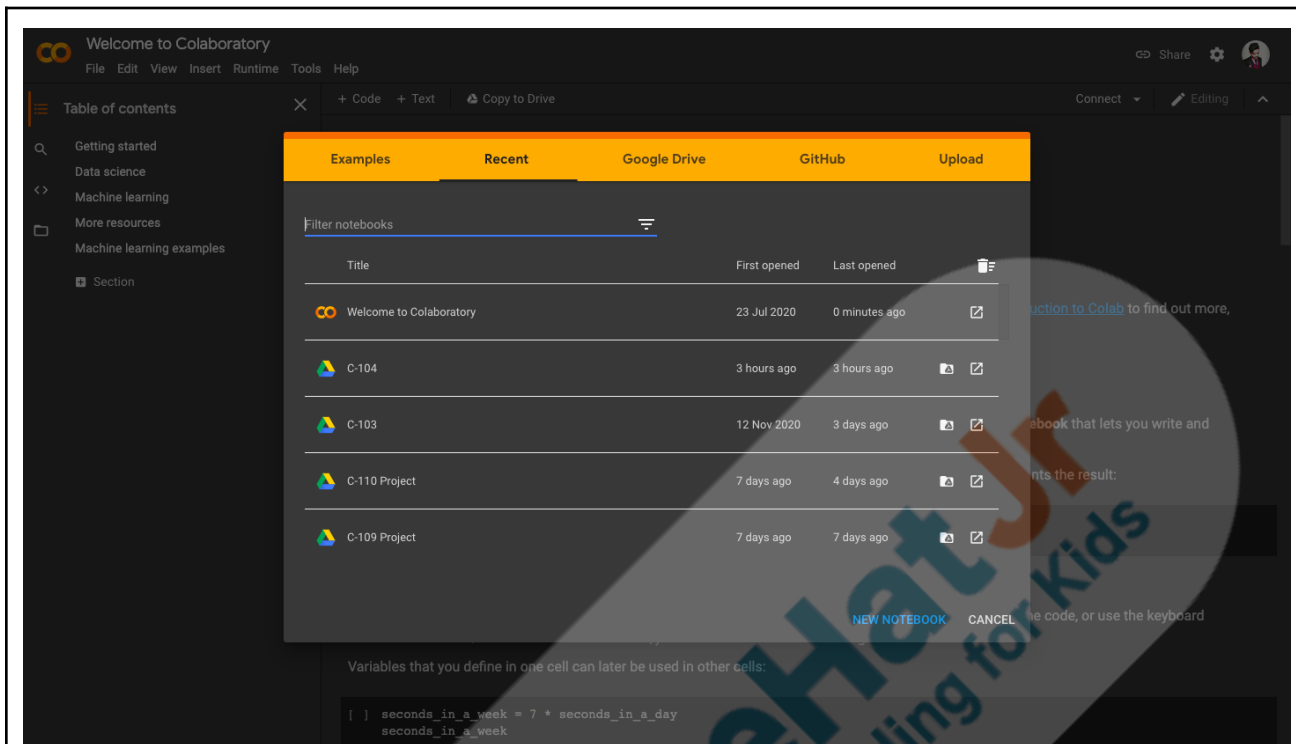


Topic	Mean, Median, Mode	
Class Description	<p>Students learn about the descriptive statistics: mean, median and mode.</p> <p>Students also learn the usage of some python inbuilt function such as Counter, items() and values().</p> <p>Students build a small python program/tool to find mean, median and mode.</p>	
Class	C104	
Class time	45 mins	
Goal	<ul style="list-style-type: none"> <li>Learn about descriptive statistics.</li> <li>Learn about finding mean, median and mode from a given data.</li> <li>Build python programs to get mean, median and mode from a given csv file.</li> </ul>	
Resources Required	<ul style="list-style-type: none"> <li>Teacher Resources               <ul style="list-style-type: none"> <li>Google Colab</li> <li>Laptop with internet connectivity</li> <li>Earphones with mic</li> <li>Notebook and pen</li> </ul> </li> <li>Student Resources               <ul style="list-style-type: none"> <li>Google Colab</li> <li>Laptop with internet connectivity</li> <li>Earphones with mic</li> <li>Notebook and pen</li> </ul> </li> </ul>	
Class structure	<b>Warm Up</b> <b>Teacher-led Activity</b> <b>Student-led Activity</b> <b>Wrap up</b>	<b>5 mins</b> <b>15 min</b> <b>15 min</b> <b>5 min</b>
<p style="text-align: center;"><b><u>CONTEXT</u></b></p> <ul style="list-style-type: none"> <li>Talk about using Descriptive data</li> </ul>		

Class Steps	Teacher Action	Student Action
<b>Step 1: Warm Up (5 mins)</b>	Hi! Do you remember what we learned in the last class?	<b>ESR:</b> - We learned about using plotly express. - We plotted line graph, scatter graph and bar graph using plotly express.
	Great! So we saw how data can be shown using graphs. Let's say when you get your report card you get a mark and percentage. So what does that percentage and marks denotes?	<b>ESR:</b> Marks denote how good we did in the subject and where we need improvement. And percentage denotes how good we did overall.
	So this kind of data is called descriptive data which gives us information. Sometimes we also find a central tendency. Central tendency is a value that tries to describe a data by identifying the central position within the given data. This is also classed as summary statistics. An example of central tendency is finding an average.	-
	Let's learn more about central tendencies in our class. Are you ready?	<b>ESR:</b> Yes.
<b>Teacher Initiates Screen Share</b>		
<p style="text-align: center;"><b><u>CHALLENGE</u></b></p> <ul style="list-style-type: none"> <li>● Get csv data and convert it to list</li> <li>● Use Counter, items(), values() to find mean, median and mode.</li> </ul>		

<p><b>Step 2: Teacher-led Activity (15 min)</b></p>	<p>We will be using Google Colab for this class!</p> <p><i>&lt;Teacher opens a new Google Colab&gt;</i></p> <p><i>&lt;Watch the short introduction video about Colab if the child has not worked with Google Colab before&gt;</i></p> <p><i>&lt;Teacher opens the link from <b>Teacher activity 2</b> and watch the video&gt;</i></p> <p>To open a new google colab, refer to <b>Teacher activity 3.</b></p>	
	<p>In Colab every project is called a <b>notebook</b>. When we open a Colab we see a pop up where we can select our previous notebook to continue our work or create a new notebook to work on a new project. We'll create a new notebook. Here we can write python code as well as text.</p>	



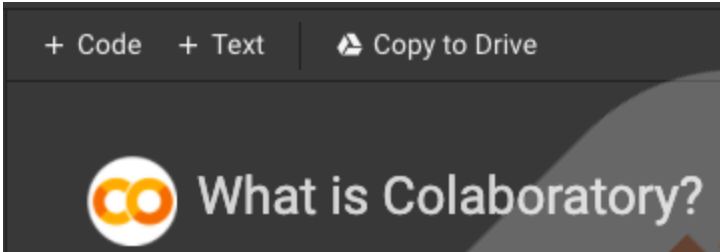
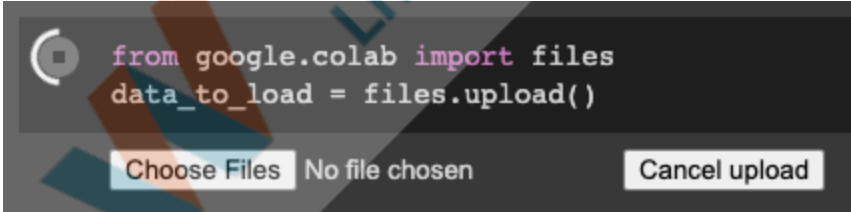
Can you guess how we can write code and text?

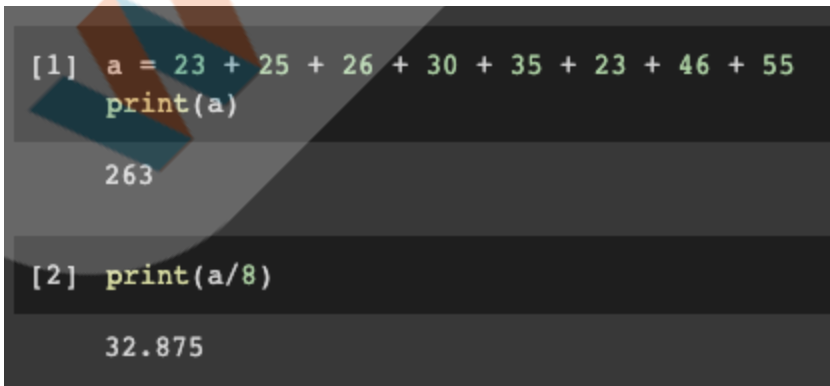
Yes, To write code we click on the code button. A code cell opens up where you can write your code and press the run button to execute your code.

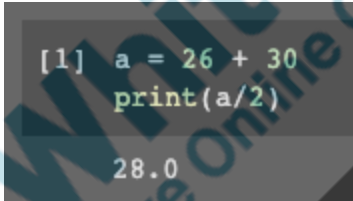
*<The teacher clicks on the code button and types `print("hello world")` in the code cell and clicks on the run button>*

Same way we can add the text in the notebook. Text can be used for general purpose like:

-Adding a heading.

	<ul style="list-style-type: none"> <li>-Adding an explanation on what your code block is doing.</li> <li>-Adding instructions.</li> </ul>	
		
	<p>Uploading and importing files in Colab is also very easy.</p> <p>To upload the files in Colab we just have to write a small piece of code.</p> <p><i>&lt;Teacher writes the following code in code cell&gt;</i></p> <p>Code:-</p> <pre>from google.colab import files data_to_load = files.upload()</pre> <p>a choose file button will appear.</p> <p>by clicking on the button we can upload the files from our local system.</p>	
		
	<p>What meaning comes to your mind when you hear the word "statistics"?</p> <p>And what comes to your mind when you hear the word "descriptive statistics"?</p>	<p><b>ESR:</b></p> <p>Statistics means a collection of data.</p> <p>Descriptive statistics means data which describes something or gives</p>

		information about something.
	<p>Yes! So descriptive statistics is a summary statistics that gives information or tells about the features of the data.</p> <p>Mean, Median and Mode are a part of descriptive statistics.</p>	-
	<p>We also know Mean by another name that is "Average". Can you tell me how we calculate the average?</p>	<p><b>ESR:</b></p> <p>To calculate average we divide the sum of values by the number of values.</p>
	<p>Perfect!! Same way we calculate mean.</p> <p>For example we have a set of numbers 23,25,26,30,35,23,46,55 to find mean:-</p> <p>mean = sum of values / no of values</p> <p>so <math>23+25+26+30+35+23+46+55 / 8</math></p> <p><math>263/8 = 32.875</math></p> <p>So our mean is 32.875</p>	<p><i>Student asks questions about finding mean value.</i></p>
 <pre>[1] a = 23 + 25 + 26 + 30 + 35 + 23 + 46 + 55     print(a) 263  [2] print(a/8) 32.875</pre>		
	<p>Now let's see what is median.</p> <p>&lt;Teacher googles the technical meaning of median and reads it for</p>	<p><i>The student asks questions about finding the median value.</i></p>

	<p><i>the student Note:- Avoid wikipedia&gt;</i></p> <p>So as the meaning says it means situated in the middle or the middle number.</p> <p>To find Median we arrange all the numbers from small to greater and then pick the number in the middle or if there are two middle numbers then we take mean to those two numbers.</p> <p>For example we have our set of numbers 25,30,35,23,46,55,23,26. First we have to sort it from lower to higher. 23,23,25,26,30,35,46,55. Now we find two middle numbers which are 26 and 30. Let's find the mean of it which is <math>(26+30)/2 = 28</math>. So the median is 28.</p>	
		
	We will first upload the data	
	<p>The final is mode. Mode is used to find the most occurring value of a data set.</p> <p>To find mode we just have to find the most occurring element of the data.</p> <p>For example: 23,25,26,30,35,23,46,55 in this data set which is the most occurring number?</p>	<p><b>ESR:</b> 23.</p> <p><i>&lt;Student asks questions about mode&gt;</i></p>

	Yes 23 so that's our mode.	
	<p>Most of our data is going to be in csv files. So we need to read it, get it in a list and from there use it to find mean median and mode.</p> <p>Can you tell me how we open a file and read it?</p>	<p><b>ESR:</b></p> <p>We use with open("filename", "r") as f: to read the file where r stands for read method on the file.</p>
	<p>yes! Here it's a little different let's see how.</p> <p>first we import csv</p> <p>The CSV module is one of the modules in Python which provides classes for reading and writing tabular information in CSV file format.</p> <p>Then we open our csv file</p> <p><b>with open('Internet Users.csv', newline='') as f:</b></p> <p>If we have csv as a text file object then we open it with a newline to avoid python's own line handling module.</p> <p>CSV provides us a csv.reader method which reads and returns the current row and then moves on to the next.</p> <p><b>file_data = list(reader)</b> adds the data to the list.</p> <p>Full code:-</p> <p><b>import csv</b></p> <p><b>with open('height-weight.csv', newline='') as f:</b></p> <p><b>    reader = csv.reader(f)</b></p> <p><b>    file_data = list(reader)</b></p>	<p><i>The student asks questions about using the newline and csv.reader method.</i></p>



```
import csv
with open('height-weight.csv', newline='') as f:
    reader = csv.reader(f)
    file_data = list(reader)
```

Python also has a collection module which has a counter method which is basically a container that keeps track of how many times the same values are added/repeated.

To use this method you simply have to import it and pass the file data to the counter method.

<teacher runs the following code on python shell and shows output>

Code:-

```
from collections import Counter
new_data= "whitehatjr"
data = Counter(new_data)
print(data)
```

*Student observes and asks questions about the counter.*

```
[2] from collections import Counter

new_data = "whitehatjr"
data = Counter(new_data)
print(data)

Counter({'h': 2, 't': 2, 'w': 1, 'i': 1, 'e': 1, 'a': 1, 'j': 1, 'r': 1})
```

Python has a **dictionary.items()** method which is used to return the list with all dictionary keys with values. dictionary.items() returns a tuple of key value pairs.

Code:

```
dictionary.items()
```

*Student asks questions about dictionary items.*

	<p>Teacher runs the following code and shows output.</p> <p>Code:-</p> <pre>new_list = data.items() print(new_list)</pre>	
<pre>[3] new_list = data.items()     print(new_list)  dict_items([('w', 1), ('h', 2), ('i', 1), ('t', 2), ('e', 1), ('a', 1), ('j', 1), ('r', 1)])</pre>		
	<p><b>dictionary.values()</b> returns the list of all the values in the dictionary.</p>	-
<pre>[4] value = data.values()     print(value)  dict_values([1, 2, 1, 2, 1, 1, 1, 1])</pre>		
	<p>Now that you know how to find mean, median and mode, I have a challenge for you to write code to find mean, median and mode. Can you do that?</p>	<p><b>ESR:</b> YES!! &lt;Student takes up the challenge&gt;</p>
<p><b>Teacher Stops Screen Share</b></p>		
	<p>Now it's your turn. Please share your screen with me.</p>	
<ul style="list-style-type: none"> <li>• Ask Student to press ESC key to come back to panel</li> <li>• Guide Student to start Screen Share</li> <li>• Teacher gets into Fullscreen</li> </ul>		
<p align="center"><b>ACTIVITY</b></p> <ul style="list-style-type: none"> <li>• Build a python program to find mean, median and mode from given csv file data.</li> </ul>		

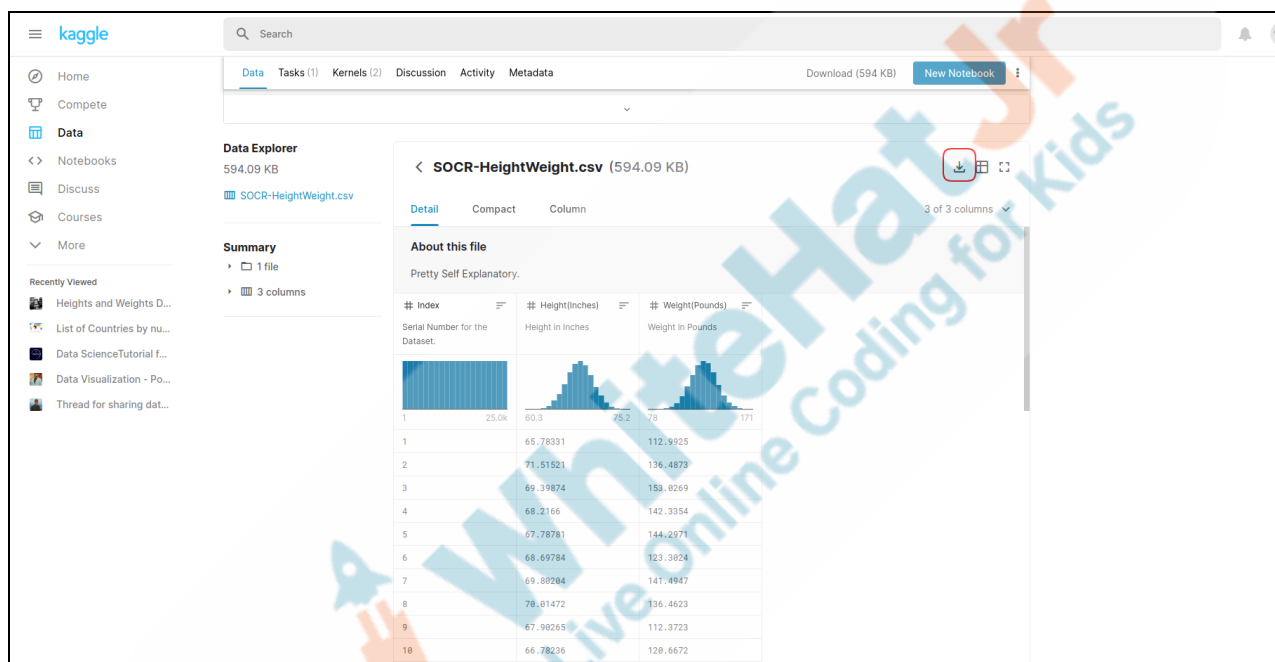
**Step 3:  
Student-Led  
Activity  
(15 min)**

<Teacher guides student towards **Student Activity 2** to open a new Google Colab.>

Teacher helps students to create a file and download the csv file.

<Student open Student Activity 1 and downloads the data>

Student opens the colab and creates a mean.py file and downloads the csv file from **student activity 1**.



Teacher helps the student to write code for reading the csv and storing data in a list.

<Student writes code to read csv file and store data in a list.>

```
[5] from google.colab import files
data_to_load = files.upload()
```

Choose Files height-weight.csv

- height-weight.csv(text/csv) - 608346 bytes, last modified: 21/05/2020 - 100% done  
Saving height-weight.csv to height-weight.csv

```
[6] import csv

with open("height-weight.csv", newline="") as f:
    reader = csv.reader(f)
    file_data = list(reader)
```

Now we need to sort data from **file\_data** to get the list of height from it.

First we remove the title list from the data using **pop()**.

Then create an empty list named **new\_data**.

Then use a for loop on **file\_data** to get the elements inside the nested lists and append them to the **new\_data** list.

<Teacher helps student with the code>

<The student codes to sort data from the given data removes the first list using **pop()**.

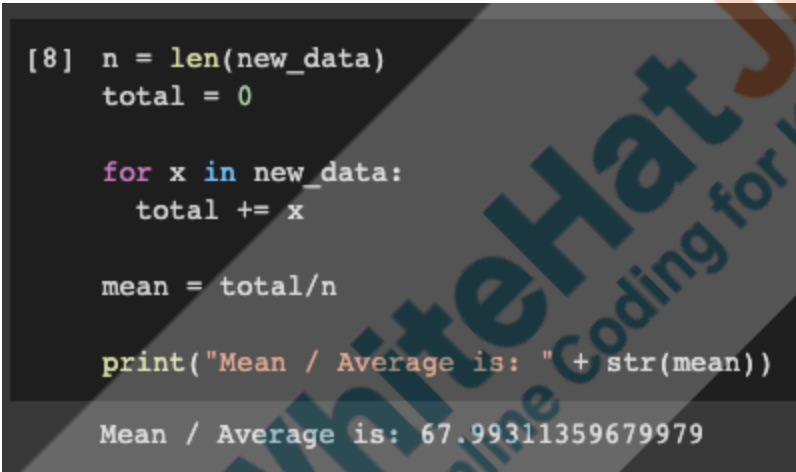
Then creates the empty list named **new\_data**.

Uses for loop on **file\_data** to get the element inside the nested lists and appends them to the **new\_data** list.>

```
[7] file_data.pop(0)

new_data = []

for i in range(len(file_data)):
    n_num = file_data[i][1]
    new_data.append(float(n_num))
```

	<p>How do we find Mean?</p> <p>Prefect. &lt;Teacher helps student find mean&gt;</p>	<p><b>ESR:</b></p> <p>We get the sum of the values and then divide by the number of values.</p> <p><i>Student codes to find the mean and then print it. Then runs the code and tests it.</i></p>
		
	<p>Now we'll code to find the <b>Median</b>. First we need to read the csv and sort the data to get a list of heights.</p> <p>Remember, since we are working in Google Colab, we have already read the csv file and we have retrieved the data for heights.</p> <p>Now we need this data in ascending order of numbers.</p> <p>To do that we'll use a sort() function, this function sorts the data in ascending order for us. And then get the length of the data and store it in n variable.</p>	<p>&lt;The student sorts the heights data in ascending order&gt;</p>

	<p><i>&lt;Teacher helps student to use sort function to get data in ascending order and get the length of data and store it in n variable.&gt;</i></p>	
<pre>[9] new_data = new_data.sort()</pre>		
	<p>The length we took before is to check if the number of values in the data are even or not. Here we use the <b>floor</b> division (floor division is shown by // . It's the same as normal division and the result is always a whole number) to get the whole number. We will also convert the string values to float using the float <b>function</b>.</p> <p>If the length is an even number then there will be 2 values as medians and we'll have to find the mean of those two values.</p> <p>If the length is an odd number then we don't need to find the mean. As the middle number we find will be the median.</p> <p><i>&lt;Teacher helps student to write a if condition to check if the length of the data is even.</i></p> <p><i>If the length is even then we find the mean of the two numbers and the result is the median.</i></p> <p><i>If the length is a odd number then get the middle number as median&gt;</i></p>	<p><i>&lt;The student writes an if condition to check if the length of the data is even. If the length is even then we find the mean of the two numbers and the result is the median.</i></p> <p><i>If the length is an odd number then get the middle number as median&gt;</i></p>

```
#using floor division to get the nearest number whole number
# floor division is shown by //
if n % 2 == 0:
    #getting the first number
    median1 = float(new_data[n//2])
    #getting the second number
    median2 = float(new_data[n//2 - 1])
    #getting mean of those numbers
    median = (median1 + median2)/2
else:
    median = new_data[n//2]

print("Median is: " + str(median))
```

Median is: 67.9957

We wrote code for mean and median, now all that is left is mode. As we know mode is used to find the most frequent occurrence of a value.

We will import Counter from collection and pass the **new\_data** to Counter method and store it in a variable called data.

Then we create a dictionary with a range of heights as keys and the occurrences of the heights as values which at start is 0.

#### ESR:

<Student codes to import Counter from collection, pass **new\_data** to the Counter method and store it in a variable called **data**. Then create a dictionary with range of height as keys and the occurrences of the heights.>

```
[23] #Calculating Mode
data = Counter(new_data)
mode_data_for_range = {
    "50-60": 0,
    "60-70": 0,
    "70-80": 0
}
```

Now we need to separate the data points in the given ranges. To do so we'll use the for loop on data.item(), using if and elif condition we'll check if the height lies between 50-60, 60-70, 70-80.

We also need to convert the string to float using float.

If the height lies in the range the occurrence count will increase.

<Teacher helps student with the code>

Code:-

**for height, occurrence in data.items():**

```
    if 50 < float(height) < 60:
        mode_data_for_range["50-60"]
        += occurrence
    elif 60 < float(height) < 70:
        mode_data_for_range["60-70"]
        += occurrence
    elif 70 < float(height) < 80:
        mode_data_for_range["70-80"]
        += occurrence
```

<The student codes to create 3 empty lists mode, mode1 and mode2.

Using for loop on get\_mode.item(), using if and elif condition we'll check if the height lies between 50-60, 60-70, 70-80.

If the height lies in the range the occurrence count will increase>



```
[24] for height, occurrence in data.items():
    if 50 < float(height) < 60:
        mode_data_for_range["50-60"] += occurrence
    elif 60 < float(height) < 70:
        mode_data_for_range["60-70"] += occurrence
    elif 70 < float(height) < 80:
        mode_data_for_range["70-80"] += occurrence
```

Then declare **mode\_range** and **mode\_occurrence** variables and set its value to 0. Using **for loop** on **mode\_data\_for\_range.items()**. Then we check  
If **occurrence** is greater than **mode\_occurrence** then using **split()** on range we get the first and second values of mode. Then find the mean of the two modes values and print it.

<Teacher helps student with the code>

<The student codes to declare **mode\_range** and **mode\_occurrence** and set its value to 0. Using for loop on **mode\_data\_for\_range.items()**.  
If **occurrence** is greater than **mode\_occurrence** then using **split()** on range we get the first and second values of mode. Then find the mean of the two mode values and print it>

```
[25] mode_range, mode_occurrence = 0, 0
for range, occurrence in mode_data_for_range.items():
    if occurrence > mode_occurrence:
        mode_range, mode_occurrence = [int(range.split("-")[0]), int(range.split("-")[1])], occurrence
mode = float((mode_range[0] + mode_range[1]) / 2)
print(f"Mode is -> {mode:2f}")

Mode is -> 65.000000
```

### Teacher Guides Student to Stop Screen Share

#### FEEDBACK

- Appreciate the student for their class
- Get them to play around with different data to find mean , median and mode.

<b>Step 4:</b> <b>Wrap-Up</b> <b>(5 min)</b>	<p>Let's quickly wrap up today's class. What did we learn?</p>	<b>ESR:</b> <ul style="list-style-type: none"> <li>- We learned about descriptive statistics.</li> <li>- We learned to use inbuilt functions like Counter, items(), values().</li> <li>- We learned to find central tendencies such as mean, mode and median.</li> </ul>
	<p>Now that you know how to find mean, mode and median using code you can experiment with different data sets. In the next class we'll learn about the standard deviation of the two data sets. Are you excited?</p>	<p>Yes!</p>
	<p>Looking forward to the next class then</p>	
<b>Project Overview</b>	<p><b>Mean Median Mode</b></p> <p><b>Goal of the Project:</b></p> <p>In this project you'll have to find the mean, median and mode from the data to determine the average weight of an 18 year old person.</p> <p><b>Story:</b></p> <p>There is a phenomenon in math that if you know 2 values you can find the 3rd value. Help Raju to find an</p>	

	<p>average weight of 18 years old people using such phenomenon.</p> <p>Write a program to find the mean median and mode of the given data</p> <p>.</p> <p>I am very excited to see your project solution and I know you will do really well.</p> <p>Bye Bye!</p>	
<div>Teacher Clicks</div> <div>✕ End Class</div>		
<b>Additional Activities</b>	<p><i>Encourage the student to write reflection notes in their reflection journal using markdown.</i></p> <p>Use these as guiding questions:</p> <ul style="list-style-type: none"> <li>• What happened today? <ul style="list-style-type: none"> <li>- Describe what happened</li> <li>- Code I wrote</li> </ul> </li> <li>• How did I feel after the class?</li> <li>• What have I learned about programming and developing games?</li> <li>• What aspects of the class helped me? What did I find difficult?</li> </ul>	<p><i>The student uses the markdown editor to write her/his reflection in a reflection journal.</i></p>

Activity	Activity Name	Links
----------	---------------	-------

Teacher Activity 1	Reference code	<a href="https://colab.research.google.com/drive/1Dg-ZKMge4G4qERq_n7WV2KFMjmjXbL4S?usp=sharing">https://colab.research.google.com/drive/1Dg-ZKMge4G4qERq_n7WV2KFMjmjXbL4S?usp=sharing</a>
Teacher Activity 2	Colab Introduction	<a href="https://youtu.be/inN8seMm7UI">https://youtu.be/inN8seMm7UI</a>
Teacher Activity 3	Google Colab Link	<a href="https://colab.research.google.com/">https://colab.research.google.com/</a>
Student Activity 1	height weight data set	<a href="https://www.kaggle.com/burnoutminer/heights-and-weights-dataset">https://www.kaggle.com/burnoutminer/heights-and-weights-dataset</a>
Student Activity 2	Google Colab Link	<a href="https://colab.research.google.com/">https://colab.research.google.com/</a>