

# **DATA VISUALIZATION**

**PYTHON , MATPLOTLIB**

**2021**

**WinHtut**

**MEMBER AT GREEN HACKERS  
FOUNDER AT NATIONAL CYBER CITY**

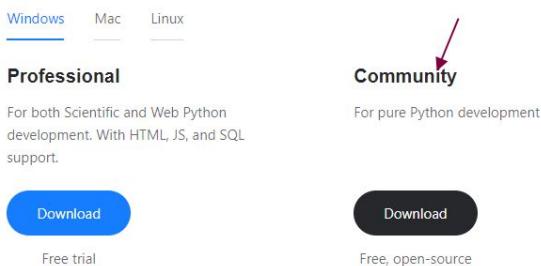
## Matplotlib For Visualization

ယခု သင်ခန်းစာမျာတော့ IDE အနေဖြင့် pycharm ကို အသုံးပြုသွားမှာ ဖြစ်ပါတယ်။ Professional project တွေပြုလုပ်တဲ့ အခါမှာ library ပြသာနများကို ဖြေရှင်း ပေးနိုင်ဖို့ pycharm ကို အသုံးပြုနိုင်ပါတယ် vs code မှာလည်း ဖြေရှင်း ပေးနိုင်သော်လည်း pycharm မှာ project စဉောက် ကတည်းက virtual enviroment တစ်ခုကို ဖန်တီး ပေးပါတယ်။ထိုကြောင့် programmer အနေဖြင့် step များစွာကို လုပ်နေစရာ မလိုပဲ project တစ်ခု ကို virtual env မှာ မြန်မြန် ဖန်တီး ပြီးသား ဖြစ်သလို အလုပ် မြန်မြန် ပြီးမြောက် ဆေါ်ပါတယ်။ pycharm ကို download လုပ်နိုင်ဖို့ အောက်ပါ Link ကိုသွားလိုက်ပါ။ မိမိတဲ့ အသုံးပြုသည့် operating system ကို အလိုလေ့ရှိခဲ့တဲ့ detect လုပ်ပြီး download page ကို ပြပေးပါလိမ့်မယ်။

<https://www.jetbrains.com/pycharm/download/>

စာရေးသူတို့ အနေဖြင့် လူတိုင်း အသုံးပြုနိုင်မည့် Community Version ကိုသာ အသုံးပြုမှာ ဖြစ်သည့် အတွက် community version ကို download ဆွဲရန် လိုအပ်ပါသည်။

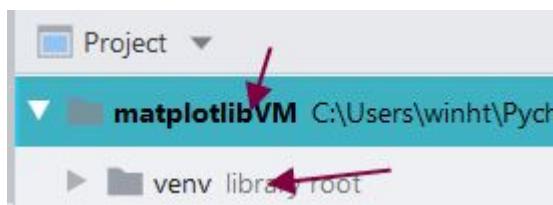
### Download PyCharm



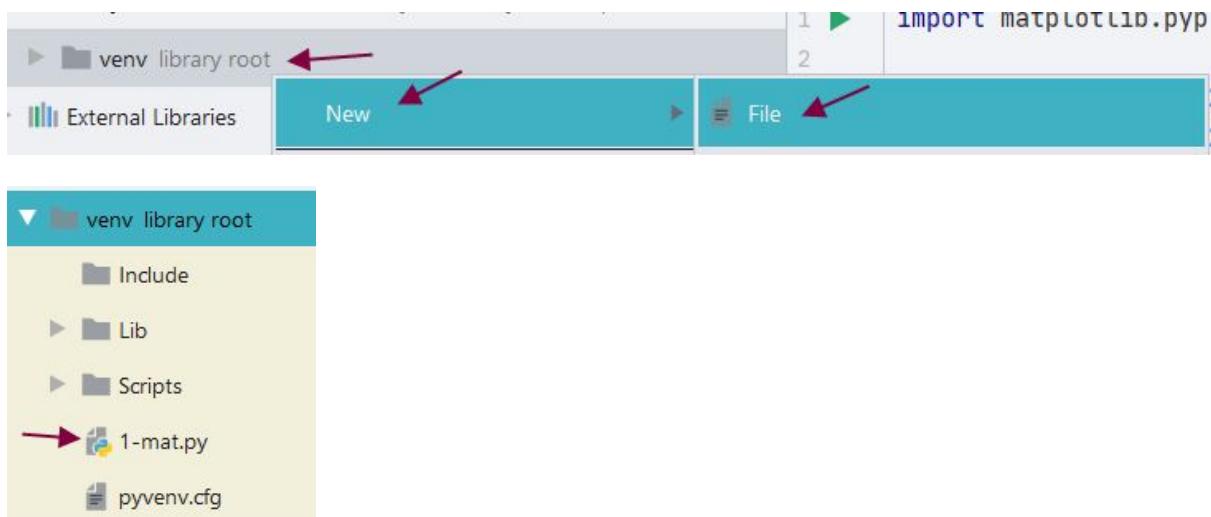
Pycharm ကို download လုပ်ပြီးသည့်နှင့် တစ်ပြိုင်နက် install လုပ်ပြီးပါက အောက်ပါ ပုံမှ အတိုင်း file >> new project သို့သွားပြီး မိမိတဲ့ ပေးချင်သည့် project name ကို ပေးပါ။ စာရေးသူ အနေဖြင့် matplotlibVM ဟုပေးလိုက်ပါသည်။



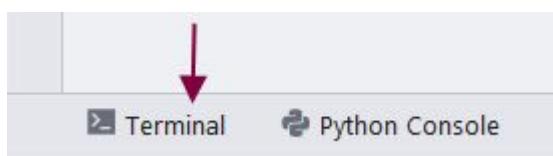
ထိုနောက် matplotlibVM ဆိုသည့် project တစ်ခုကို ဖန်တီးပေးပါလိမ့်မယ် ထို matplotlibVM ထဲထို့ ဝင်ကြည့်လျှင် အောက်ပါ အတိုင်း venv ဆိုသည့် virutal envrioment တစ်ခု တည်ဆောက်ထားတာကို တွေ့ရပါမည်။



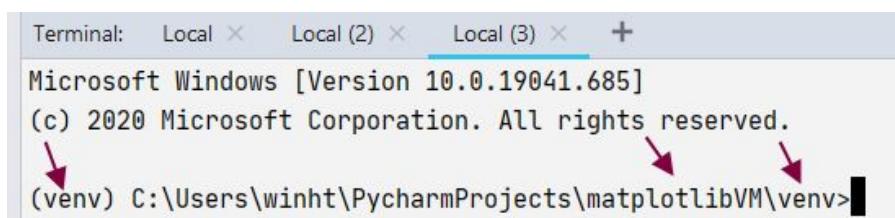
ထို venv အား right click ဆောက်ပြီး new မှ တစ်ဆင့် File သို့သွားပြီး 1-mat.py ဆိုသည့် python file တစ်ခုကို တည်ဆောက်လိုက်ပါသည်။



အထက်ပါ အဆင့်များ ပြီးပါက venv အား right click တစ်ခါ ထပ်ဆောက်ပြီး open in terminal ဆိုတာကို နှိပ်လိုက်ပါ ထို့နောက် terminal ပေါ်လာမည် ဖြစ်ပါသည် ယရ နည်းအတိုင်း အဆင်မပြုပါက ဘယ်ဘက် အောက်ဒေါင်တွင် Terminal ကို နှိပ်ပြီးတော့ terminal ကို အသုံးပြု နိုင်ပါသည်။



Terminal သို့ ရောက်သွားပါက အောက်ပါ ပုံအတိုင်း venv ထဲတွင် ရောက်နေသည်ကို မြင်ရပါမည်။ ထို virtual environment ထဲတွင် မိမိတို့ အသုံးပြုမည့် library များကို သီးသန် install လုပ်သွားမှာ ဖြစ်ပါတယ်။



ပထမဆုံး အနေဖြင့် matplotlib ကို install လုပ်ပေးရန် လိုအပ်ပါသည်။ pip install matplotlib ဆိုလျှင် matplotlib ရဲ့ နောက်ဆုံး version ကို install လုပ် ပေးမှာ ဖြစ်ပါတယ်။ install လုပ်ပြီးတဲ့ အခါမှာ မိမိတို့ install လုပ်ထားသည့် package များကို ပြန်ကြည့် လိုလျှင် pip list နှင့် ကြည့်နိုင်ပါသည်။ pip သည်

Pip installs Packages , Pip install Python သိမဟုတ် preferred installer program ကို ထိလိုခင်:  
ဖြစ်သည်။

## Packages Management

စာရေးသူတို့ လက်ရှိ ရှိနေသည့် matplotlibVM project ထဲမှ venv ထဲတွင် အောက်ပါ packages များ ရှိနေပါ။ စာရေးသူ အနေဖြင့် matplotlib version 3.3.3 နှင့် numpy version 1.19.4 တို့ကို installလုပ်ထားပါသည်။

```
(venv) C:\Users\winht\PycharmProjects\matplotlibVM\venv>pip list
```

Package	Version
cycler	0.10.0
kiwisolver	1.3.1
matplotlib	3.3.3 ←
numpy	1.19.4 ←
Pillow	8.0.1
pip	20.3.3
pyparsing	2.4.7
python-dateutil	2.8.1
setuptools	51.1.0
six	1.15.0

အကယ်၍ package version အလိုက် အတိအကျ install လုပ်လို ပါကအောက် အတိုင်း ပြုလုပ်နိုင်ပါသည်။ pip install matplotlib==3.3.3

```
(venv) C:\Users\winht\PycharmProjects\matplotlibVM\venv>pip install matplotlib==3.3.3  
Requirement already satisfied: matplotlib==3.3.3 in c:\users\winht\pycharmprojects\matplot
```

ထို့ပြင် မိမိတို့ install လုပ်ထားပြီးသား package ကို upgrade လုပ်လိုပါကလည်း pip install --upgrade Package name ဟုရေးပြီးပြုလုပ် နိုင်ပါသည်။

```
(venv) C:\Users\winht\PycharmProjects\matplotlibVM\venv>pip install --upgrade matplotlib  
Requirement already satisfied: matplotlib in c:\users\winht\pycharmprojects\matplotlibvm\venv\
```

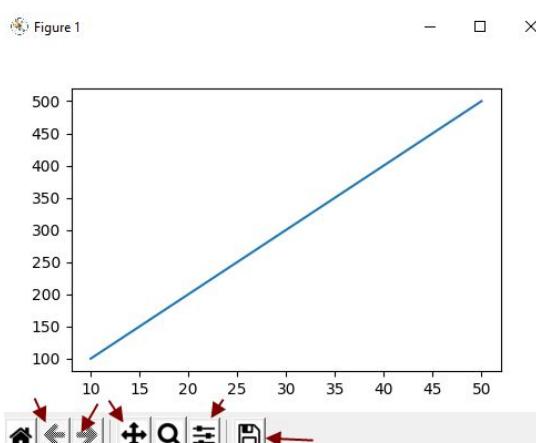
Install လုပ်ထားပြီးသား packages များကို uninstall လုပ်လို ပါကလည်း pip uninstall packagename ဟုရေးပြီး ဥပမာ pip uninstall matplotlib ယခု အတိုင်း ရေးပြီး ပြုလုပ် နိုင်ပါသည်။ pip နဲ့ ပတ်သက်ပြီး နောက်ဆုံး မှတ်ထားဖို့ လိုတဲ့ တစ်ခု ကတော့ show ဖြစ်ပါတယ် သူကတော့ package တစ်ခုရဲ့ detail အချက် အလက် အားလုံးကို ပြ ပေးနိုင်ပါတယ်။

```
(venv) C:\Users\winht\PycharmProjects\matplotlibVM\venv>pip show matplotlib
Name: matplotlib
Version: 3.3.3
Summary: Python plotting package
Home-page: https://matplotlib.org
Author: John D. Hunter, Michael Droettboom
Author-email: matplotlib-users@python.org
License: PSF
Location: c:\users\winht\pycharmprojects\matplotlibvm\venv\lib\site-packages
Requires: python-dateutil, cycler, numpy, pillow, kiwisolver, pyparsing
Required-by:
```

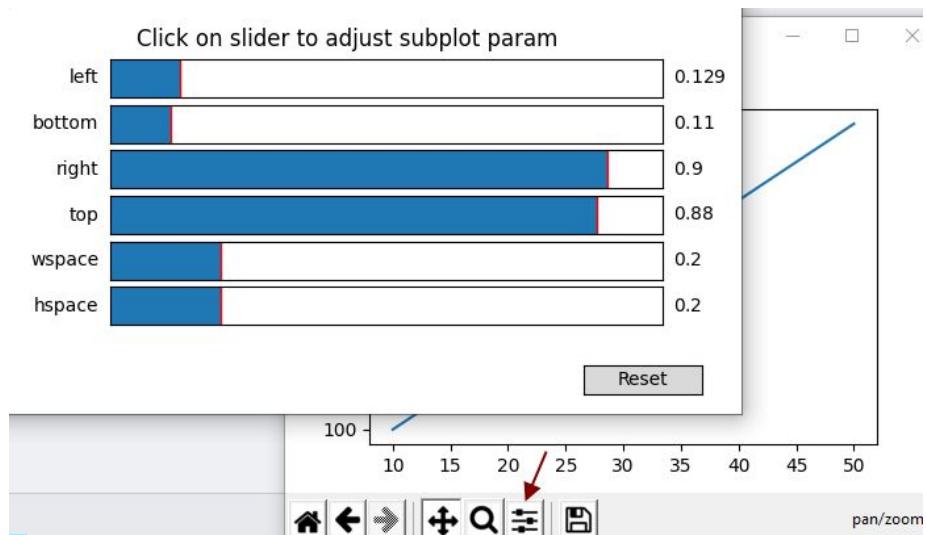
စာရေးသူတို့ matplotlibVM project ထဲမှ 1-mat.py ဆိုသည့် python file ထဲကို ပြန်သွားပြီး matplotlib သင်ခန်းစာများ စရေးကြည့်ကြပါနဲ့။ ပထမဆုံး အနေဖြင့် matplotlib ထဲမှ pyplot ကို import လုပ်ပါမည်။ ထို့ပြင် list နှစ်ခု တည်ဆောက်ပြီး ထို list နှစ်ခုတဲ့မှ အချက်လက် များကို plot ဖြင့် ပြန် ပြပါမည်။ program ရေးသား ပုံမှာ အောက်ပါ အတိုင်း ဖြစ်သည်။

```
1 ➤ import matplotlib.pyplot as plt
2
3 xData = [10,20,21,24,26,29,30,34,39,50]
4 yData = [100,200,210,240,260,290,300,340,390,500]
5 plt.plot(xData,yData)
6 plt.show()
```

Data များကို visualization လုပ်ဖြနိုင်စွဲ method နှစ်ခုကို အသုံးပြုရပါသည် ပထမ တစ်ခုသည် plot ဖြစ်ပြီး သူကတော့ မိမိတို့ plot လုပ်လိုသည့် data များကို parameter အနေဖြင့် ထည့်ပေးရပါတယ်။ ဒုတိယ တစ်ခုကတော့ show ဖြစ်ပါတယ် သူကတော့ plot လုပ်လို တဲ့ အခါတိုင်း အောက်ဆုံးတွင် မပါ မဖြစ်ပါရမယ့် function တစ်ခု ဖြစ်ပါတယ်။ အထက်ပါ program ကို run ကြည့်မည် ဆိုလျှင် အောက်ပါ အတိုင်း output များကို တွေ့ရမှာပါ။ program run နဲ့ အတွက် line1 ဘေးမှ အစိမ်းရောင်လေးကို နှိပ်ပြီး run နိုင်သည်။



Program run လိုက်သည်နှင့် figure ကို ဖော်ပြန့် windows form တစ်ခု ကျလာပါသည်။ ထို windows form တွင် အကျယ်ချုပြုး ကြည့်ရှု နိုင်ရန် left button pans and right button zoom ဆိုသည့် ဓလုပ်ကို နှိပ်ပြီးကြည့်ရှု နိုင်သည်။ ကြည့်ရှု ပြီးသည့် အခါ မူလ နေရာဘူး ပြန်ရောက် နိုင်ရန် အတွက်လည်း Home button တစ်ခု ပါဝင် နေပါသေးသည်။ ထိုပြင် မိမိတို့ အသေးစိတ် ကြည့်လိုသည့် နေရာကို ကြည့်ရှု နိုင်သည်။ Figure ကို မိမိတို့ အဆင်ပြေသလို ကြည့်ရှုနိုင်ဖို့ အတွက်လည်း Configure subplots ဆိုသည့် button တစ်ခု ပါဝင်သည့် ထို button ကို နှိပ်လိုက်လျှင် အောက်ပါ အတိုင်း Figure တစ်ခု ပေါ်လာမည် ဖြစ်ပြီး မိမိတို့ စိတ်ကြိုက် ပုံစံ ကို ချိန်းနိုင်သည်။



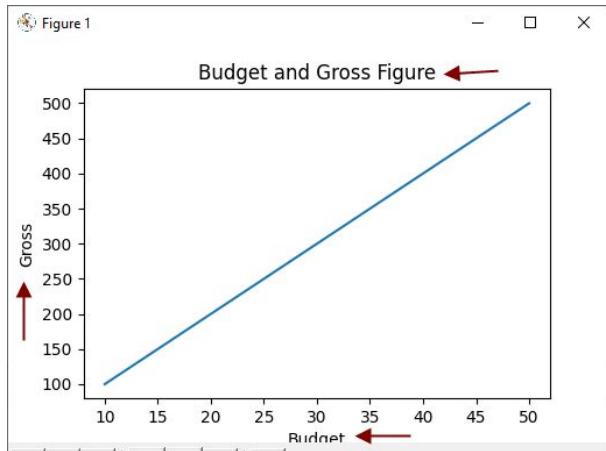
နောက်ဆုံး တစ်ခု အနေနဲ့ ကတော့ data visualize လုပ်ပြီးရလာတဲ့ အနေအထားကို image တစ်ခု အနေဖြင့် သိမ်းလိုပါက save ဆိုတဲ့ button တစ်ခု ပါဝင်နေပါသေးတယ် ထို save button ကို နှိပ်ပြီး png image file type အနေဖြင့် save နိုင်ပါသေးတယ်။ နောက်တစ်ဆင့် အနေဖြင့် figure ကို x axis အတွက် xlabel နှင့် y axis အတွက် ylabel ထိုပြင် figure အတွက် title တို့ကို ထည့်ပေးပါမည်။

```

5     plt.plot(xData,yData)
6     plt.xlabel("Budget") ←
7     plt.ylabel("Gross") ←
8     plt.title("Budget and Gross Figure") ←
9     plt.show()

```

အထက်တွင် ဖော်ပြထားသော line 6 , 7 , 8 တို့ကို ထည့်လိုက်ပါက output တွင် label များပါလာသည်ကို မြင်ရပါမည်။



### Matplotlib Example Project

Matplotlib ကို ပိုမို နားလည် နိုင်ဖို့ အတွက် project တစ်ခု ရေးကြည့်ပါမည် ယခု project တွင် x and y နှစ်ခုတည်းသာ ရှိမည် မဟုတ်ပဲ x တစ်ခုနှင့် y အများကြီး ထားပါမည်။ ယခု project တွင် လအလိုက် လုပ်ငန်းများ၊ ဝင်ငွေများ အကြောင်းကို Figure နဲ့ ဖော်ပြု သွားမှာ ဖြစ်ပါတယ်။ နှစ်တစ်နှစ်တွင် လများသည် တစ်ခု သာ ရှိသည့် အတွက် x ဝင်ရှိုးတွင် ထားပါမည်။ ဝင်ငွေဝင်ရာ လုပ်ငန်း များသည် တစ်ခုထက် ပိုမိုင်တဲ့ အတွက် y ဝင်ရှိုးတွင် ထားပါမည်။ စားရေးသူ အနေဖြင့် လုပ်ငန်း သုံးခုကို ထည့်ထားမှာ ဖြစ်တဲ့ အတွက် y ဝင်ရှိုးတွင် လုပ်ငန်း သုံးခုစာ data များကို ဖော်ပြသွားမှာ ဖြစ်ပါတယ်။

လုပ်ငန်းသုံးခုမှ ဝင်ငွေ အတွက် list 3 ရှိနှင့် month များ အတွက် list တစ်ခု စုစုပေါင်း list 4 ရှိ ရှိပါမည်။

```
import matplotlib.pyplot as plt
```

```
month = [1,2,3,4,5,6,7,8,9,10,11,12]
```

```
income_job1=[1500000,2000000,1000000,3000000,4000000,500000,1000000,1500000,350000  
0,5000000,2000000,10000000]
```

```
income_job2=[1000000,2000000,2500000,2000000,5000000,500000,2000000,1000000,200000  
0,3000000,5000000,4000000]
```

```
income_job3=[100000,50000,80000,1000000,150000,200000,2500000,3000000,3400000,4000  
000,4500000,5000000]
```

ထို Data လေး အခု အတွက် Plot လုပ်ရမည့် ပုံမှာ အောက်ပါ အတိုင်း ဖြစ်သည်။

```

1 import matplotlib.pyplot as plt
2 month = [1,2,3,4,5,6,7,8,9,10,11,12]
3 income_job1=[1500000,2000000,1000000,3000000,4000000,500000,1000000,1500000,3500000,5000000,2000000,10000000]
4 plt.plot(month , income_job1,color='red',linestyle='--',label='JOB1')
5
6 income_job2=[1000000,2000000,2500000,2000000,5000000,500000,2000000,1000000,2000000,3000000,5000000,4000000]
7 plt.plot(month , income_job2,label='JOB2')
8
9 income_job3=[100000,50000,80000,100000,150000,200000,250000,300000,3400000,4000000,4500000,5000000]
10 plt.plot(month , income_job3,color='g',linestyle='-.',label='JOB3')

```

Line 4 တွင် month data နှင့် income\_job1 အား plot တစ်ခု လုပ်ပြီး line 7 တွင်လည်း month data နှင့် income\_job2 တို့အား plot ထပ်လုပ်ထားပါသည်။ ထိုပြင် line 10 တွင်လည်း month data နှင့် income\_job3 တို့ကို plot လုပ်ထားပါသည်။ ရှေ့မှ ရေးသည့် data သည် X axis အတွက် ဖြစ်သောကြောင့် Plot လုပ်တိုင်း တွင်လည်း x axis အတွက် month data ကို ရှေ့မှ ထည့်ထားပါသည် ထို့ကြောင့် Output အနေဖြင့် x axis တွင် month data တစ်ခုတည်းနှင့် y axis အတွက် income\_job data သုံးခုလုံးကို ရရှိမှာ ဖြစ်ပါတယ်။

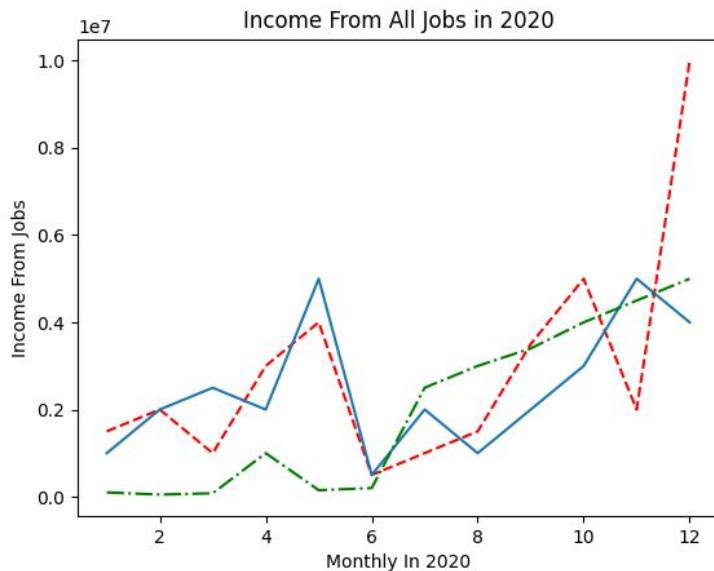
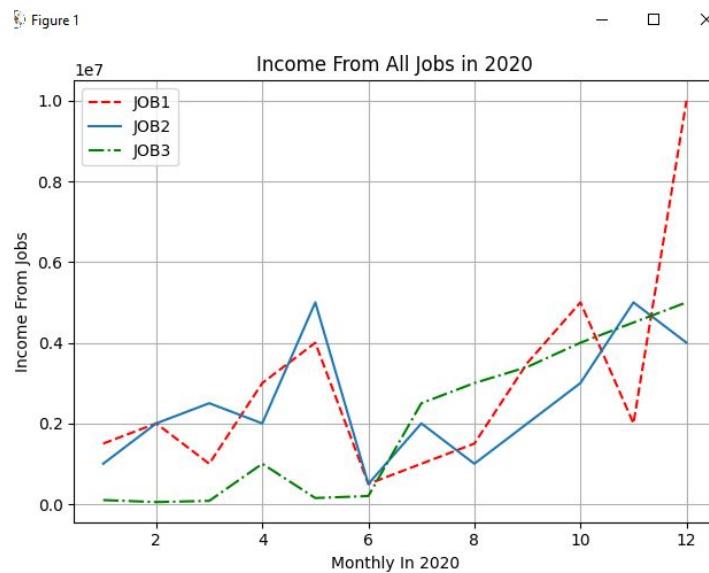


Figure ထဲမှ data လိုင်းများကို မိမိတို့ အသုံးပြုလိုသည့် color , style နှင့် label များ ထိုးပြီး အသုံးပြန်ပါသည်။ line 7 တွင် color အတွက် red ကို သုံးထားပြီး linestyle အတွက် ‘ -- ’ ကို သုံးထားပါသည်။ label properties သည် figure ထဲတွင် ဘယ် လိုင်း အမျိုးအစားသည် ဘယ် data အမျိုးအစား ဖြစ်သလဲ ဆိတာကို ဖော်ပြန့် အတွက် အသုံးပြုပါသည်။ စာရေးသူ အနေဖြင့် income\_job1 နှင့် income\_job3 အတွက်သာ color နှင့် style တို့ကို သုံးထားပါသည်။ income\_job2 တွင်ပေါ်နေသော color နှင့် style သည် default သာ ဖြစ်ပါသည်။ အထက်ပါ figure ထဲတွင် label များ မပေါ်သေးပါဘူး ထို့ကို ပေါ်နိုင် အတွက်ဆိုလျှင် legend() ဆိုသည့် method ကို အသုံးပြု ပေးရမှာ ဖြစ်ပါတယ်။ ထိုပြင် grid အကွက်များ ထည့်လို ပါကလည်း grid() method ကို သုံးနိုင်သလို xlabel , ylabel များ အတိအကျ ပြည့်စုစွာ figure တွင် ပေါ်နိုင်ရန် အတွက်လည်း tight\_layout() ဆိုသည့် method ကို အသုံးပြုရပါမည်။ ထို method များ အသုံးပြုပုံကိုအောက်ပါ ပုံတွင် ဖော်ပြထားသည်။

```

3   month = [1,2,3,4,5,6,7,8,9,10,11,12]
4   income_job1=[1500000,2000000,1000000,3000000,4000000,500000,1000000,1500000,3500000,5000000,2000000,10000000]
5   plt.plot(month , income_job1,color='red',linestyle='--',label='JOB1')
6
7   income_job2=[1000000,2000000,2500000,2000000,5000000,500000,2000000,1000000,2000000,3000000,5000000,4000000]
8   plt.plot(month , income_job2,label='JOB2')
9
10  income_job3=[100000,50000,80000,1000000,150000,200000,2500000,3000000,3400000,4000000,4500000,5000000]
11  plt.plot(month , income_job3,color='g',linestyle='-.',label='JOB3')
12
13  plt.xlabel('Monthly In 2020')
14  plt.ylabel('Income From Jobs')
15  plt.title('Income From All Jobs in 2020')
16  plt.legend() ←
17  plt.grid(True) ←
18  plt.tight_layout() ←
19  plt.show()

```



Trend lines များ၏ styles များကိုတော့ မိမိတို့ လိုအပ်သလို အသုံးပြုနိုင်ရန် အောက်တွင်ဖော်ပြထားသည်။

#### Character Description

- '-' solid line style
- '--' dashed line style
- '-.-' dash-dot line style
- :: dotted line style
- .' point marker
- ,' pixel marker
- 'o' circle marker
- 'v' triangle\_down marker
- '^' triangle\_up marker
- '<' triangle\_left marker

```
'>' triangle_right marker
'1' tri_down marker
'2' tri_up marker
'3' tri_left marker
'4' tri_right marker
's' square marker
'p' pentagon marker
'*' star marker
'h' hexagon1 marker
'H' hexagon2 marker
'+' plus marker
'x' x marker
'D' diamond marker
'd' thin_diamond marker
'|' vline marker
'-' hline marker
```

Trend line များ၏ color များ အတွက် ဆို အောက်ပါ colors များကို အသုံးပြု နိုင်သည်။

character color

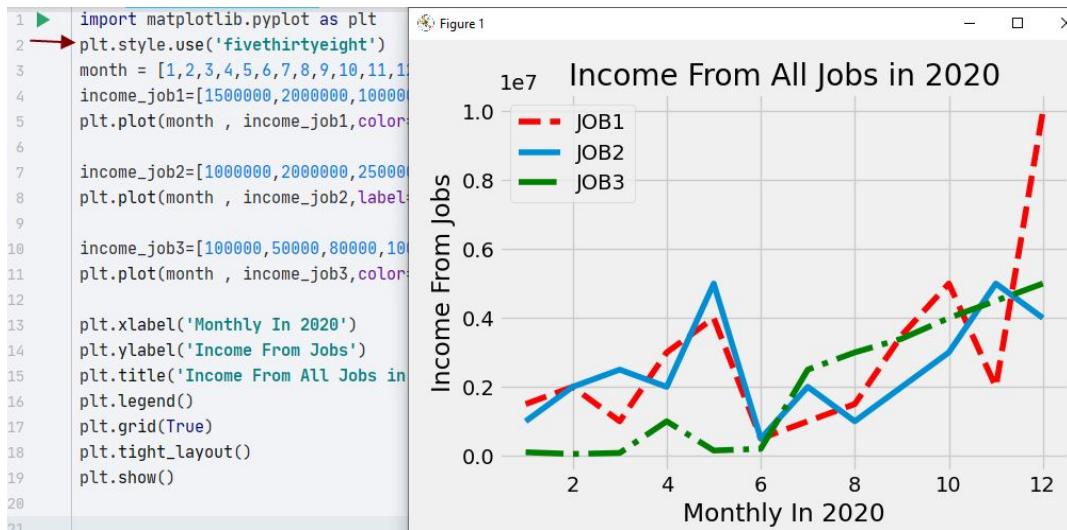
```
'b' blue
'g' green
'r' red
'c' cyan
'm' magenta
'y' yellow
'k' black
'w' white
```

ထို့ပြင် figure ရဲ့ style ကိုလည်း မိမိတို့ လိုအပ်သလို ပြောင်းလဲ နိုင်ရင် style များစွာ ရှုပါသေးသည်။  
styles များကို သိနိုင်ရင် print(plt.style.available) ဟုရေးပြီး run ကြည့်ပါက သိနိုင်သည်။အောက်တွင် ဖော်ပြထားသော style များ ဖြင့် မိမိတို့၏ figure style ကို အသုံးပြု နိုင်ပါသည်။

```
['Solarize_Light2', '_classic_test_patch', 'bmh', 'classic', 'dark_background', 'fast',
'fivethirtyeight', 'ggplot', 'grayscale', 'seaborn', 'seaborn-bright', 'seaborn-colorblind',
'seaborn-dark', 'seaborn-dark-palette', 'seaborn-darkgrid', 'seaborn-deep',
```

```
'seaborn-muted', 'seaborn-notebook', 'seaborn-paper', 'seaborn-pastel', 'seaborn-poster',
'seaborn-talk',         'seaborn-ticks',        'seaborn-white',       'seaborn-whitegrid',
'tableau-colorblind10']
```

စာရေးသူ အနေဖြင့် အောက်ပါ program တွင် fivethirtyeight ဆိုသည့် style ကို သုံးထားပါသည်။ အောက်ပါ program ၌ line 2 တွင်ကြည့်ပါ။



## Bar Charts

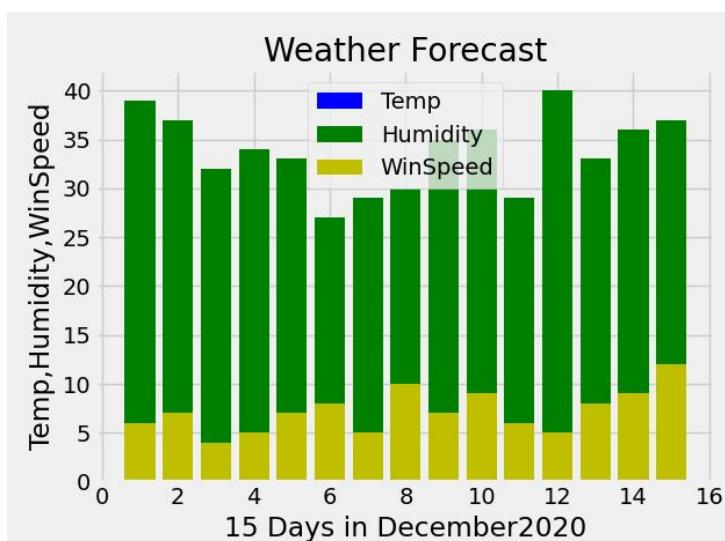
ရှေ့က သင်ခန်းစာမျာတော့ plot ကို သုံးခဲ့ကြပြီး ယခု သင်ခန်းစာမျာတော့ matplotlib ထဲက bar chart ကို အသုံးပြုသွားပါမည်။ Bar chart သည် X axis data တစ်ခုတည်းနှင့် Y axis data အမျိုးအစား များစွာတို့ကို အတူနှိုင်းယူ၍ ပြန့် အဆင်ပြုပါတယ်။ ဥပမာ December 1 ရက်နေ့မှာ Temperature ဘယ်လောက် Humidity ဘယ်လောက် , လေတိက်နှုန်း ဘယ်လောက် ရှိတယ် စတဲ့ အချက်တွေကို နှင့် ယူ၍ကြည့်လို ရပါတယ်။ X axis ကတော့ December 1 ဖြစ်ပြီးကျန်သည့် data များမှာတော့ Y axis ဖြစ်ပါတယ်။ Bar Chart ကို အသုံးပြုပုံမှာ plt.plot မှ plot နေရာတွင် bar ကို အသုံးပြုရှုသာ ဖြစ်ပါတယ်။ ရေးသား ထားပုံကို အောက်ပါ program တွင် လေ့လာ ကြည့်ပါ။

```

1 import matplotlib.pyplot as plt
2
3 plt.style.use("fivethirtyeight")
4 #Day for december 2020 in Pyin Oo Lwin
5 decDays =[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15]
6 #temp in C
7 Temp=[16,15,10,9,14,20,25,16,15,10,9,14,20,25,17]
8 #Humidity is the concentration of water vapor present in the air
9 Humidity=[39,37,32,34,33,27,29,30,35,36,29,40,33,36,37]
10 #wind speed in mile per hour/mph
11 WinSpeed=[6,7,4,5,7,8,5,10,7,9,6,5,8,9,12]
12
13 plt.bar(decDays,Temp,color='b',label='Temp')
14 plt.bar(decDays,Humidity,color='g',label='Humidity')
15 plt.bar(decDays,WinSpeed,color='y',label='WinSpeed')
16
17 plt.xlabel('15 Days in December2020')
18 plt.ylabel('Temp, Humidity, WinSpeed')
19 plt.title('Weather Forecast')
20 plt.legend()
21 plt.grid(True)
22 plt.tight_layout()
23 plt.show()

```

အထက်ပါ program တွင် Y axis ပုံ Temp , Humidity and WinSpeed စသည့် Data သုံးခုကို  
သုံးထားလော်လည်း output တွင် Data နှစ်မျိုးတည်းရဲ့ bar ကို သာ ပြပေးပါလိမ့်မယ်။  
ဘာကြောင့်လည်းဆိုးတော့ bar method သည် parameter တွင် bar( x , height , width , bottom ,  
align ) ထိုပုံစံ အတိုင်း သွားပါတယ်။



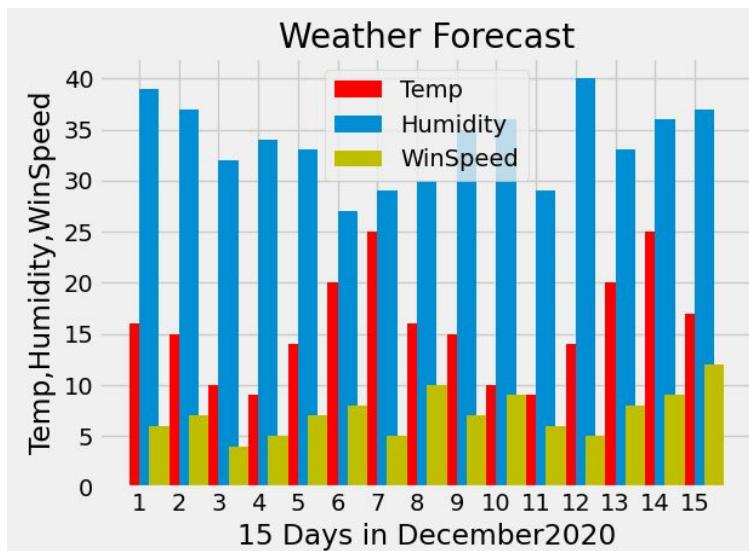
X parameter နေရာသည် x axis အတွက်ဖြစ်ပြီး ထို x axis မှာ data များစွာကို ပြချင်လျှင် bar ရဲ့  
အတူ width နဲ့ တည်နေရာကို သတ်မှတ် ပေးလိုပါတယ်။ အထက်ပါ program တွင် bar တစ်ခု စီ အတွက်  
0.25 စီ သတ်မှတ် ပေးပြီး နေရာကိုပါ သတ်မှတ် ပေးထားပါတယ်။ ပထမဆုံး မြှားကတော့ bar တစ်ခုစီရဲ့

width အတူကို သတ်မှတ် ပေးတာ ဖြစ်ပြီး အောက်က မြားသုံးခုကတော့ posititon ကို သတ်မှတ် ပေးတာပါ။ ပထမဆုံး တစ်ခုတွင် + 0.0 ကို ရေးထားသည့် အတွက် အစတွင် ဖော်ပြ ပေးပြီး ဒုတိယ တစ်ခုတွင် +0.25 ဟုရေးထာသည့် အတွက် 0.25 နေရာတွင် ဖော်ပြ ပေးပါတယ်။ နောက်ဆုံး တစ်ခုသည် +0.50 ဖြစ်သည့် အတွက် 0.50 နေရာတွင် ဖော်ပြ ပေးပါတယ်။

```

1 import matplotlib.pyplot as plt
2 import numpy as np
3 plt.style.use("fivethirtyeight")
4 #Day for december 2020 in Pyin Oo Lwin
5 decDays =[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15]
6 xData = np.arange(len(decDays))
7 width = 0.50 ←
8 #temp in C
9 Temp=[16,15,10,9,14,20,25,15,10,9,14,20,25,17]
10 #Humidity is the concentration of water vapor present in the air
11 Humidity=[39,37,32,34,33,27,29,30,35,36,29,40,33,36,37]
12 #wind speed in mile per hour/mph
13 WinSpeed=[6,7,4,5,7,8,5,10,7,9,6,5,8,9,12]
14
15 plt.bar(xData+0.0,Temp,width,color='r',label='Temp')
16 plt.bar(xData+0.25,Humidity,width,color='g',label='Humidity')
17 plt.bar(xData+0.50,WinSpeed,width,color='y',label='WinSpeed')
18 plt.xticks(ticks=xData,labels=decDays)
19 plt.xlabel('15 Days in December2020')
20 plt.ylabel('Temp, Humidity, WinSpeed')
21 plt.title('Weather Forecast')
22 plt.legend()
23 plt.grid(True)
24 plt.tight_layout()
25 plt.show()
```

Line 18 မှ xticks သည် DecDays ထဲမှ data များကို x axis တွင် ပြပေးချင်သည့် အတွက် ရေးပေးထားခြင်း ဖြစ်ပါတယ်။ Details ပြောရမည် ဆိုလျှင် xticks method သည် အနည်းဆုံး parameter နှစ်ခု ယူပါတယ် ပထမ တစ်ခုသည် location ဖြစ်ပြီး ဒုတိယ တစ်ခုသည် ထို location နေရာတွင် ပေါ်ပေးမည့် Data ဖြစ်ပါတယ်။ ဥပမာ ပြရမည် ဆိုလျှင် x=[1,2,3] , data=[a,b,c] ဟု သတ်မှတ်ပေးမည် ဆိုလျှင် x axis ရဲ့ 1,2,3 နေရာတွင် a,b,c စာညွှေ့ data များ ဖော်ပြ နေမှာ ဖြစ်ပါတယ်။ xticks([0, 1, 2], ['January', 'February', 'March']) ယခုကဲ့သို့ ရေးမည် ဆိုလည်း ရပါတယ်။ Program output ကိုတော့အောက်မှာ ဖော်ပြထားပါတယ်။



## Bar Charts and CSV

ယခု သင်ခန်းစာမျာတော့ CSV မှ data တွေကို ရယူပြီး bar chart နဲ့ ပြန် ပြ လေးမှာ ဖြစ်ပါတယ်။ libraries များ အနေဖြင့် pandas , collections , matplotlib တို့ကို အသုံးပြုသွားပါမည်။ စာရေးသူတို့ အနေဖြင့် ယခင် matplotlibVM ဆိုသည့် project ထဲတွင်ပင် ရေးသားသွားမှာ ဖြစ်တဲ့ အတွက် ထပ်မံလိုအပ် နေ့သေးသည့် pandas and collections library များကို install လေးရန် လိုအပ်ပါသည်။

Package	Version
collection	0.1.6
cycler	0.10.0
kiwisolver	1.3.1
matplotlib	3.3.3
numpy	1.19.4
pandas	1.2.0
Pillow	8.0.1
pip	20.3.3
pyparsing	2.4.7
python-dateutil	2.8.1
pytz	2020.5
setuptools	51.1.0
six	1.15.0

Libraries များအားကို install လုပ်ပြီးနောက် လိုအပ်သည့် library များကို import လုပ်ပါမည် ထိုနောက် csv file အား ဖတ်ပါမည်။ ထိုကြောင့် အောက်ပါ program အား စတင် ရေးသား ပါသည်။

```

2 import pandas as pd
3 from collections import Counter
4 from matplotlib import pyplot as plt
5
6 plt.style.use("fivethirtyeight")
7
8 data = pd.read_csv('ProgrammingUsed.csv')
9 ids = data['Id'] ←
10 lang_responses = data['UsedLanguages'] ←

```

နောက် တစ်ဆင့် အနေဖြင့် counter function တိုအ အသုံးပြုမှာ ဖြစ်တဲ့ အတွက် counter function အကြောင်းကို အသေးစိတ် ဦးစွာ ရှင်းလိပါသည်။ Counter သည် input အနေဖြင့် list,tuple , dictionary , string စသည့် input များဖြစ်တဲ့ iterable objects များ ကို ရယူပြီး output အနေဖြင့် ထို iterable object တွေထဲမှာ ပါဝင်တဲ့ elements အရေအတွက် တိုကို count လုပ်ပြီး ပြန်လေးပါတယ်။ စာရေးသတ္တု အနေဖြင့်လည်း ယခု projects မှာ csv ထဲမှ data တွေကို count လုပ်ချင်ပါတယ် ဥပမာ javascript အသုံးပြုသူ ဘယ်နှစ်ယောက် ရှိလဲ ဆိတာမျိုးပါ။ sample data နဲ့ ပြရမည် ဆိုလျှင် `['x','y','z','x','x','y', 'z','x','b','w']` data ယခု ဖော်ပြထားတဲ့ data တွေထဲမှာ x ကဘယ်နှစ်ခုပါသလဲ y က ဘယ်နှစ်ခု ပါသလဲ စသဖြင့် ရေတွက်ပြီး ခွဲထုတ် ပေးပါတယ် ထိုသို့ ခွဲထုတ်ပြီး ရလာတဲ့ data တွေကိုလည်း key and value ပုံစံဖြင့် ပြန်ပြီး ပေးပါတယ်။

```

1 ► from collections import Counter
2 myIter= ['x','y','z','x','x','y', 'z','x','b','w']
3 print(Counter(myIter))
4 #output
5 #Counter({'x': 5, 'y': 2, 'z': 2, 'b': 1, 'w': 1})

```

ထို့ပြင် lang\_responses ထဲမှ data များကို ထုတ်ကြည့်သည့် အဓိကအကိပါ အတိုင်း data များစွာကို ရှုပါတယ် ထို data များရဲ့ တစ်ခုနှင့်တစ်ခု ကြားတွင် semicolon များ မြား နေသည်ကို တွေ့ရပါမည်။ ထို semicolon များကို ဖျက်ထုတ်ရန် လိုအပ် နေပါသေးသည်။

```

9 lang_responses = data['UsedLanguages']
10
11 language_counter = Counter()
12
13 for response in lang_responses:
14     print(response)

```

## Output

```

Bash/Shell/PowerShell;C++;Python;Ruby;Other(s):
C;C++;Other(s):
Bash/Shell/PowerShell;C++;HTML/CSS;JavaScript
Bash/Shell/PowerShell;HTML/CSS;Python;Other(s):
Assembly;C++;HTML/CSS;VBA
C;C++;Java
Assembly;Bash/Shell/PowerShell;C;C#;HTML/CSS;Java;JavaScript;PHP;SQL;Other(s):
C;C++;HTML/CSS;JavaScript;PHP;Python;SQL
HTML/CSS;Java;JavaScript
HTML/CSS;JavaScript;Python
Bash/Shell/PowerShell;Go;HTML/CSS;JavaScript;WebAssembly
HTML/CSS;JavaScript;Other(s):
Bash/Shell /PowerShell ;HTML /CSS ;Java ;JavaScript ;PHP ;SQL ;Other(s) :

```

ထို data များ ကြားတွင်မြားနေသော semicolon များကို ဖျက်ထုတ်နိုင်ရန် split function ကို အသုံးပြု နိုင်ပါသည် split function ထဲတွင် parameter အနေဖြင့် ; ကို ထည့်ပေးလိုက်မည် ဆိုလျှင် ထို ; များ ကို ဖျက်ထုတ် ပေးမည် ဖြစ်ပြီး output အနေဖြင့် list ပုံစံကို ပြန်လည် ရရှိမှာ ဖြစ်ပါတယ်။

```

12
13     for response in lang_responses:
14         print(response.split(';'))
15
16     ['Bash/Shell/PowerShell', 'Go', 'HTML/CSS', 'JavaScript', 'WebAssembly']
17     ['HTML/CSS', 'JavaScript', 'Other(s):']
18     ['Bash/Shell/PowerShell', 'HTML/CSS', 'Java', 'JavaScript', 'PHP', 'SQL', 'Swift']

```

နောက်တစ်ဆင့် အနေဖြင့် List ပုံစံနဲ့ ရရှိလာသော data များအား ရောက် ရန် လိုအပ်ပါသေးသည်။ ထို့ကြောင့် counter function ကို အသုံးပြုပါမည် အထက်တွင် ဖော်ပြ ခဲ့သည့် အတိုင်း counter function သည် သူ့ထဲသို့ ထည့်ပေးလိုက်သော data များအား ရောက်ပြီး ပြန်လည် ခဲ့ထုတ် ပေးပါတယ်။ ထို သို့ ခဲ့ထုတ်ပြီး ရလာတဲ့ data တွေကို update လုပ်ဖို့ လိုပါတယ် ဥပမာ ပထမ Looping ပတ်လိုက်တယ် html တစ်ခါ ပါတယ် ဒုတိယ တစ်ခါ ထပ်ပတ်တယ် နောက် တစ်ခါ ထပ်ပါတယ် html က ဒါဆိုရင် html ချဉ်းပဲ နှစ်ခု ရှိပြုပေါ့ ထိုနည်း တူ နောက် တစ်ခါကို Looping ထပ်ပတ်ပြီး ပါလာရင်လည်း နောက်တစ်ကြိမ် ထပ်ပြီး Update ထပ်လုပ်ရမှာပါ ထိုသို့ လုပ်ဆောင် ပေးဖို့ update function ကို အသုံးပြု နိုင်ပါတယ် sample program ကို အောက်မှာ ဖော်ပြ ပေးထားပါတယ်။

```

13     for response in lang_responses:
14         language_counter.update(response.split(';'))
15

```

```
Counter({'JavaScript': 59219, 'HTML/CSS': 55466, 'SQL': 47544, 'Python': 36443, 'Java': 35917, 'Ba
```

အထက်ပါ အတိုင်း languages တွေ အားလုံးကို ရောက်ပြီးသား ရပါ လိမ့်မယ်။ ရလာတဲ့ data တွေကို ကြည့်မည် ဆိုလျှင် ရှုံးက languages တွေဖြစ်ပြီး နောက်ကတော့ languages တွေကို အသုံးပြုတဲ့ နှစ်းမြတ်ပါတယ်။ ထို့ပြင် languages အမျိုးအစားပေါင်း များစွာ ရှိတဲ့ ထဲက top 10 ဆယ်ရကိုလည်း ခဲ့ထုတ်ဖို့ လိုအပ်ပါသေးတယ်။ အကယ်၍ ထိုသို့ top 10 မပြုလိုပဲ ရှိသမျှ languages တွေ အားလုံးရဲ့ အသုံးပြုနှစ်းကို ပြည့်ဆိုလည်း ရပါတယ်။

```

13     for response in lang_responses:
14         language_counter.update(response.split(';'))
15
16     languages =[] # to store languages
17     rate =[] # to store using times of languages
18
19     for i in language_counter.most_common(10):
20         languages.append(i[0])
21         rate.append(i[1])
22
23     print(languages,rate)

```

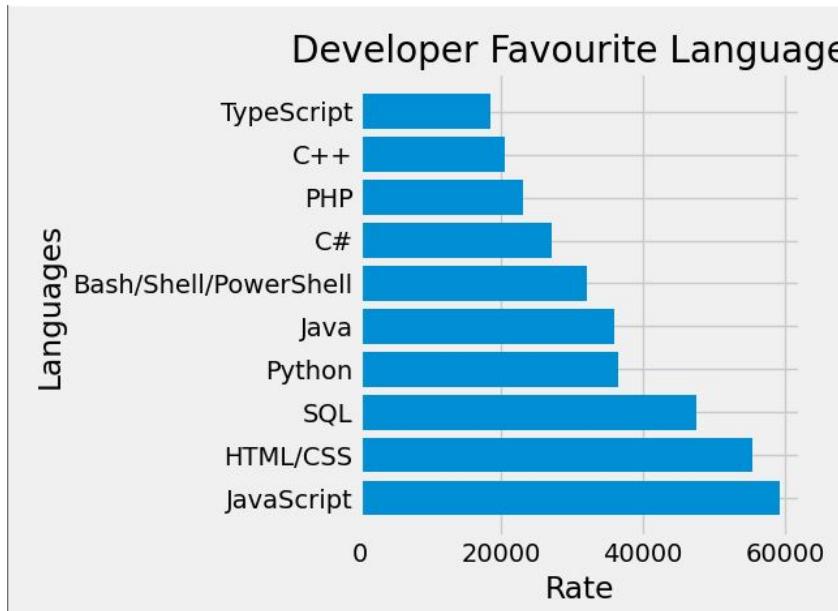
Line number 16 and 17 ကတော့ languages တွေနဲ့ ထို languages တွေကို အသုံးပြုတဲ့ နှစ်းတွေကို ခွဲပြီး သိမ်းထားဖို့ list နှစ်ခု တည်ဆောက်လိုက်ခြင်း ဖြစ်ပါတယ်။ line 19 မှာတော့

အသုံးအများဆုံး 10 ခုကိုသာ လိုချင်သည့် အတွက် most\_common(10) ကို အသုံးပြု လိုက်ခြင်း ဖြစ်ပါတယ်။ line 19 မှာ Looping တစ်ကြိမ် ပတ်လိုက်တိုင်း ထွက်လာတဲ့ data ထဲက language ကို languages ဆိုတဲ့ list ထဲကို ထည့်ပြီး ခုတိယ တစ်ရုဖြစ်တဲ့ အကြိမ်အရေ အတွက်ကိုတော့ rate ဆိုတဲ့ list ထဲ ထည့်ပါတယ် ထို Looping ပြီးဘားတဲ့ အချိန် အသုံးအများဆုံး ဆယ်ခုရဲ့ language နဲ့ rate ကို ခွဲထားပြီးသား ဖြစ်နေပါပြီ။ ဒါမီ လိုချင်သည့် data နှစ်ခု ရပြီ ဆိုတော့ ထို data တွေကို plot လုပ်လိုပါပြီ။ program အပြည့်အစုံကိုတော့ အောက်မှာ ဖော်ပြထားပါတယ်။

```

1 ► import pandas as pd
2
3   from collections import Counter
4
5   from matplotlib import pyplot as plt
6
7
8   plt.style.use("fivethirtyeight")
9
10
11  data = pd.read_csv('ProgrammingUsed.csv')
12  ids = data['Id']
13  lang_responses = data['UsedLanguages']
14
15
16  language_counter = Counter()
17
18
19  for response in lang_responses:
20      language_counter.update(response.split(';'))
21
22
23  languages =[] # to store languages
24  rate =[] # to store using times of languages
25
26
27  for i in language_counter.most_common(10):
28      languages.append(i[0])
29      rate.append(i[1])
30
31
32  plt.barh(languages,rate)
33  plt.title("Developer Favourite Languages")
34  plt.ylabel("Languages")
35  plt.xlabel("Rate")
36  plt.tight_layout()
37  plt.show()

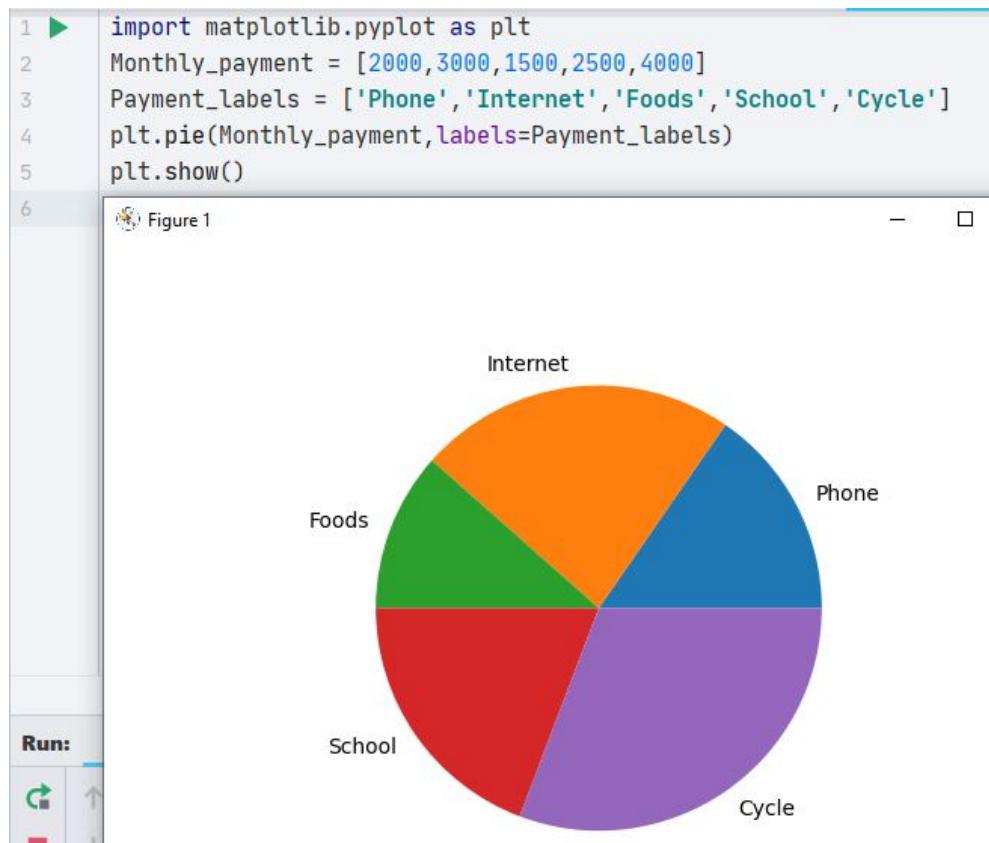
```



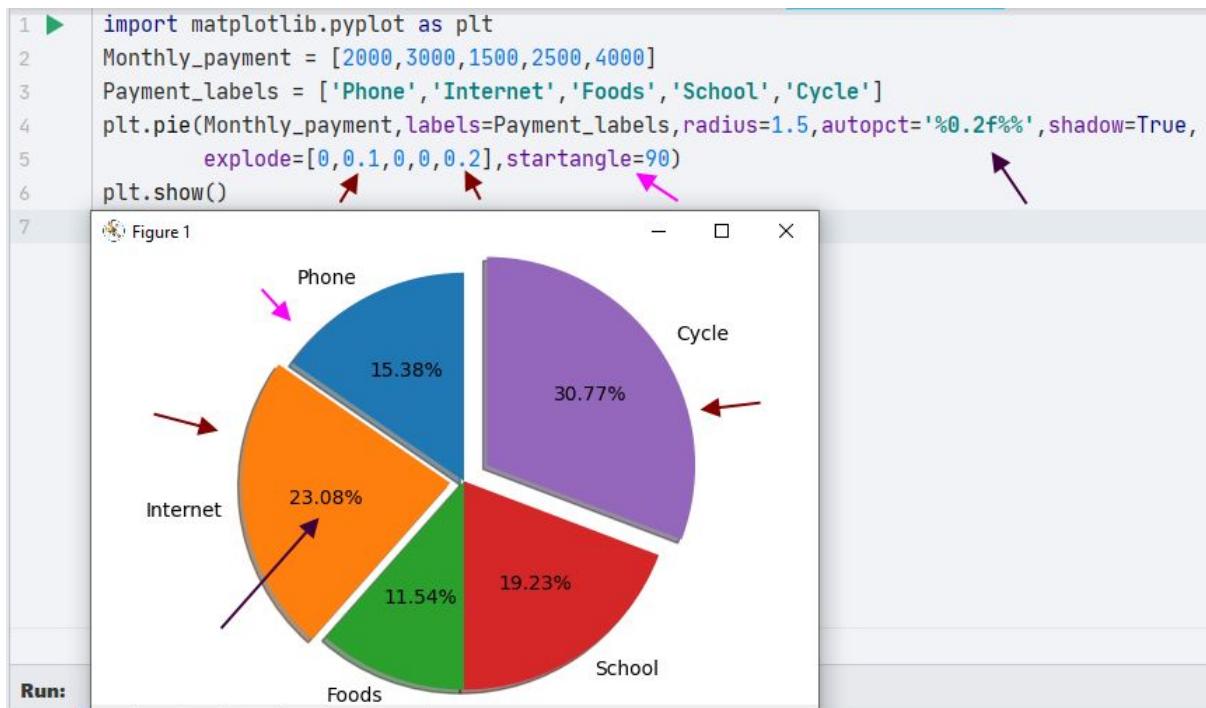
အကယ်၍ JavaScript ကိုအပေါ်ဆုံးမှာ ပေါ်စေချင်တယ် ဆိုရင် reverse function ကို အသုံးပြု နိုင်ပါတယ်။reverse function ကို line 22 တွင် languages.reverse() နှင့် line 23 တွင် rate.reverse() စသည့် code နှစ်ကြောင်းထပ်မံ ရေးသား ပေးရမှာ ဖြစ်ပါတယ်။ ထို့ပြင် ရှိသမျှ languages အားလုံး၏ အသုံးပြုမှုကို လိုချင်ရင်တော့ line 19 မှ most\_common method ကို ဖြုတ်လိုက် နိုင်ပါတယ်။

### Pie Chart

Pie Chart ဟာ စက်ရိုင်းပုံ graph တစ်ခု ဖြစ်ပြီး မတူညီတဲ့ အချက်အလက်တွေကို စုပေါင်းပြီး ပြပေးပါတယ်။ အချက်လက်တွေရဲ့ ပါဝင်တဲ့ ပမာဏပေါ် မတူတည်းပြီး data တစ်ခုနဲ့ တစ်ခုကြား ခြားနားစွာ ပြသ ပေးပါတယ်။ Pie chart ကို အသုံးပြုမယ် ဆိုရင် data ၆ ခု သို့မဟုတ် ၆ ခုထက် ပိုနည်းတာကို ပြသ သင့်ပါတယ်။ ၆ ခုထက် ပိုများသွားမယ်ဆိုရင် ကျွန်တော်တို့မျှက်လုံးကို data တစ်ခုနဲ့ တစ်ခုကြားမှန်ကန်စွာ ခွဲခြားနိုင်မှ အားနည်းလာတဲ့ အတွက်ကြောင့်ပါ။ Pie Chart ဟာလည်း အခြား chart များနည်းတူ အနည်းဆုံး list နှစ်ခု ယူပါတယ်။ Pie chart ကို အသုံးပြု ဖို့အတွက် pie ဆိုတဲ့ method အကို အသုံးပြုပါတယ်။



နောက်တစ်ဆင့် အနေဖြင့် မိမိတို့ pie chart ရဲ့ size ကိုလည်း ကြီးလို့ ရပါတယ် ဆေးတယ် ထိုသို့  
ကြိုးဖို့အတွက်ဆိုရင် radius ဆိုတဲ့ property ကိုသုံးရပါမည်။ ထိုပြင် pie chart ထဲမှာ ပါဝင်တဲ့ data တွေရဲ့  
ပါဝင်မှု အတိအကျကို percentage နဲ့ ပြလိပါက autopct property ကို သုံးနိုင် ပါတယ်။ ထိုပြင် မိမိတို့  
ပြချင်တဲ့ data ကို သီးသန့် ခွဲပြလိပါကလည်း explode property ကို သုံးနိုင်သလို data တွေကိုလည်း မိမိတို့  
ပြချင်တဲ့ နေရာအတိုင်း ရုံးနှင့်လို့ရပါသေးတယ် ဥပမာ အထက်ပါ Pie chart မှာဆိုလျှင် phone သည် 0  
degree ကနေ ပပါတယ် အကယ်၍ phone ကို 45 degree ကနေ စလိုပါက startangle ဆိုတဲ့ property  
ကိုသုံးပြီး မိမိတို့ စစေချင်တဲ့ degree ကို ထည့်ပေါ်နိုင်ပါတယ်။



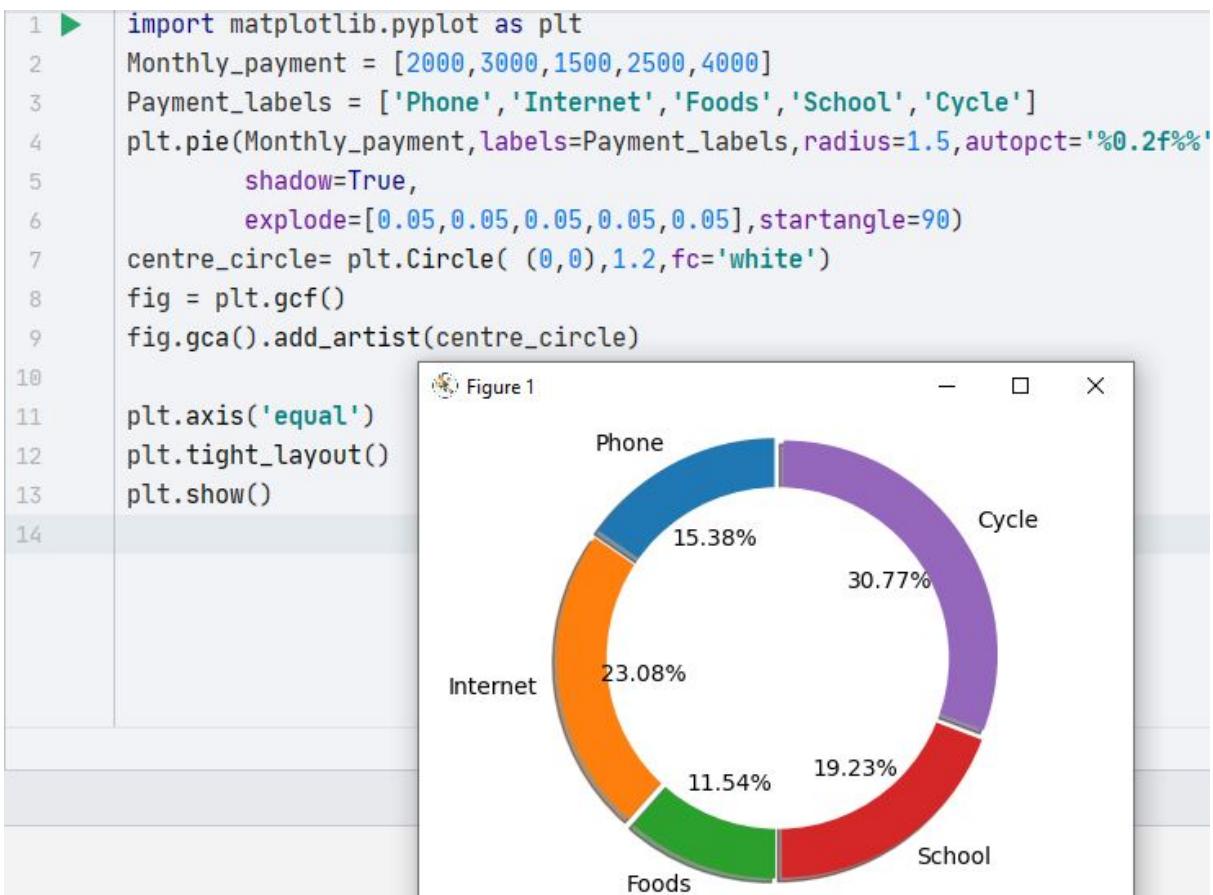
အထက်ပါ program မှာဆိုလျှင် radius အနေဖြင့် 1.5 ကို သုံးထားသည့် တွက် ယခင် program size ထက် .5 ပိုကြီးမှာ ဖြစ်ပါတယ် ဘာကြောင့်လည်း ဆိုတော့ radius ရဲ့ default size ဟာ 1 ဖြစ်ပါတယ်။ ထို့ပြင် data များကို percentage အလိုက် ပြထားပြီး ဒါသမ နောက် သူည် နှစ်လုံးကို ပြထားပါတယ် 0.2f သည် point နှစ်ခုကို ဆိုလိုခြင်း ဖြစ်ပါတယ်။ ထို့ပြင် shadow ကိုသုံးထားသည့် အတွက် pie chart ရဲ့ အောက်ဘက်ခြမ်းမှာ shadow ဝါဝင် နေတာကို မြင်ရမှာပါ။ explode ကိုသုံးရာတွင် list ပေးသော အခါ ပါဝင်သော အစိတ်အပိုင်း အရေအတွက် အတိုင်း အတိအကျ ထည့်ပေးရပါမည် သို့မှာသာ ဘယ်အပိုင်းကို explode လုပ်ချင်လဲဆိုတာ အတိအကျ သိနိုင်မှာပါ အထက်ပါ program မှာ ဆိုလျှင် list ထဲမှာ data 6 ခု ထည့်ထားပြီး ခုတိယ data နှင့် နောက်ဆုံး data ကို explode လုပ်ထားပါသည်။ ဒီနေရာမှာ တစ်ခု ရွှေပ်တွေးနိုင်တာက ခုတိယ နဲ့ နောက်ဆုံးဆိုတာကို ဘယ်လို ခွဲခြားနိုင်သလဲ။ နာရီ လက်တံပြောင်းပြန် ပုံစံဖြင့် ခွဲခြားနိုင်ပါတယ် ပထမဆုံး ပုံတွင် Phone သည် ရွှေခုံးမှ ဖြစ်ပါတယ် ထို phone data သည် 0 degree ကနေ စပါတယ်။ ခုတိယ ပုံကို ကြည့်လျှင် Phone ရဲ့ အစသည် 90 degree သို့ပြောင်းသွားပါတယ် အဘယ်ကြောင့်ဆိုလော် startangle property ကို 90 degree သို့ သတ်မှတ် ပေးလိုက်သောကြောင့် ဖြစ်သည် ထိုကြောင့် Phone သည် ပထမ ဖြစ်ပြီး Internet သည် ခုတိယ ဖြစ်သည် ထိုနည်းတူ Cycle သည် နောက်ဆုံးဖြစ်သည် ယခု ဖော်ပြချက်သည် Pie chart ကို ကြည့်ပြီးခွဲခြား ခြင်း ဖြစ်သည် ထို့ပြင် မိမိထို ပေးထားသော list ကို ကြည့်ပြီးတော့လည်း ခွဲခြား နိုင်ပါသည်။

### Matplotlib Circle ( Donut Chart)

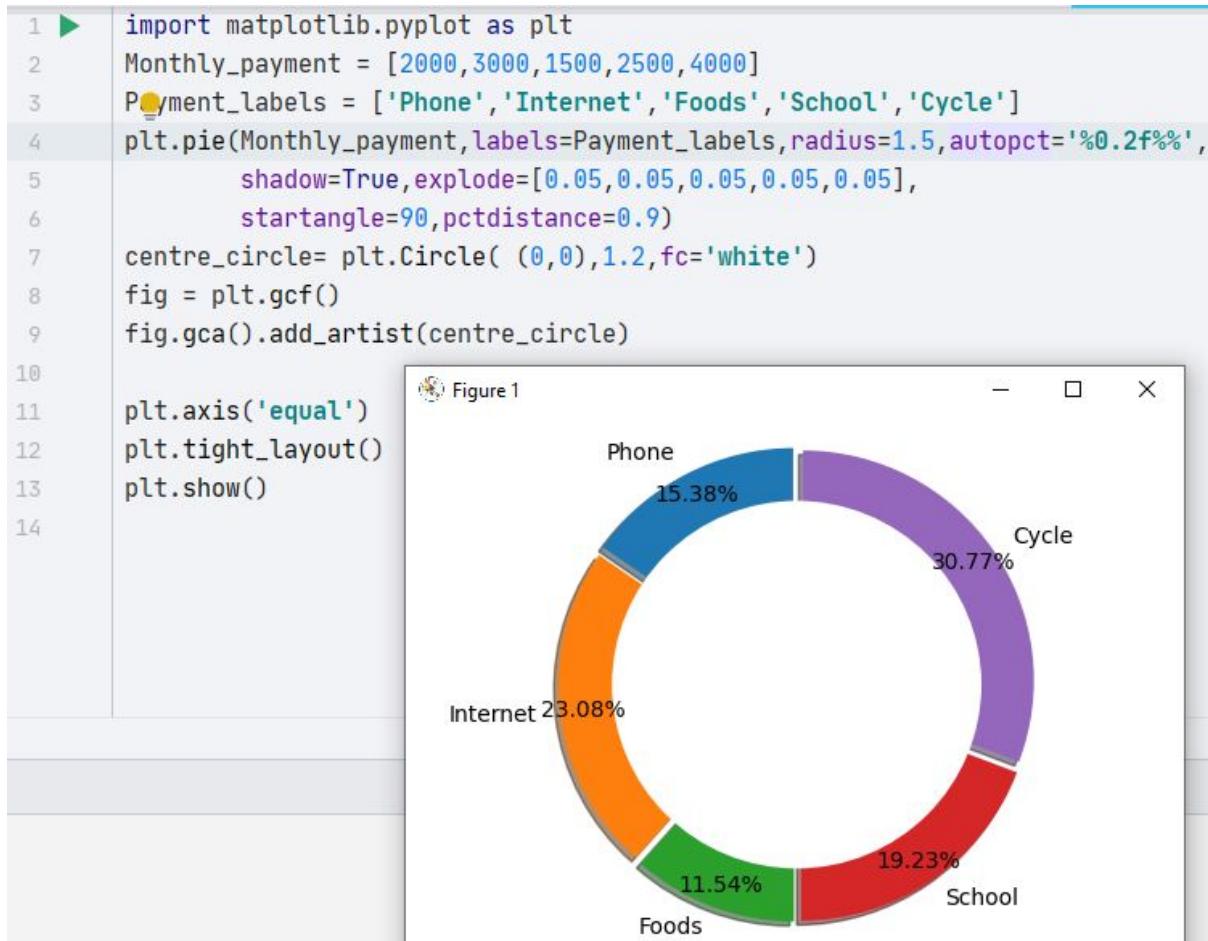
ယခင် သင်ခန်းစာမှာတော့ pie chart ကို ဖန်တီးခဲ့ပြီး ယခု သင်ခန်းစာမှာတော့ donut chart ကို ဖန်တီးကြည့်ပါမည်။ Donut Chart ဆိုတော့အလယ်မှာ စက်ပိုင်းပုံ အပေါက်နဲ့ ဖြစ်မှာ ထိုကြောင့် ထို

စက်ရိုင်းပုံ အပေါက် တစ်ခုကို ဖန်တီးပါမည်။ Matplotlib တွင် Circle တစ်ခုကို ဖန်တီးရာ၏ အပိုင်နှစ်ပိုင်းရှုပါသည် ပထမ တစ်ပိုင်းသည် circle size , color တို့ကို သတ်မှတ် ပေးသည့် အပိုင်းဖြစ်ပြီး ဒုတိယ အပိုင်းသည် ထို circle ကို ဆွဲ သည့် အပိုင်းဖြစ်ပါသည်။ Circle ရဲ့ အစိတ်အပိုင်းများကို ဖန်တီးရာတွင် Matplotlib မှ circle ဆိုသည့် method ကို အသုံးပြုပြီး ဆွဲရာတွင် add\_artist ဆိုသည့် method ကို သုံးပါသည်။

Circle method သည် circle အတွက် parameter သုံးခု ယူပါသည် ပထမ တစ်ခုသည် circle ရဲ့ centre အတွက် ဆိုလိုခြင်း ဖြစ်ပြီး ဒုတိယ တစ်ခုသည် စက်ရိုင်းရဲ့ size r(radius) ဖြစ်သည်။ နောက်ဆုံး တစ်ခုသည် ထို စက်ရိုင်းထဲတွင် ဖြည့်မည့် color ဖြစ်ပါသည်။ ထိုကဲ့သို့ သတ်မှတ်ပြီးသော circle ရဲ့ object တစ်ခုလုံးကို add\_artist ဆိုသည့် method ထဲသို့ paramter တစ်ခု အနေဖြင့် ထည့်ပေးလိုက်ရပါသည်။



အထက်ပါ program ကို ကြည့်မည် ဆိုလျှင် percentage တန်ဘိုးတွေဟာ နေရာ အတည်တကျမဖြစ်တော့ပဲ သက်ဆိုင်ရာ data များနဲ့ အနည်းငယ် ကွာဟာ နေသည်ကို မြင်ရပါမည်။ data များနှင့် တစ်ထပ်တည်း ကျနေစေရန် pctdistance ဆိုသည့် method ကို သုံးရပါမည် ထို function သည် autopct ဆိုသည့် attribute နှင့် အလုပ် အတူ လုပ်ပါသည်။ အကယ်၍ autopct ကို မသုံးဘူး ဆိုလျှင်တော့ pctdistance ကို သုံးစရာ မလိုပါဘူး။ pctdistance က autopct က ထုတ်ပေးလိုက်တဲ့ value တွေကို သူနဲ့ သက်ဆိုင်ရာ နေရာတွေမှာ ပေါ်ပေးနိုင်ဖို့ ကူညီ ပေးပါတယ် သူရဲ့ default value က 0.6 ဖြစ်ပါတယ်။



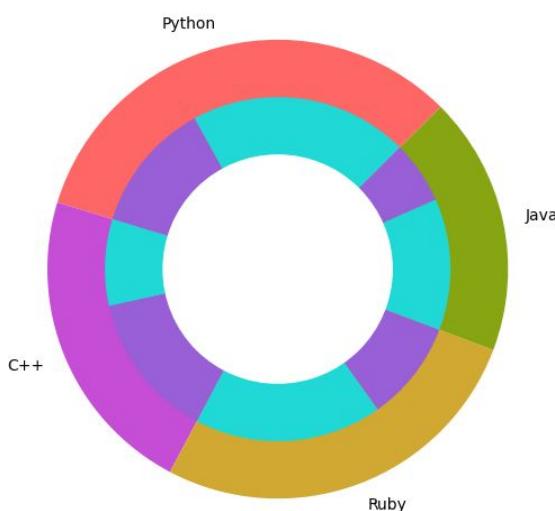
နောက်ထပ် program တစ်ပုဒ်အနေဖြင့် ရှေ့တွင် ရေးဆွဲခဲ့သော Donut Chart အလည်တွင် နောက်ထပ် cricle တစ်ခု ထပ်ဆွဲပါမည်။ ပထမဆုံး အနေဖြင့် problem တစ်ခုကို စတင် စဉ်းစား ကြည့်ကြပါနဲ့ programming languages 4 ခုရှိသည် အသုံးပြုသည့် လူပိုးရေမှာ 500,400,379,330 တွင် အမျိုးသမီး အမျိုးသား များလည်း ပါဝင်နေပါသည်။ ထိုအချက်လက် များပေါ်တွင် မူတည်ပြီး Pie chart နှစ်ခုကို ဆွဲကာ ပထမ တစ်ခုသည် data ချက်လက်များကို ဖော်ပြုမည် ဖြစ်ပြီး ဒုတိယ တစ်ခုသည် man or woman များကို ဆွဲခြေား ကာ ဖော်ပြု လေးမည့် pie chart ဖြစ်သည်။

```

1 import matplotlib.pyplot as plt
2
3 Programming = ['Python', 'C++', 'Ruby', 'Java']
4 Users = [504, 337, 415, 280]
5 Labels_gender = ['Man', 'Woman', 'Man', 'Woman', 'Man', 'Woman', 'Man', 'Woman']
6 man_woman = [315, 189, 125, 212, 270, 145, 190, 90]
7 pro_colors = ['#ff6666', '#C64DD5', '#D1A831', '#87A513']
8 m_w_colors = ['#20D9D6', '#985FD6', '#20D9D6', '#985FD6', '#20D9D6', '#985FD6',
9     '#20D9D6', '#985FD6']
10
11 plt.pie(Users, labels=Programming, colors=pro_colors, startangle=45)
12 plt.pie(man_woman, colors=m_w_colors, radius=0.75, startangle=45)
13
14 donut_chart = plt.Circle((0,0),0.5,color='black', fc='white', linewidth=0)
15
16 fig = plt.gcf()
17 fig.gca().add_artist(donut_chart)
18
19 plt.axis('equal')
20 plt.tight_layout()
21 plt.show()

```

အထက်ပါ program တွင် line 3 and 4 မှာ programming languages တွေနှင့် ထို programming languages တွေအား အသုံးပြုတဲ့ users တွေအတွက် list နှစ်ခု ဆောက်ထားပါတယ်။ line 4 and 5 တွင် အသုံးပြုသူ man and woman နှင့် man ဘယ်နှစ်ယောက် woman ဘယ်နှစ် ယောက်သုံးလဲ ဆိုတာ သိနိုင်စေရန် list နှစ်ခု ထပ်ဆောက် လိုက်ပါသည်။ line 7 and 8 တွင်တော့ programming languages တွေအတွက် color နှင့် man and woman အတွက် colors များကို ခွဲပေးထားပါသည်။ line 11 and 12 တွင် pie chart နှစ်ခု အတွက် data များကို ထည့်ပေး ပါသည် ပထမ Pie chart အတွက် radius ကို 0.5 အနေဖြင့်သာပေး ထားပြီး ဒုတိယ တွင်တော့ 0.75 ကို ပေးထားပါသည်။ ထို့ပြင် နှစ်ခုလုံးရဲ့ startangle ကိုလည်း 45 degree စီ ပေးထားပါသည်။

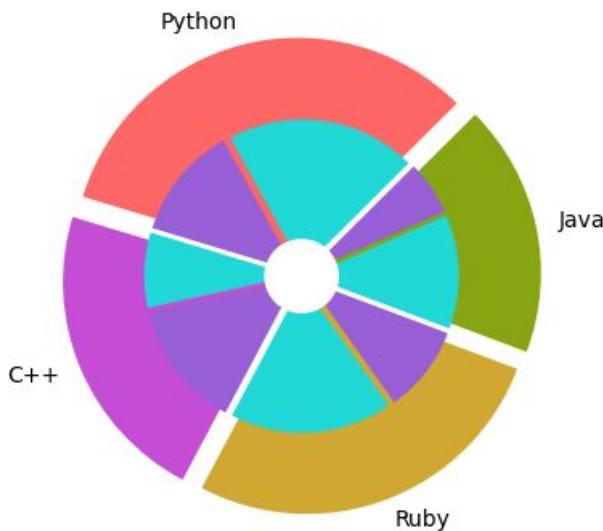


အကယ်၍ အထက်ပါ ပုံတွင် explode ကို ထပ်ထည့်လိုပါက ပိုမိုထင်ရှားစေရန် circle ငဲ့ radius နဲ့ အတူ ဆောက်ပါ အတိုင်း ပြောင်းနိုင်ပါသည်။

```

12
13     plt.pie(Users, labels=Programming, colors=pro_colors,
14             startangle=45,explode=explode, radius=3) ←
15     plt.pie(man_woman,colors=m_w_colors, radius=2, ←
16             startangle=45,explode=explode_gender)
17

```



### Matplotlib Real Time Data

ယခု သင်ခန်းစာမျက်တော့ real time data တွေကို matplotlib နဲ့ ဖော်ပြသွားမှာ ဖြစ်ပြီး real time data တွေကိုတော့ ယခု သင်ခန်းစာများ computer cpu ကနေ တစ်ဆင့် ရယူမှာ ဖြစ်ပါတယ်။ cpu နှင့် ပတ်သက်သည့် function call များကို ခေါ်သုံး နိုင်ရန် အတွက် psutil ဆိုသည့် library ကို install လုပ်ပေးရမှာ ဖြစ်ပါတယ်။ ထိုပြင် real time data များသည် အမြဲတမ်း တိုးပွားလာမှာ ဖြစ်တဲ့ အတွက် ထို တိုးပွားလာတဲ့ data တွေကို လိုက်ပြ ပေးနိုင်ရန် FuncAnimation ဆိုသည့် method ကို matplotlib.animation ထဲမှ import လုပ်ပေးရန်လည်း လိုအပ်ပါသည်။

```

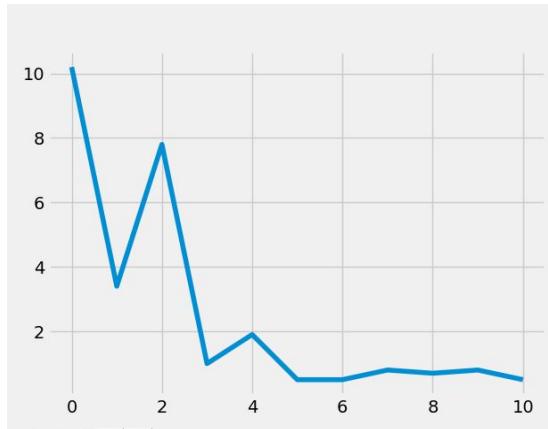
1 import psutil
2 import matplotlib.pyplot as plt
3 from itertools import count
4 from matplotlib.animation import FuncAnimation
5 plt.style.use('fivethirtyeight')
6
7 x_vals=[]
8 y_vals=[]
9
10 index=count()
11 def draw(i):
12     x_vals.append(next(index))
13     y_vals.append(psutil.cpu_percent())
14     plt.cla()
15     plt.plot(x_vals,y_vals)
16 animat = FuncAnimation(plt.gcf(),draw, interval=1000)
17
18 plt.tight_layout()
19 plt.show()

```

အထက်ပါ program တွင် line 10 နှင့် count method ကို ခေါ်ထားပါသည် count method သည် ပုံမှန် အားဖြင့် parameter နှစ်ခု ယူပါသည် ပထမ တစ်ခုသည် စမှတ်ဖြစ်ပြီး ဒုတိယ တစ်ခုသည် step ဘယ်လို ပုံစံ တိုးသွားမလဲ ဆိုတာ ဖြစ်ပါတယ်။ စာရေးသူ အနေဖြင့် count method ထဲတွင် မည်သည့် parameter မှ ထည့်ပေးထားခြင်း မရှိသည့် အတွက် default zero ဖြင့် အလုပ် လုပ်သွားမည် ဖြစ်သည့် စာရေးသူ အနေဖြင့် line 12 တွင် next ကို သုံးထားတာ ဖြစ်သည့် အတွက် သူ့ရဲ့ နောက် values တွေကို ဖော်ပြ နေမည် ဖြစ်သည်။ ပထမဆုံး အကြိမ်တွင် zero က စဉ်သော်လည်း next ဒုတိယ value သည် 1 ဖြစ်သည့် အတွက် 1 ကနေ စတင် သတ်မှတ်လိမ့်မည် ဖြစ်သည်။ ယခု အဆင့်အား ပိုမို နားလည်ရန် debug လုပ်သင့်ပေးသည်။ line 13 တွင် cpu\_percent() ကို သုံးထားပါသည် cpu\_percent() method သည် ယခုလက်ရှိ မိမိတို့အသုံးပြုနေသော cpu ရဲ့ အသုံးပြုနှင့် ကို percentage နဲ့ ပြန်လည် ဖော်ပြ ပေးပါသည်။ အကယ်၍ cpu\_percent() method ကို သုံးမည် ဆိုလျှင် parameter အနေဖြင့် time interval(seconds) ကိုထည့် ပြီး သုံးသင့်ပါသည် သို့သော် စာရေးသူသည် line 16 တွင် interval ကို 1000 milliseconds ခံထားပါသည် 1000 milliseconds သည် 1 second ဖြစ်ပါသည်။ line 14 တွင် ရေးထားသော cla() method သည် clear an axes ကို ဆိုလိုခြင်း ဖြစ်ပြီး ယခုလက်ရှိ figure တွင် ဖော်ပြနေသော axes တွေကို clear လုပ်ပေးရန် သုံးပါသည်။

Line 16 FuncAnimation method ကတေသာ့ matplotlib ထဲမှ animation ဆိုတဲ့ class မှ လာခြင်း ဖြစ်ပြီး ပြောင်းလဲ နေတဲ့ data တွေ wave form တွေ စတဲ့ အချက်လက် တွေကို animation ပုံစံဖြင့် ဖော်ပြ ရာတွင် အသုံးပြုပါတယ်။ plt.gcf() method သည် get current figure ကို ဆိုလိုခြင်း ဖြစ်ပြီး ယခုလက်ရှိ ဖော်ပြထားတဲ့ figure ကို ရယူလို တဲ့ အချိန်မှာ အသုံးပြုပါတယ် အကယ်၍ figure မရှိဘူး ဆိုလျှင်လည်း figure() function ကို သုံးပြီး figure တစ်ခုကို ဖန်တီးပါသည်။ ဒုတိယ တစ်ခုကတေသာ့ draw

ဆိတဲ့ function ကို လုမ်းခေါ်တာ ဖြစ်ပါတယ် draw function ကို ခေါ်တဲ့ အရှင်မှာ သူနောက်မှာ ဘာ parameter မှ ပါမသွားလော်လည်း draw function ကို ကြော်တဲ့ အခါမှာတော့ parameter တစ်ခုကို ထည့်ပေးထားရပါတယ် ဘာကြောင့်လည်း ဆိတော့ draw function ကို ခေါ်တဲ့ အရှင်မှာ FuncAnimation method ကနေ တစ်ဆင့် frame , \*fargs ဆိတဲ့ parameter နှစ်ခု ထည့်ပေးလိုက်တဲ့အ တွက်ကြောင့် ဖြစ်ပါတယ် draw ထဲမှ i ကို တစ်နည်းအားဖြင့် frame number လိုလည်း ခေါ်ပါတယ်။ နောက်ဆုံး တစ်ခု interval ကတော့ frame တစ်ခု နဲ့ တစ်ခုကြား ခေါ်တဲ့ ကြာချိန် ဖြစ်ပါတယ်။ ထို interval ရဲ့ default value သည် 200 milliseconds ဖြစ်ပါတယ်။



### Wave Form(FuncAnimation)

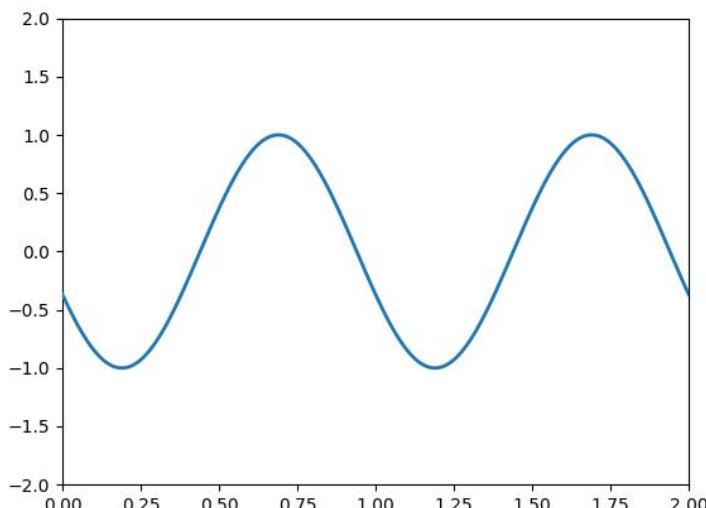
နောက်ထပ် program တစ်ပုဒ်အနေဖြင့် FuncAnimation ကို ပိုမို ကွွမ်းကျင်သွားစေရန် wave form program ကို ဆက်လေ့လာကြပါစို့။

```

1 import numpy as np
2 from matplotlib import pyplot as plt
3 from matplotlib import animation
4
5 fig = plt.figure()
6 ax = plt.axes(xlim=(0, 2), ylim=(-2, 2))
7 line, = ax.plot([], [], lw=2)
8 def init():
9     line.set_data([], [])
10    return line,
11 def animate(i):
12     x = np.linspace(0, 2, 1000)
13     y = np.sin(2 * np.pi * (x - 0.01 * i))
14     line.set_data(x, y)
15    return line,
16 anim = animation.FuncAnimation(fig, animate, init_func=init,
17                               frames=200, interval=20, blit=True)
18
19 plt.show()
```

အထက်ပါ program တွင် line 5 နဲ့ fig object တစ်ခုကို Figure() method မှ တည်ဆောက်ထားပါသည် Figure များ ဆွဲရန် အတွက် ဖြစ်ပါသည်။ Line 6 တွင် axes method ကို သုံးပြီး xlim and ylim value များကို သတ်မှတ်ပေးထားပါသည်။ ကတော့ linewidth ကို ဆိုလိုချင်တာပါ။ line

7 , line 10 , line 15 များနောက်မှ line နောက်တွင် , ပါဝင်သည်ကို သတိပြုပါ။ , မပါဝင်ဘူးဆိုလျှင် Line2D object is not iterable ဆိုတဲ့ TypeError တက်ပါလိမ့်မယ် line 12 and 13 သည် numpy ထဲမှ linspace နှင့် sin method ကို ခေါ်သုံးထားတာ ဖြစ်တဲ့ အတွက် မရင်းတော့ပါ ရှုနိုင်းသင်ခန်းစာများတွင် ရှင်းခဲ့ပြီး ဖြစ်ပါသည်။ FuncAnimation method ထဲတွင် သုံးခုမျှကိုပြု၍ init\_func သည် နောက်ထပ် ဘာမှ မရှိသေးတဲ့ frame တစ်ခု ထပ်ဆွဲဖို့ အတွက် function call ခေါ်လေးတာ ဖြစ်ပါတယ် အခေါ်ခံရသော function နှင့် နောက်ဆုံး parameter ဖြစ်သည့် blit သည် ဆက်စပ် နေပါတယ်။ အကယ်၍ blit ရဲ့ value ကို True ဟု ပေးခဲ့မည် ဆိုလျှင် init\_func မှ ခေါ်လိုက်သော init function သည် return value ကို ပေးကို ပေးရမည် ဖြစ်ပြီး ထို value များကို blitting algorithm က အသုံးပြုပြီး figure ရဲ့ ဘယ်အပိုင်းတွေဟာ ပြန်ဆွဲဖို့ လိုသလိုဆိုတာကို ဆုံးဖြတ်ပါတယ်။ အကယ်၍ blit value သည် false ဖြစ်နေခဲ့မည် ဆိုလျှင် return value တွေကို ပြန်မသုံးတော့ပါဘူး။ သတိပြုရန် အချက်မှာ line 15 မှ line, သည် modified ဖြစ်တဲ့ data အသစ်တွေ ဖြစ်တဲ့ tuple data တွေကို return ပြန် ပေးပေါ်ပြုပြီး ထို data တွေသည် Animation framework ထဲ၌ ဘယ်အပိုင်းကို animate လုပ်ရမယ် ဆိုတာ ပြောပြ ပေးပါတယ်။



### 3D Plot

ယခု lesson မှာတော့ chart တွေကို 3d plot တွေဖြင့် ဖော်ပြ သွားမှာ ဖြစ်ပါတယ် 3d plot ဖြင့် ဖော်ပြ ပေးနိုင်ဖို့ အတွက် axes3d ကို Import လုပ်ပေးရန် လိုအပ်ပြီး ထို object သည် mpl\_toolkits.mplot3d class ထဲမှ ဖြစ်ပါသည် ထို့ကြောင့် matplotlib library ကို install လုပ်ထားရုံဖြင့် အသုံးပြု နိုင်ပါသည်။

X and y အတွက် value တွေကိုတော့ line 4 and 5 တွင်ဖော်ပြထားပြီး numpy array ကို သုံးပြီး float data တွေကို ထုတ်ထားပါသည်။ Line 7 မှာတော့ meshgrid method ကိုသုံးကာ One dimensional array နှစ်ခုကောင့် multi dimensional array ရအောင် ဖန်တီး လိုက်တာပါ တိမ့် တစ် x and y positions များကို ရှုပါတယ်။ numpy သင်ခန်းစာရဲ့ meshgrid array မှာ ဖော်ပြ ခဲ့ပြီး ဖြစ်ပါတယ်။ line

8 တွင် x , y ရဲ့ မှ value များကို အမြင့်များကို ဖော်ပြချင်တဲ့ အတွက် z တန်ဘိုး ကိုပါ ရှာထားပါတယ်။  
ထို့ကြောင့် data တစ်ခုမှာ x , y နှင့် အမြင့် ပမာဏ z တန်ဘိုးပါ ရှိနေမှာ ဖြစ်ပါတယ်။

```

1 import matplotlib.pyplot as plt , numpy as np
2 from mpl_toolkits.mplot3d import axes3d
3
4 x = np.array(range(0,3),float)
5 y = x.copy()
6
7 xpos ,ypos = np.meshgrid(x,y)
8 z = np.random.rand(3,3)
9
10 print("xpos\n",xpos)
11 print("ypos\n",ypos)
12 print("z",z)

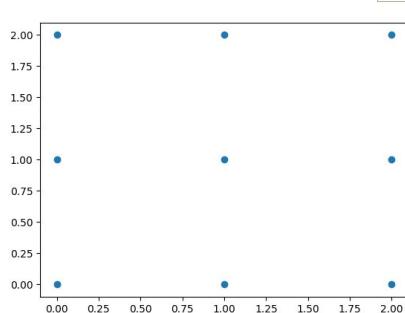
```

```

xpos
[[0. 1. 2.]
 [0. 1. 2.]
 [0. 1. 2.]]
ypos
[[0. 0. 0.]
 [1. 1. 1.]
 [2. 2. 2.]]
z [[0.35921468 0.59425909 0.72024001]
 [0.43059528 0.50480525 0.65090816]
 [0.1035428 0.43071996 0.65811003]]

```

အထက်ပါ program ရဲ့ Output များကို ကြည့်မည် ဆိုလျှင် xpos data 9 ခု ypos data 9 ခုကို ရှိနေပါသည်။ ထို xpos and ypos ရဲ့ x,y ရဲ့ အမှတ်များကိုထုတ်ကြည့်မည် ဆိုလျှင် x,y စုစုပေါင်း အမှတ် ၉ ခု ရှိနေပါပြီ။ သို့သော် ယခု အတိုင်း plot လုပ်ကြည့်မည် ဆိုလျှင် scatter plot နှင့်အောက်ပါ ပုံအတိုင်းထုတ်ကြည့်နိုင်သည်။ plt.scatter(xpos,ypos)



ဒုတိယ တစ်ခင့် အနေဖြင့် xpos and ypos များကို row ပုံစံ ပြန်ပြောင်းစွဲ လိုပါသေးသည် အဘယ်ကြောင့်ဆိုသော် meshgrid မှ data များသည် two dimention ဖြစ်နေပါသည် ထို data အားလုံးအား row ပုံစံဖြင့် One dimensional အတိုင်းပြန်ယူလိုပါက flatten() method ကို အသုံးပြုရပါမည်။ သတိပြုရန် အချက်မှာ 3D ဖြစ်စွဲ ဆိုလျှင် x,y,z values များ လိုအပ်ပါသည်။ z values ရ ရန်

အတွက် `np.zero_like(xpos)` ကို သုံးလိုက်ပါသည် `zero_like` သည် `zero` တွေချည်းပဲ ဖော်တီး ပေးမှာ ဖြစ်ပြီး `xpos` သည် `zero` အရေး အတွက် ဖြစ်ပါသည်။

```

10     xpos = xpos.flatten()
11     ypos = ypos.flatten()
12     zpos = np.zeros_like(xpos)

```

က တိယ အဆင့် အနေဖြင့် x,y,z တို့ရဲ့ dx, dy and dz dimensions များကို ဖော်တီး ပေးရန် လိုအပ်ပါသည်။ dx and dy values များကို `np.ones(xpos)` ဆိုပြီး နှစ်ခုလုံး အတွက် ဖော်တီးပေး နိုင်ပါသည်။  
`dx= np.ones(xpos)`

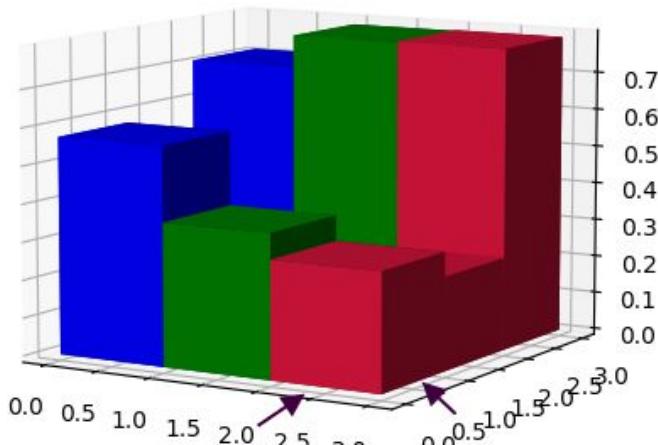
`dy=np.ones(xpos) or dy.copy()` ဟူရေး သားနိုင်ပါသည်။ dz ကတော့ မိမိတို့ ထည့်ပေးချင်သည့် dimensions များကို ထည့်ပေးနိုင်ပါသည်။ ဥပမာ အနေဖြင့် အောက်ပါ အတိုင်း ရေးသား နိုင်သည်။

```

14     dx = np.ones(10)
15     dy = np.ones(10)
16     dz = [1,2,3,4,5,6,7,8,9,10]

```

အထက်ပါ အတိုင်း `np.ones` ကို သုံးမည် ဆိုလျှင် Numpy.float64 ဆိုသည့် error တက်နိုင်ပါသည်အကယ်၍ error တက်ခဲ့လျှင် `np.ones_like(zpos)` ကို သုံးနိုင်ပါသည်။ ထိုကဲ့သို့ အသုံးပြုမည် ဆိုလျှင် 1 များထုတ်ပေးပါမည်။ 1 များ ထုတ်ပေးခဲ့လျှင် 1 အပြည့် ဖြစ်နေသည့် အတွက် အောက်ပါ ပုံအတိုင်း တစ်ခုနှင့် တစ်ခု ပူးက် နေနိုင်ပါသည်။ ထို့ကြောင့် စာရေးသူ အနေဖြင့် program ထဲတွင် `0.5*np.ones_like(zpos)` ဟု ရေးထားပါသည်။

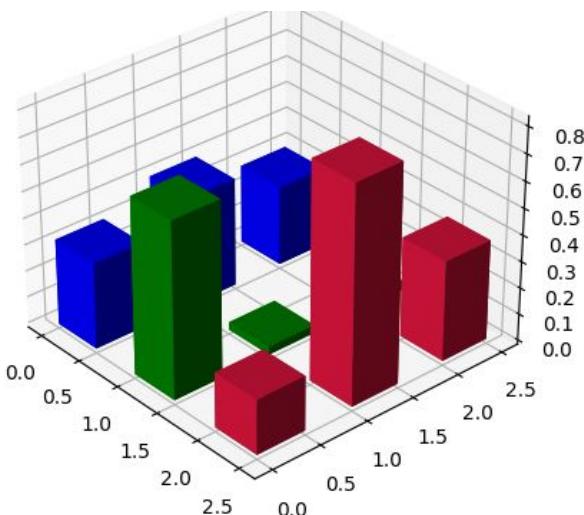


ယခုဆိုလျှင် 3d bar ဆိုဒ္ဓါး အတွက် data အချက်လက်များ အားလုံး လုံလောက်ပြီ ဖြစ်ပါသည်။ program အပြည့်အစုံကို စာရေးသူ အနေဖြင့်အောက်မှာ ဖော်ပြထားပါသည်။ စာရေးသူ အနေဖြင့် dz အတွက် values များကို `np.random.rand` မှ တစ်ဆင့် ရယူ ထားပါသည်။ ထို့သို့ ရရှိလာသော values များကို `flatten` သုံးကာ စာရေးသူတို့ လိုချင်သည့် ပုံစံ အတိုင်း ပြန်ပြောင်း ထားပါသည်။

```

1 import matplotlib.pyplot as plt , numpy as np
2 from mpl_toolkits.mplot3d import axes3d
3
4 x = np.array(range(0,3),float)
5 y = x.copy()
6
7 #meshgrid ကိုထိုးပြီး data များ ဖန်တီးထားပါသည်။
8 #မိမိတို့ အစားထိုးလုပ်သည့် data များကို x နေရာတွင် သုံးခိုင်သည်
9 xpos ,ypos = np.meshgrid(x,y)
10 z = np.random.rand(3,3) # z dimension အတွက်ဖြစ်သည်
11
12 xpos = xpos.flatten()
13 ypos = ypos.flatten()
14 zpos = np.zeros_like(xpos)
15 #x,y,z အတွက် တည်နေရာများ ဖန်တီး ပေးခြင်း ဖြစ်ပါသည်
16 #3d ဖြစ်သည်အတွက် x နှင့် y အတွက် တည်နေရာများ ဖော်ပြန်သွားခဲ့သည်။
17
18 # 1 တွေချည်းပဲ ထွက်လာသည့် values
19 dx = np.ones_like(zpos)
20 dy = dx.copy()
21 dz = z.flatten()
22
23 colors = ["b","g","crimson"]*3
24 fig = plt.figure(figsize=(7,5))
25 ax = fig.add_subplot(111 , projection ="3d")
26 ax.bar3d(xpos , ypos ,zpos,dx ,dy,dz,color=colors)
27 plt.show()

```



အထက်တွင် ဖော်ပြထားသော program အား x\_label , y\_label and z\_label များနှင့် အတူ x တွင် ပါဝင်သော အရေ အတွက် y တွင် ပါဝင်သော အရေ အတွက်တို့ကို ထည့်နိုင် ပါသေးသည်။ အထက်ပါ ပုံကို စာရေးသုတေသန MPT, Ooredoo , Telenor စသည့် data များ ထည့်သွင်း ကြည့်ပါမည်။ x နေရာတွင် အရေအတွက်ကို ထားပြီး y နေရာတွင် services များကို ထည့်ကာ ဥပမာ voice , message , internet တို့ကို ထားပြီး z နေရာတွင် ထို services များ အသုံးပြုခြင်း အတွက် ကုန်ကျ စရိတ်များကို ထားပါမည်။

```
1 ► import matplotlib.pyplot as plt , numpy as np
2   from mpl_toolkits.mplot3d import axes3d
3
4   x = np.array(range(0,3),float)
5   y = x.copy()
6
7   #meshgrid ကိုသုံးပြီး data များ ဖန်တီးထားပါသည်။
8   #မိမိတဲ့ အစားထိုးလိုသည့် data များကို x and y နေရာတွင် သုံးနိုင်သည်
9   xpos ,ypos = np.meshgrid(x,y)
10  z = np.random.rand(3,3) # z dimension အတွက်ဖြစ်သည်
11
12  xpos = xpos.flatten()
13  ypos = ypos.flatten()
14  zpos = np.zeros_like(xpos)
15  #x,y,z အတွက် တည်နေရာများ ဖန်တီး ပေးခြင်း ဖြစ်ပါသည်
16  #3d ဖြစ်သည်အတွက် x y z values သုံးခဲ့ ဖန်တီး ပေးရသည်။
17
18  # 1 တွေချည်းပဲ ထွက်လာသည့် values
19  dx = 0.5*np.ones_like(zpos)
20  dy = dx.copy()
21  dz = z.flatten()
22
23  colors = ["b","g","crimson"]*3
24  fig = plt.figure(figsize=(7,5))
25  ax = fig.add_subplot(111 , projection ="3d")
26
27  # figure ခဲ့ title ကိုထည့်မည်
28  ax.set_title("Myanmar Operators", fontsize=20)
29
30  # figure ခဲ့ x အတွက် lable ကို သတ်မှတ်ပေးမည်
31  ax.set_xticklabels([3,2,1]) #နံပါတ်စဉ် တပ်ပေးခြင်း
32  ax.set_xticks(range(3)) # အခေါ်အတွက်ကို ဖော်ပြ ပေးခြင်း
33  ax.set_xlabel("GoodServices",labelpad=7)
```

```

34
35     # y သည်လည်း x အတိုင်းပင် ဖြစ်သည် ထို့ကော် ဗျားလုံး
36     # ဖော်ပြုသည်။
37     ax.set_yticklabels(["Voice", "SMS", "Internet"])
38     ax.set_yticks(range(3))
39     ax.set_ylabel("Operators", labelpad=7)
40
41     # z အတော့ဌာ pricing ကို ဖော်ပြု၍ အတွက်ထိုးထားပါသည်။
42     ax.set_zticklabels([0, 100, 200, 300, 400])
43     ax.set_zticks(np.arange(0.0, 1.0, 0.2))
44     ax.set_zlabel("Pricing Kyats")
45
46     x_leg=plt.Rectangle((0,0),1,1,fc="b")
47     y_leg=plt.Rectangle((0,0),1,1,fc="g")
48     z_leg=plt.Rectangle((0,0),1,1,fc="crimson")
49
50     labels =[ "MPT", "Ooredoo", "Telenor"]
51     ax.legend([x_leg,y_leg,z_leg],labels,fontsize=12,loc=(1.1,0.4))
52     ax.bar3d(xpos , ypos , zpos, dx ,dy,dz,color=colors)
53     plt.show()

```

## Myanmar Operators

