**Project Two**

**Programming with Matplotlib**

***Objectives of this lab:***

* Use pip to install python packages.
* Use the matplotlib module and submodules for graphing.
* Create bar graphs, line graphs, and pie charts.

Project Name: Project 2

Group Members: **ADIT GAUTAM SOLO**

Project Description: Python with Matplotlib library

Date: 1.28.2025

**Section 3.23 in Zybooks** introduces the topic of plotting simple graphs. In this section you will find the necessary tools that will help you with this project. **You will need to read this section**.

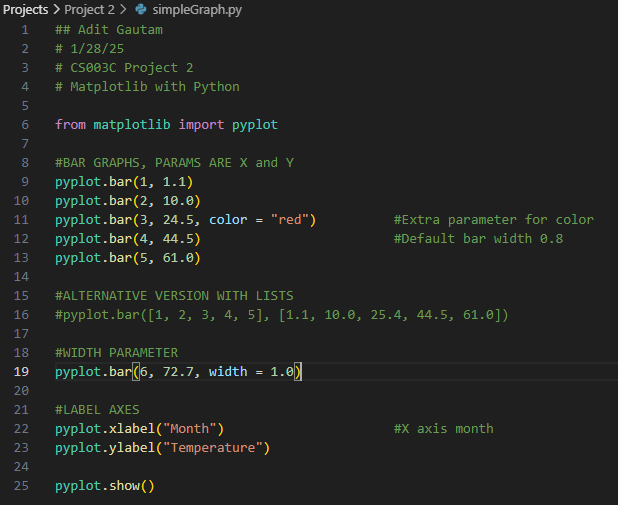
**Part One: Bar Charts**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

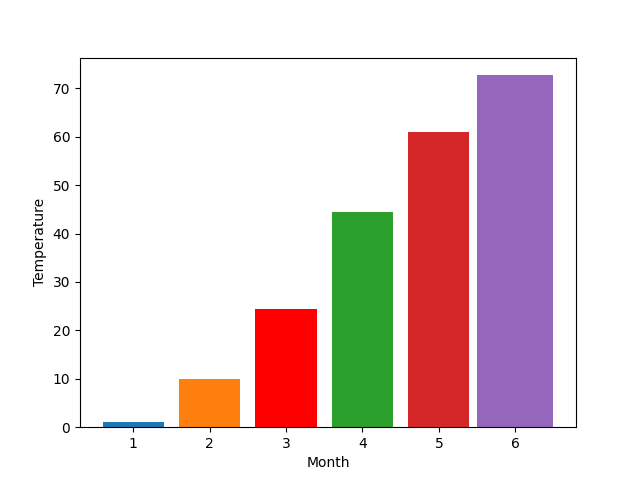
**Exercise 1.1**: Create a simple bar graph.

CONTINUED BELOW

**Include a Copy/Paste your code here (with header and documentation):**

****

**Include a full screenshot of your execution here:**



| ## Adit Gautam # 1/28/25 # CS003C Project 2 # Matplotlib with Python # Simple Bar Graph  from matplotlib import pyplot  #BAR GRAPHS, PARAMS ARE X and Y pyplot.bar(1, 1.1) pyplot.bar(2, 10.0) pyplot.bar(3, 24.5, color = "red") #Extra parameter for color pyplot.bar(4, 44.5) #Default bar width 0.8 pyplot.bar(5, 61.0)  #ALTERNATIVE VERSION WITH LISTS #pyplot.bar([1, 2, 3, 4, 5], [1.1, 10.0, 25.4, 44.5, 61.0])  #WIDTH PARAMETER pyplot.bar(6, 72.7, width = 1.0)  #LABEL AXES pyplot.xlabel("Month") #X axis month pyplot.ylabel("Temperature")  pyplot.show() |
| --- |

**Exercise 1.2**

Suppose you have been offered a job in beautiful Fairbanks, Alaska, and you are considering whether to accept. Perhaps you are concerned about the climate. This is how warm it gets on average each month:

A grid of calendar pages

Description automatically generated

Open a file called **Fairbanks.py**:

Create a variable for each month and assign a value for the temperature according to the chart above:

*january = 1.1*

Plot the temperature for each month.

*pyplot.bar(1, january)*

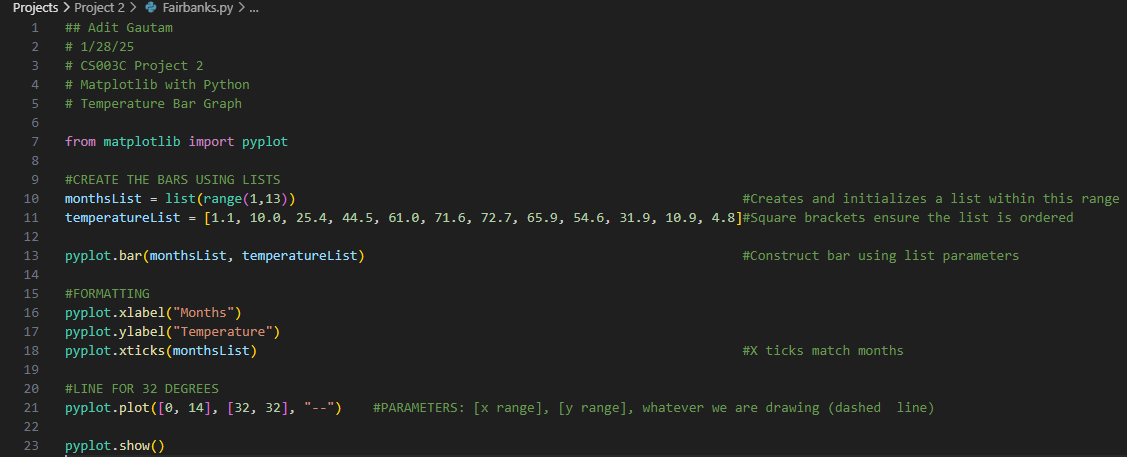
Draw a line across the graph for 32 degrees Fahrenheit:

*pyplot.plot([0, 14], [32, 32], "--")*

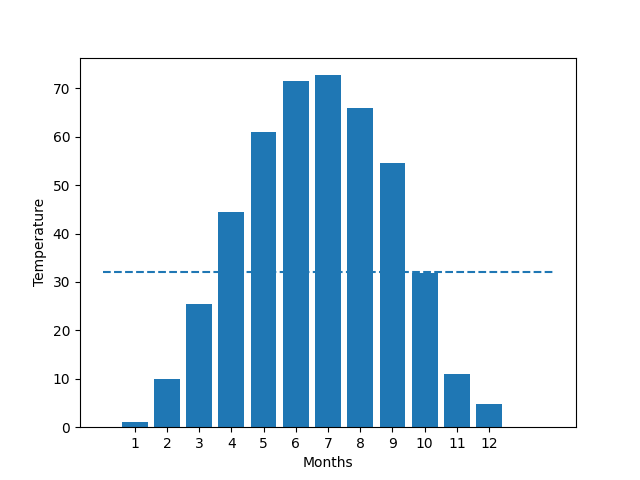
Format the graph (**refer to table 3.23.2: Plotting Functions in Zybooks**) and display it.

* Add a title.
* Add x-label and y-label.
* Add x-ticks.

**Include a Copy/Paste your code here (with header and documentation):**

****

**Include a full screenshot of your execution here:**



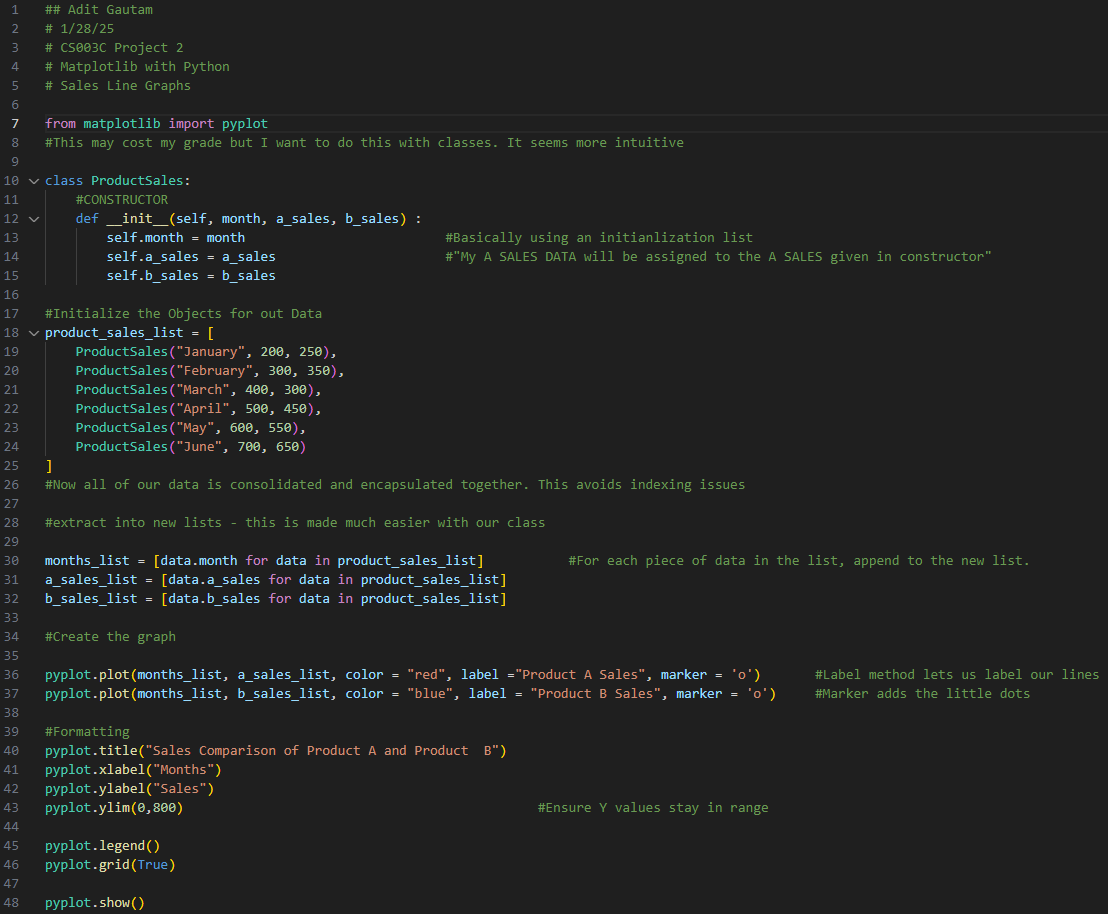
| **## Adit Gautam # 1/28/25 # CS003C Project 2 # Matplotlib with Python # Temperature Bar Graph  from matplotlib import pyplot  #CREATE THE BARS USING LISTS monthsList = list(range(1,13)) #Creates and initializes a list within this range temperatureList = [1.1, 10.0, 25.4, 44.5, 61.0, 71.6, 72.7, 65.9, 54.6, 31.9, 10.9, 4.8]#Square brackets ensure the list is ordered  pyplot.bar(monthsList, temperatureList) #Construct bar using list parameters  #FORMATTING pyplot.xlabel("Months of the Year") pyplot.ylabel("Temperature (in degrees F)") pyplot.xticks(monthsList) #X ticks match months  #LINE FOR 32 DEGREES pyplot.plot([0, 14], [32, 32], "--") #PARAMETERS: [x range], [y range], whatever we are drawing (dashed line)  pyplot.show()** |
| --- |

**Part Two: Line Graphs**

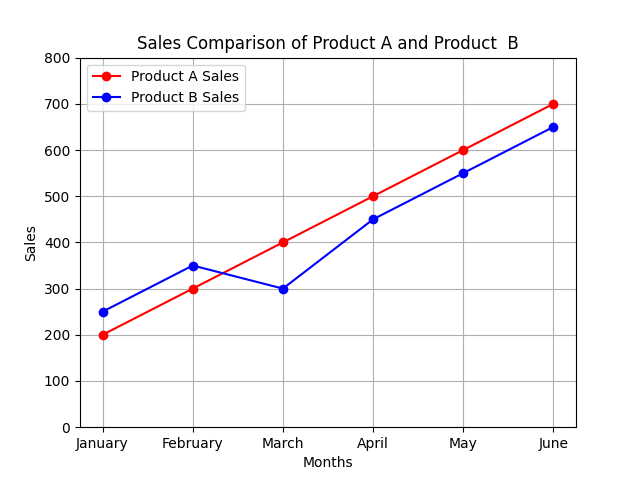
**Exercise 2.1**

**Include a Copy/Paste your code here (with header and documentation):**

**I did this with classes. Not sure if that’s allowed.**

****

**Include a full screenshot of your execution here:**

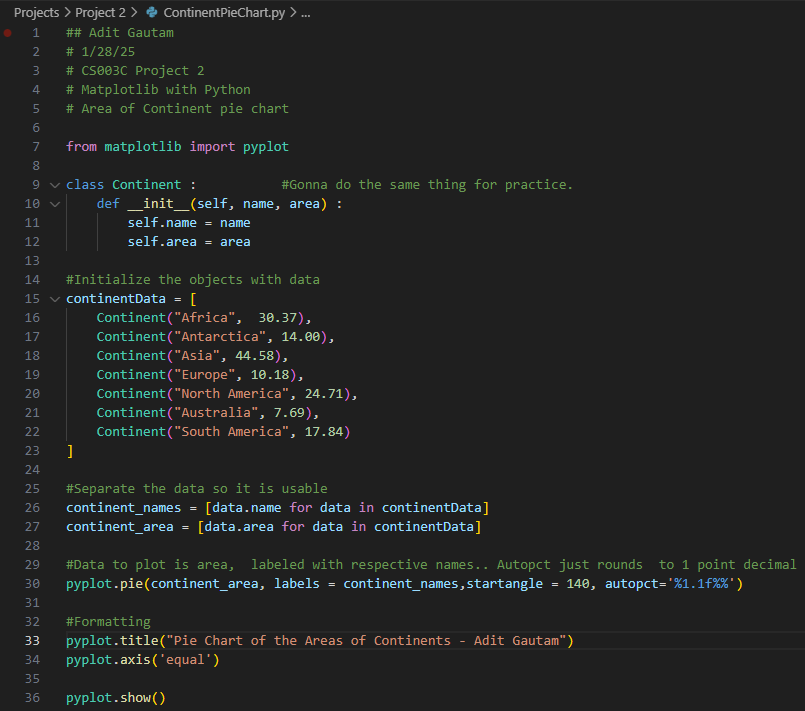
****

| **## Adit Gautam # 1/28/25 # CS003C Project 2 # Matplotlib with Python # Sales Line Graphs  from matplotlib import pyplot #This may cost my grade but I want to do this with classes. It seems more intuitive  class ProductSales:  #CONSTRUCTOR  def \_\_init\_\_(self, month, a\_sales, b\_sales) :  self.month = month #Basically using an initianlization list  self.a\_sales = a\_sales #"My A SALES DATA will be assigned to the A SALES given in constructor"  self.b\_sales = b\_sales  #Initialize the Objects for out Data product\_sales\_list = [  ProductSales("January", 200, 250),  ProductSales("February", 300, 350),  ProductSales("March", 400, 300),  ProductSales("April", 500, 450),  ProductSales("May", 600, 550),  ProductSales("June", 700, 650) ] #Now all of our data is consolidated and encapsulated together. This avoids indexing issues  #extract into new lists - this is made much easier with our class  months\_list = [data.month for data in product\_sales\_list] #For each piece of data in the list, append to the new list. a\_sales\_list = [data.a\_sales for data in product\_sales\_list] b\_sales\_list = [data.b\_sales for data in product\_sales\_list]  #Create the graph  pyplot.plot(months\_list, a\_sales\_list, color = "red", label ="Product A Sales", marker = 'o') #Label method lets us label our lines pyplot.plot(months\_list, b\_sales\_list, color = "blue", label = "Product B Sales", marker = 'o') #Marker adds the little dots  #Formatting pyplot.title("Sales Comparison of Product A and Product B") pyplot.xlabel("Months") pyplot.ylabel("Sales") pyplot.ylim(0,800) #Ensure Y values stay in range  pyplot.legend() pyplot.grid(True)  pyplot.show()** |
| --- |

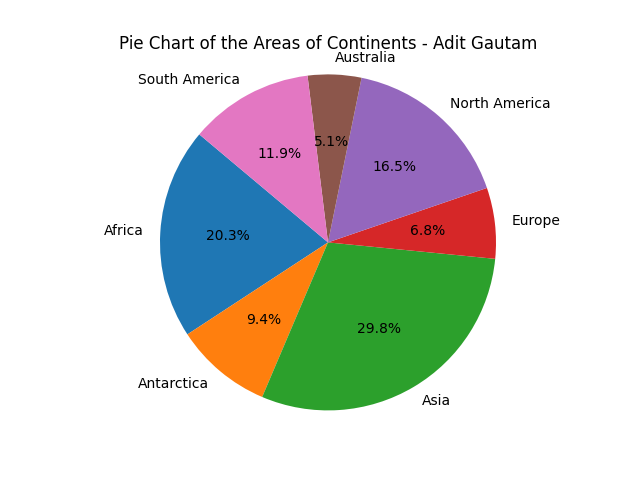
**Part Three: Pie Charts**

**Exercise 3.1**

**Include a Copy/Paste your code here (with header and documentation):**

****

**Include a full screenshot of your execution here:**

****

| **## Adit Gautam # 1/28/25 # CS003C Project 2 # Matplotlib with Python # Area of Continent pie chart  from matplotlib import pyplot  class Continent : #Gonna do the same thing for practice.  def \_\_init\_\_(self, name, area) :  self.name = name  self.area = area  #Initialize the objects with data continentData = [  Continent("Africa", 30.37),  Continent("Antarctica", 14.00),  Continent("Asia", 44.58),  Continent("Europe", 10.18),  Continent("North America", 24.71),  Continent("Australia", 7.69),  Continent("South America", 17.84) ]  #Separate the data so it is usable continent\_names = [data.name for data in continentData] continent\_area = [data.area for data in continentData]  #Data to plot is area, labeled with respective names.. Autopct just rounds to 1 point decimal pyplot.pie(continent\_area, labels = continent\_names,startangle = 140, autopct='%1.1f%%')  #Formatting pyplot.title("Pie Chart of the Areas of Continents - Adit Gautam") pyplot.axis('equal')  pyplot.show()** |
| --- |