data=tips, kind="box")
```

```
<seaborn.axisgrid.FacetGrid at 0x7efec2a9e640>
```



III. Exercises

1. Job Market

```
dataset = pd.read_csv('/content/job-market.csv')
```

```
dataset = dataset.dropna()
```

```
dataset['Id'] = dataset['Id'].dropna()
```

```
dataset['Id'].astype(int)
```

```
121      37404238
122      37404195
125      37404288
126      37404267
127      37404230
           ...
10091    37388929
10094    37388912
10096    37388901
10097    37388898
10098    37388893
Name: Id, Length: 5898, dtype: int64
```

```
Job = dataset['Classification'].values
Job
```

```
array(['Trades & Services', 'Trades & Services', 'Education & Training',
       ..., 'Administration & Office Support', 'Hospitality & Tourism',
       'Call Centre & Customer Service'], dtype=object)
```

```
!pip install pyspark
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting pyspark
  Downloading pyspark-3.4.0.tar.gz (310.8 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 310.8/310.8 MB 4.1 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: py4j==0.10.9.7 in /usr/local/lib/python3.9/dist-packages (from pyspark) (0.10.9.7)
Building wheels for collected packages: pyspark
```

```
  Building wheel for pyspark (setup.py) ... done
  Created wheel for pyspark: filename=pyspark-3.4.0-py2.py3-none-any.whl size=311317145 sha256=ec41a6f231bb5e3c4c75cc3e9d43becfc43d8577
  Stored in directory: /root/.cache/pip/wheels/9f/34/a4/159aa12d0a510d5ff7c8f0220abbea42e5d81ecf588c4fd884
Successfully built pyspark
Installing collected packages: pyspark
Successfully installed pyspark-3.4.0
```

```
from pyspark.sql import SparkSession
spark = SparkSession.builder.appName('SparkByExamples.com').getOrCreate()


rdd=spark.sparkContext.parallelize(Job)


rdd2=rdd.map(lambda x: (x,1))
for element in rdd2.collect():
    print(element)
```

```
('Accounting', 1)
('Administration & Office Support', 1)
('Information & Communication Technology', 1)
('Call Centre & Customer Service', 1)
('Manufacturing, Transport & Logistics', 1)
('Manufacturing, Transport & Logistics', 1)
('Trades & Services', 1)
('Hospitality & Tourism', 1)
('Accounting', 1)
('Retail & Consumer Products', 1)
('Sales', 1)
('Hospitality & Tourism', 1)
('Administration & Office Support', 1)
('Hospitality & Tourism', 1)
('Manufacturing, Transport & Logistics', 1)
('Trades & Services', 1)
('Education & Training', 1)
('Accounting', 1)
('Trades & Services', 1)
('Call Centre & Customer Service', 1)
('Real Estate & Property', 1)
('Information & Communication Technology', 1)
('Administration & Office Support', 1)
('Trades & Services', 1)
('Call Centre & Customer Service', 1)
('Trades & Services', 1)
('Call Centre & Customer Service', 1)
('Hospitality & Tourism', 1)
('Sales', 1)
('Human Resources & Recruitment', 1)
('Accounting', 1)
('Retail & Consumer Products', 1)
('Hospitality & Tourism', 1)
('Administration & Office Support', 1)
('Sales', 1)
('Manufacturing, Transport & Logistics', 1)
('Manufacturing, Transport & Logistics', 1)
('Trades & Services', 1)
('Administration & Office Support', 1)
('Construction', 1)
('Accounting', 1)
('Hospitality & Tourism', 1)
('Manufacturing, Transport & Logistics', 1)
('Real Estate & Property', 1)
('Legal', 1)
('Retail & Consumer Products', 1)
('Trades & Services', 1)
('Healthcare & Medical', 1)
('Manufacturing, Transport & Logistics', 1)
('Administration & Office Support', 1)
('Sales', 1)
('Retail & Consumer Products', 1)
('Manufacturing, Transport & Logistics', 1)
('Science & Technology', 1)
('Administration & Office Support', 1)
('Administration & Office Support', 1)
('Hospitality & Tourism', 1)
('Call Centre & Customer Service', 1)
```

```
mapping = rdd2.collect()


mapping = sorted(mapping)
```

```
mapping[100]
```

```
('Accounting', 1)
```

```
arrayCount = []
count = 1
for i in range(len(mapping)-1):
  if mapping[i][0] == mapping[i+1][0]:
    count = count + 1
  else:
    arrayCount.append(count)
    count = 1
```

```
arrayTypes = []
count = 1
for i in range(len(mapping)-1):
  if mapping[i][0] != mapping[i+1][0]:
    arrayTypes.append(mapping[i][0])
```

```
print(len(arrayCount))
print(len(arrayTypes))
```

```
29
29
```

```
len(arrayTypes)
```

```
29
```

```
arrayFull = []
for i in range(len(arrayTypes)):
  arrayFull.append((arrayTypes[i], arrayCount[i]))
```

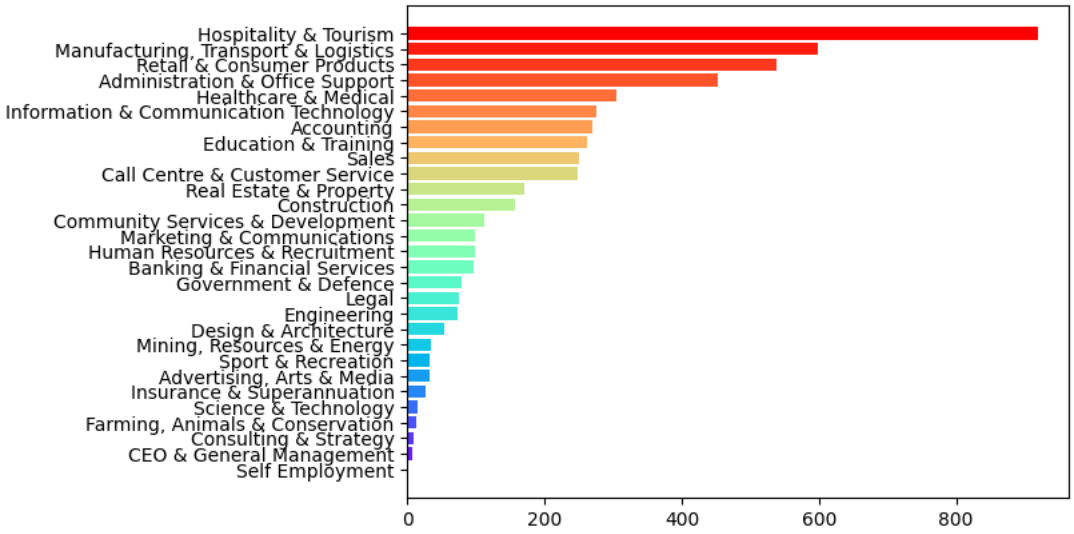
```
arrayFull.sort(key=lambda x:x[1])
```

```
Types = []
Counts = []
for each in range(len(arrayFull)):
  Types.append(arrayFull[each][0])
  Counts.append(arrayFull[each][1])
```

```
Types
```

```
['Self Employment',
 'CEO & General Management',
 'Consulting & Strategy',
 'Farming, Animals & Conservation',
 'Science & Technology',
 'Insurance & Superannuation',
 'Advertising, Arts & Media',
 'Sport & Recreation',
 'Mining, Resources & Energy',
 'Design & Architecture',
 'Engineering',
 'Legal',
 'Government & Defence',
 'Banking & Financial Services',
 'Human Resources & Recruitment',
 'Marketing & Communications',
 'Community Services & Development',
 'Construction',
 'Real Estate & Property',
 'Call Centre & Customer Service',
 'Sales',
 'Education & Training',
 'Accounting',
 'Information & Communication Technology',
 'Healthcare & Medical',
 'Administration & Office Support',
 'Retail & Consumer Products',
 'Manufacturing, Transport & Logistics',
 'Hospitality & Tourism']
```

```
colors = plt.cm.rainbow(np.linspace(0, 1, len(Types)))


plt.barh(Types, Counts, color=colors)
plt.show()
```




```
dataset
```

	Date	Location	Area	Classification	SubClassification	Requirement	FullDescription	LowestSalary	HighestSalary
	2018-10-07T00:00:00.000Z	Melbourne	Bayside & South Eastern Suburbs	Trades & Services	Welders & Boilermakers	Trade qualified person with skills in welding ...	\n * ...	0.0	30.0
	2018-10-07T00:00:00.000Z	Brisbane	Southern Suburbs & Logan	Trades & Services	Welders & Boilermakers	Perm rate $30. Structural steel fab & weld out...	One of Australia's best engineering worksho...	0.0	30.0

```
HighestSalary = dataset['HighestSalary'].values
LowestSalary = dataset['LowestSalary'].values
```

	07T00:00:00.000Z			Training	Childhood	searching for			

```
for each in Salary:
```

```
60.0
```

	07T00:00:00.000Z		M5 Corridor		Engineering	the people of t...	people o...		
	2018-10-07T00:00:00.000Z	Brisbane	CBD & Inner Suburbs	Information & Communication Technology	Networks & Systems Administration	Systems Engineer to work on BAU/Projects for a...	The Company This org...	0.0	30.0
	...	...	...	...	...	...	...	...	...
	2018-10-04T00:00:00.000Z	Melbourne	Eastern Suburbs	Science & Technology	Quality Assurance & Control	Global leaders in medical technology | Full -T...	A new and exciting opportunity for driven i...	50.0	60.0
	2018-10-04T00:00:00.000Z	Melbourne	CBD & Inner Suburbs	Administration & Office Support	PA, EA & Secretarial	The Executive Assistant will provide high leve...	V/Line, Victoria's largest regional passeng...	50.0	60.0
	2018-10-		CBD &	Administration &	Administrative	Dual position within the	Randstad are		

✓ 0 giây hoàn thành lúc 10:23