In [20]:
```
import numpy as np
import pandas as pd
import scipy as sp
```

In [21]:
```
%matplotlib inline
import matplotlib.pyplot as plt
plt.style.use('ggplot')
```

In [22]:
```
%%file hw_data.csv
id,sex,weight,height
1,M,190,77
2,F,120,70
3,F,110,68
4,M,150,72
5,O,120,66
6,M,120,60
7,F,140,70
```

```
Overwriting hw_data.csv
```

# Python

## 1. Finish creating the following function that takes a list and returns the average value.

Add each element in the list to `total` and return `total`

### DO NOT use a library function nor `sum()`

In [23]:
```
def average(my_list):
    total = 0
    for item in my_list:
        #do something with item!
          ##  print(item)
        total = total + item

    return total/len(my_list)

average([1,2,1,4,3,2,5,9])
```

Out[23]:
```
3.375
```

## 2. Using a Dictionary keep track of the count of numbers (or items) from a list

In [24]:
```
def counts(my_list):
    counts = dict()
    for item in my_list:
        if item in counts:
            counts[item] = counts[item] + 1
        else:
            counts[item] = 1
    return counts

counts([1,2,1,4,3,2,5,9])
```

Out[24]:
```
{1: 2, 2: 2, 4: 1, 3: 1, 5: 1, 9: 1}
```

## 3. Using the `counts()` function you created above and the `.split()` function, return a dictionary of most occuring words from the following paragraph. Bonus, remove punctuation from words.

In [25]:
```
paragraph_text = '''
For a minute or two she stood looking at the house, and wondering what to do next, whe
The Fish-Footman began by producing from under his arm a great letter, nearly as large
Then they both bowed low, and their curls got entangled together.
Alice laughed so much at this, that she had to run back into the wood for fear of thei
Alice went timidly up to the door, and knocked.
'There's no sort of use in knocking,' said the Footman, 'and that for two reasons. Fir
'Please, then,' said Alice, 'how am I to get in?'
'There might be some sense in your knocking,' the Footman went on without attending to
'I shall sit here,' the Footman remarked, 'till tomorrow—'
At this moment the door of the house opened, and a large plate came skimming out, stra

import string # Need String Punctuation

pgph_clean = paragraph_text.translate(str.maketrans("","",string.punctuation))
pgph_clean_1 = pgph_clean.replace("'", "")
pgph_clean_2 = pgph_clean_1.replace("'", "")
pgph_clean_3 = pgph_clean_2.replace("—", "")
word_list = pgph_clean_3.split()
word_list_2 = counts(word_list)

print(word_list_2)
```

```
{'For': 3, 'a': 15, 'minute': 1, 'or': 2, 'two': 2, 'she': 6, 'stood': 1, 'looking':
2, 'at': 6, 'the': 32, 'house': 2, 'and': 17, 'wondering': 1, 'what': 2, 'to': 15, 'd
o': 1, 'next': 2, 'when': 2, 'suddenly': 1, 'footman': 3, 'in': 9, 'livery': 3, 'cam
e': 2, 'running': 1, 'out': 5, 'of': 9, 'woodshe': 1, 'considered': 1, 'him': 3, 'b
e': 2, 'because': 3, 'he': 5, 'was': 8, 'otherwise': 1, 'judging': 1, 'by': 3, 'his':
6, 'face': 2, 'only': 2, 'would': 1, 'have': 1, 'called': 1, 'fishand': 1, 'rapped':
1, 'loudly': 1, 'door': 6, 'with': 2, 'knuckles': 1, 'It': 1, 'opened': 2, 'another':
1, 'round': 1, 'large': 3, 'eyes': 2, 'like': 1, 'frog': 1, 'both': 2, 'footmen': 1,
'Alice': 5, 'noticed': 1, 'had': 4, 'powdered': 1, 'hair': 1, 'that': 3, 'curled': 1,
'all': 3, 'over': 2, 'their': 3, 'heads': 1, 'She': 1, 'felt': 1, 'very': 2, 'curiou
s': 1, 'know': 2, 'it': 3, 'about': 1, 'crept': 1, 'little': 2, 'way': 1, 'wood': 2,
'listen': 1, 'The': 2, 'FishFootman': 2, 'began': 1, 'producing': 1, 'from': 2, 'unde
r': 1, 'arm': 1, 'great': 2, 'letter': 1, 'nearly': 2, 'as': 4, 'himself': 1, 'this':
4, 'handed': 1, 'other': 2, 'saying': 1, 'solemn': 2, 'tone': 2, 'Duchess': 2, 'An':
2, 'invitation': 2, 'Queen': 2, 'play': 2, 'croquet': 2, 'FrogFootman': 1, 'repeate
d': 2, 'same': 2, 'changing': 1, 'order': 1, 'words': 1, 'From': 1, 'for': 3, 'Then':
1, 'they': 1, 'bowed': 1, 'low': 1, 'curls': 1, 'got': 1, 'entangled': 1, 'together':
1, 'laughed': 1, 'so': 2, 'much': 1, 'run': 1, 'back': 1, 'into': 3, 'fear': 1, 'hear
ing': 1, 'her': 2, 'peeped': 1, 'gone': 1, 'sitting': 1, 'on': 4, 'ground': 1, 'nea
r': 1, 'staring': 1, 'stupidly': 1, 'up': 3, 'sky': 2, 'went': 2, 'timidly': 1, 'knoc
ked': 1, 'Theres': 1, 'no': 2, 'sort': 1, 'use': 1, 'knocking': 2, 'said': 3, 'Footma
n': 3, 'reasons': 1, 'First': 1, 'Im': 1, 'side': 1, 'you': 6, 'are': 2, 'secondly':
1, 'theyre': 1, 'making': 1, 'such': 1, 'noise': 2, 'inside': 2, 'one': 2, 'could':
2, 'possibly': 1, 'hear': 1, 'And': 1, 'certainly': 1, 'there': 1, 'most': 1, 'extrao
rdinary': 1, 'going': 1, 'withina': 1, 'constant': 1, 'howling': 1, 'sneezing': 1, 'e
very': 1, 'now': 1, 'then': 2, 'crash': 1, 'if': 3, 'dish': 1, 'kettle': 1, 'been':
1, 'broken': 1, 'pieces': 2, 'Please': 1, 'how': 1, 'am': 2, 'I': 4, 'get': 2, 'Ther
e': 1, 'might': 3, 'some': 1, 'sense': 1, 'your': 1, 'without': 1, 'attending': 1, 'w
e': 1, 'between': 1, 'us': 1, 'instance': 1, 'were': 1, 'knock': 1, 'let': 1, 'He':
1, 'time': 1, 'speaking': 1, 'thought': 1, 'decidedly': 1, 'uncivil': 1, 'But': 2, 'p
erhaps': 1, 'cant': 1, 'help': 1, 'herself': 1, 'top': 1, 'head': 2, 'any': 1, 'rat
e': 1, 'answer': 1, 'questionsHow': 1, 'aloud': 1, 'shall': 1, 'sit': 1, 'here': 1,
'remarked': 1, 'till': 1, 'tomorrow': 1, 'At': 1, 'moment': 1, 'plate': 1, 'skimmin
g': 1, 'straight': 1, 'Footmans': 1, 'just': 1, 'grazed': 1, 'nose': 1, 'broke': 1,
'against': 1, 'trees': 1, 'behind': 1}
```

## 4. Read in a file using `open()` and iterated through the file line-by-line write each line from the file to a new file in a `title()`-ized. Create your own file for input

`This is the first line` -> `This Is The First Line`

Hint: There's a function to do this

In [26]:
```
jab = open("C:/Users/Brett/Desktop/JHU_AAP/ML/mlnn-main-personal/mlnn-main/02/jabberwc
# jab = open(r"C:\Users\Brett\Downloads\demofile.txt",'r')

jab_string = jab.read()
#print(jab_string)


new_jab = []
for line in jab_string.splitlines():
    line = str.title(line)
    new_jab.append(line)
    print(line)
```

```
’Twas Brillig, And The Slithy Toves
      Did Gyre And Gimble In The Wabe:
All Mimsy Were The Borogoves,
      And The Mome Raths Outgrabe.

“Beware The Jabberwock, My Son!
      The Jaws That Bite, The Claws That Catch!
Beware The Jubjub Bird, And Shun
      The Frumious Bandersnatch!”

He Took His Vorpal Sword In Hand;
      Long Time The Manxome Foe He Sought—
So Rested He By The Tumtum Tree
      And Stood Awhile In Thought.

And, As In Uffish Thought He Stood,
      The Jabberwock, With Eyes Of Flame,
Came Whiffling Through The Tulgey Wood,
      And Burbled As It Came!

One, Two! One, Two! And Through And Through
      The Vorpal Blade Went Snicker-Snack!
He Left It Dead, And With Its Head
      He Went Galumphing Back.

“And Hast Thou Slain The Jabberwock?
      Come To My Arms, My Beamish Boy!
O Frabjous Day! Callooh! Callay!”
      He Chortled In His Joy.

’Twas Brillig, And The Slithy Toves
      Did Gyre And Gimble In The Wabe:
All Mimsy Were The Borogoves,
      And The Mome Raths Outgrabe.
```

# Numpy

## 1. Given a list, find the average using a numpy function.

In [27]:
```
simple_list = [1,2,1,4,3,2,5,9]
np.mean(simple_list)
```

Out[27]:
```
3.375
```

## 2. Given two lists of Heights and Weights of individual, calculate the BMI of those individuals, without writing a `for-loop`

```
In [28]: heights = [174, 173, 173, 175, 171]
         weights = [88, 83, 92, 74, 77]

         np_heights = np.array(heights)
         np_weights = np.array(weights)

         bmis = np_weights / (np_heights/100)**2
         print(bmis)
```

```
[29.06592681 27.73229978 30.73941662 24.16326531 26.33288875]
```

## 3. Create an array of length 20 filled with random values (between 0 to 1)

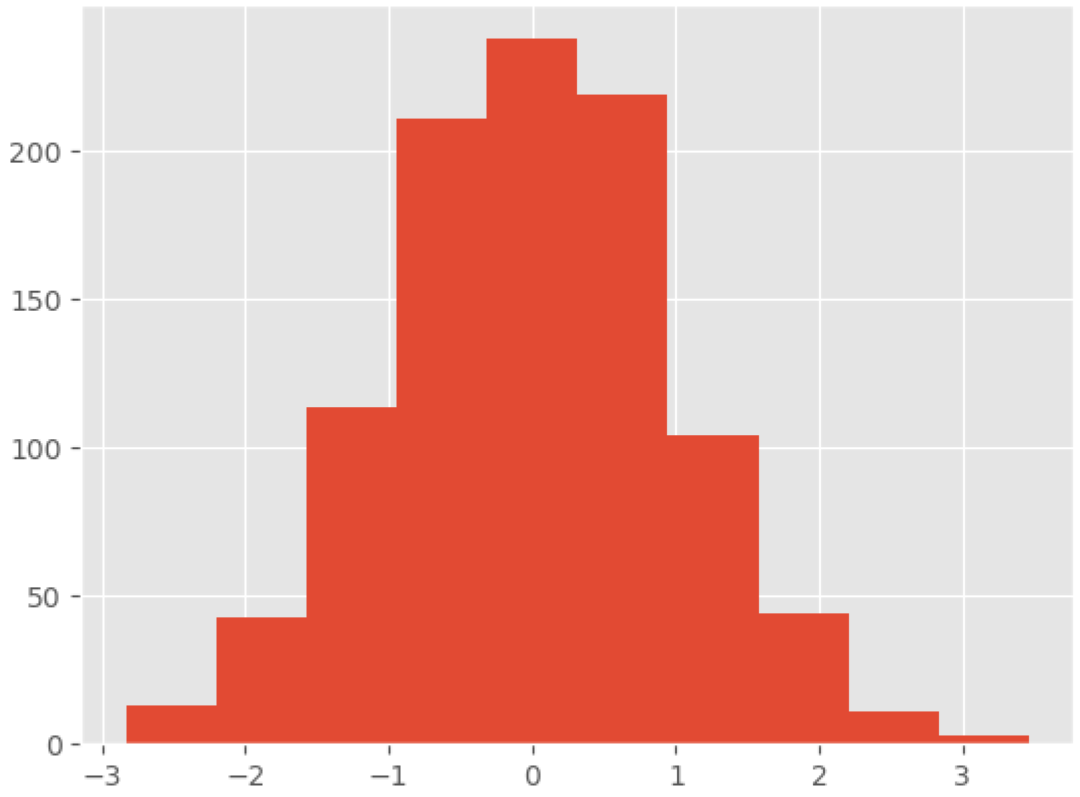
```
In [29]: a = np.random.rand(20)
         print(a)
```

```
[0.06291923 0.2917711  0.83345192 0.70268561 0.29247932 0.42189441
 0.98258318 0.07106039 0.40789724 0.12669976 0.08017007 0.60755141
 0.97814297 0.58792827 0.8761329  0.0674297  0.22863646 0.57869073
 0.83657193 0.49404576]
```

## 4. Create an array with at least 1000 random numbers from normal distributions (normal). Then, plot a histogram of these values (`plt.hist`).

```
In [30]: b = np.random.randn(1000)
         plt.hist(b)
```

```
Out[30]: (array([ 13.,  43., 114., 211., 238., 219., 104.,  44.,  11.,   3.]),
          array([-2.830537  , -2.2014745 , -1.57241199, -0.94334948, -0.31428698,
                  0.31477553,  0.94383803,  1.57290054,  2.20196304,  2.83102555,
                  3.46008805]),
          <BarContainer object of 10 artists>)
```



## Pandas

### 1. Read in a CSV () and display all the columns and their respective data types

In [31]:
```
hw = pd.read_csv('C:/Users/Brett/Desktop/JHU_AAP/ML/mlnn-main-personal/mlnn-main/02/hw
hw.head(20)
```

Out[31]:

| id | sex | weight | height |
|---|---|---|---|
| 1 | M | 190 | 77 |
| 2 | F | 120 | 70 |
| 3 | F | 110 | 68 |
| 4 | M | 150 | 72 |
| 5 | O | 120 | 66 |
| 6 | M | 120 | 60 |
| 7 | F | 140 | 70 |

### 2. Find the average weight

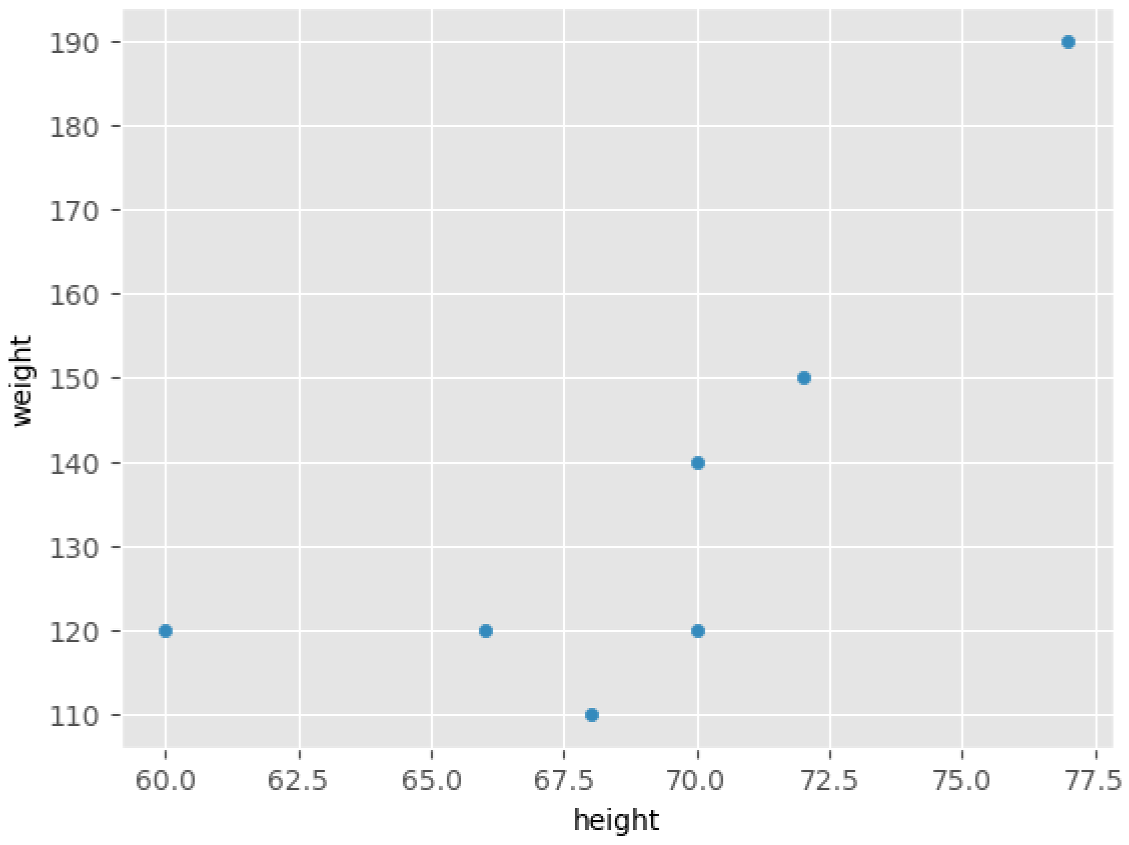
In [32]:
```
hw.weight.mean()
```
Out[32]:
```
135.71428571428572
```

## 3. Find the Value Counts on column `sex`

In [33]:
```
hw.sex.value_counts()
```
Out[33]:
```
M    3
F    3
O    1
Name: sex, dtype: int64
```

## 4. Plot Height vs. Weight

In [34]:
```
hw.plot.scatter(
    x = 'height',
    y = 'weight')
```
Out[34]:
```
<Axes: xlabel='height', ylabel='weight'>
```



## 5. Calculate BMI and save as a new column

In [35]:
```
hw['weight']
hw['height']
bmi = (703*hw['weight'])/(hw['height']**2) # These appear to be imperial units
```

```
hw['bmi'] = bmi
hw
```

Out[35]:

| | sex | weight | height | bmi |
|---|---|---|---|---|
| **id** | | | | |
| **1** | M | 190 | 77 | 22.528251 |
| **2** | F | 120 | 70 | 17.216327 |
| **3** | F | 110 | 68 | 16.723616 |
| **4** | M | 150 | 72 | 20.341435 |
| **5** | O | 120 | 66 | 19.366391 |
| **6** | M | 120 | 60 | 23.433333 |
| **7** | F | 140 | 70 | 20.085714 |

## 6. Save sheet as a new CSV file `hw_dataB.csv`

In [36]:
```
hw.to_csv('hw_dataB.csv')
```

## Run the following (Mac)

In [37]:
```
!cat hw_dataB.csv
```

```
'cat' is not recognized as an internal or external command,
operable program or batch file.
```

## Run the following (Windows)

In [38]:
```
!type hw_dataB.csv
```

```
id,sex,weight,height,bmi
1,M,190,77,22.528250969809413
2,F,120,70,17.216326530612246
3,F,110,68,16.723615916955016
4,M,150,72,20.341435185185187
5,O,120,66,19.366391184573004
6,M,120,60,23.433333333333334
7,F,140,70,20.085714285714285
```