| Topic | Functions | |
|---|---|---|
| **Class Description** | **Students learn how to use pre-defined functions in python. Students also learn how to write their own functions. Students write a function which takes a filename from the user and outputs the number of words in the file.** | |
| **Class** | **C98** | |
| **Class time** | **45 mins** | |
| **Goal** | • Write a custom defined function in Python<br>• Write a program which takes file path from the user and prints the number of words in the file | |
| **Resources Required** | • Teacher Resources<br>  ○ Laptop with internet connectivity<br>  ○ Earphones with mic<br>  ○ Notebook and pen<br>• Student Resources<br>  ○ Laptop with internet connectivity<br>  ○ Earphones with mic<br>  ○ Notebook and pen | |
| **Class structure** | **Warm Up**<br>**Teacher-led Activity**<br>**Student-led Activity**<br>**Wrap up** | **15 min**<br>**15 min**<br>**5 min**<br>**5 min** |

| Class Steps | Teacher Action | Student Action |
|---|---|---|
| **Step 1: Warm Up (15 min)** | Hello <student name>!<br>Last class, we were introduced to this new and powerful general purpose language called Python.<br>Do you remember what we learned? | **ESR:**<br>We learned the python syntax to create variables, conditional programming , writing loops etc. |

| | | |
|---|---|---|
| | Great! We also wrote a program which takes input from the user and tells us the number of words in the message input by the user.<br><br>This class, we will be learning about functions in python. We will learn about some of the functions which are pre-defined in python. We will also learn about how to write custom functions.<br>Can you remember and connect with functions which we used in javascript? What are functions? | **ESR:**<br>Functions are blocks of code which can be invoked inside the main program. Functions can take some arguments and use them to do some specific task. |
| | Have we used functions in python so far? | **ESR:**<br>Yes!<br>input()<br>print() |
| | print() is a predefined function in python. We will learn more about the other predefined functions in python. We will also learn how we can write our own function. Let's get started. | - |
| **Teacher Initiates Screen Share** | | |
| **Step 2: Teacher-led Activity (15 min)** | You have already used two important pre-defined functions in python - print() and input().<br><br>Let's learn a few more pre-defined functions which we will be using in our program. | |

| | | |
|---|---|---|
| | | |
| | One of the pre-defined functions in python is called open().<br><br>open() function can be used to open any file in read, write or append mode. The function returns a file object which we can read in our program.<br>Let's quickly look at the reference document for open() function in python.<br><Teacher opens the doc from Teacher Activity 1> | *Student looks at the documentation for open. Student Activity 1.* |
| | Let's try to use the open() function to open a file from our current folder.<br><br>1. Teacher creates a file called "test.txt".<br>2. Teacher opens python console and uses f = open("test.txt") to store the file in 'f' variable.<br>3. Teacher uses f.read() to read the file and then prints it on the console.<br><br>**Note:**<br>- By default a file opens in read mode. We could have also used open("test.txt",'r') to open the file explicitly in the read mode.<br>- File is an object in python which has several functions defined on them. read() is a function defined on files. | *The student learns how to use the open() function.* |

| | | |
|---|---|---|
| | - "\n" represents the Enter or return key and add a new line. | |

```
rajeev@atlantis ~/D/gettingFilesList> python3
Python 3.6.9 (default, Apr 18 2020, 01:56:04)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> f = open("test.txt")
>>> f.read()
'This is a test file. This is a sample text.\n'
>>>
```

| | | |
|---|---|---|
| | You can also print a file line by line in python.<br><br>*Teacher writes code to print the file line by line.*<br><br>**f = open("test.txt")**<br>**fileLines = f.readlines()**<br><br>readlines() defined on file object splits the lines in the file and stores it line by line as a list. We can print these lines by calling a for loop on the list.<br><br>for line in fileLines:<br>  print(line) | *The student understands how to use functions defined on file objects to read the file line by line.* |

```
>>> f = open("test.txt")
>>> fileLines = f.readlines()
>>> for line in fileLines:
...     print(line)
...
This is a test file. This is a sample text.

>>>
```

| | | |
|---|---|---|
| | There are several other functions defined on files which you can explore on your own by referring to the documentation. | *The student can look at the documentation for functions defined on the file.* |
| | We saw that a file is an object in python.<br>In fact, everything in python is an object. For example - a variable which holds a string is also an object in python.<br>Let's look at it.<br><br>Let's store a string inside a variable.<br><br>introString = "My name is Wily, I am 15 years old."<br><br>introString is a string object on which several functions are defined.<br>One of the functions defined on introString is called split().<br><br>Can you guess what split() does? | **ESR:**<br>It will split the text into words. |
| | Yes! split() splits the given string by their whitespaces, gives each individual word and stores them in a list.<br><br>We can write:<br><br>words = introString.split()<br>print(words) | *Student sees the output of split() used over a string.* |

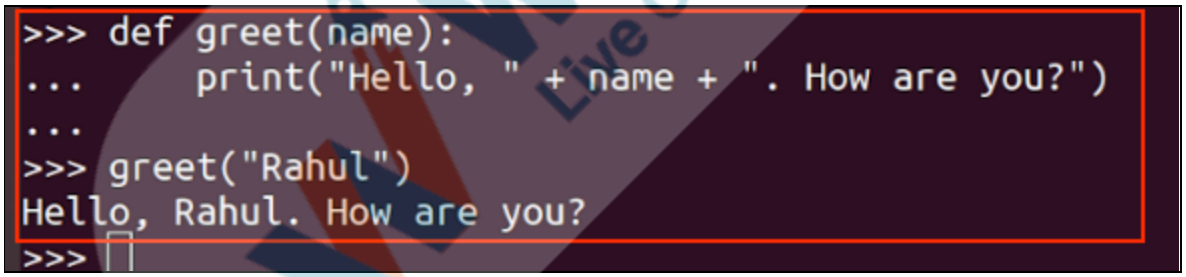
<table>
<tr><td colspan="3">

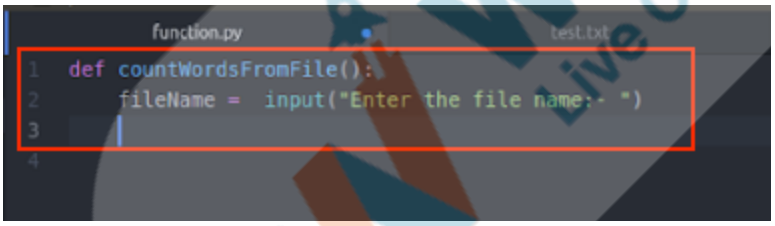
```
>>> introString = "My name is Wily, I am 15 years old."
>>> words = introString.split()
>>>
>>> print(words)
['My', 'name', 'is', 'Wily,', 'I', 'am', '15', 'years', 'old.']
>>>
```

</td></tr>
<tr><td></td><td>split() splits using whitespace by default.<br><br>You can also specify which separator to use to split the items.<br>For example, we could use comma (,) as the separator:<br><br>phrases = introString.split(",")</td><td>The student learns how to use split() with a custom separator.</td></tr>
<tr><td colspan="3">

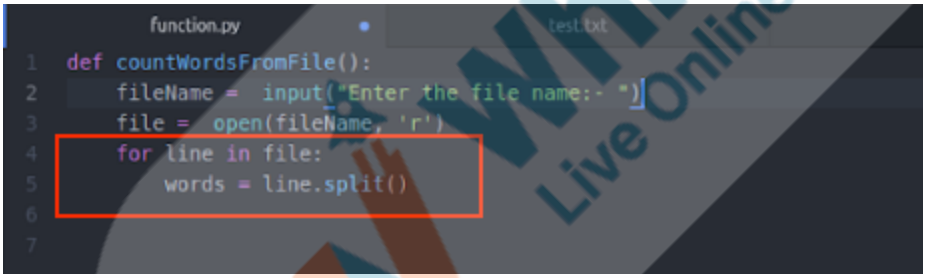
```
>>> introString = "My name is Wily, I am 15 years old."
>>> words = introString.split(",")
>>> print(words)
['My name is Wily', ' I am 15 years old.']
>>>
```

</td></tr>
<tr><td></td><td>Again, there are several functions defined on string objects in Python. You can look over them in the documentation for python.</td><td>The student sees the documentation for string functions in python.</td></tr>
<tr><td></td><td>Now, we have used several functions which are pre-defined in python.<br><br>Let's learn how to define a custom function in python. You already know how to define a custom function in javascript. It is similar in python except for the change in syntax.</td><td>The student understands the syntax for defining functions in python.</td></tr>
</table>

| | | |
|---|---|---|
| | To define a function in python, we use the syntax:<br><br>**def functionName():**<br><block of code for the function><br><br>You can also pass arguments to your function inside the brackets - similar to how we did in javascript.<br><br>**def functionName(variable):**<br><block of code for the function> | |
| | Let's quickly try to define a function greet() which takes name as an input and greets the person by name.<br><br>*Teacher writes code to create the function greet() which takes name as an input and prints a greeting with the person's name.* | *The student helps in writing the function greet().* |
|  | | |
| | In the last class we wrote code to count the number of words and characters from a string input by the user.<br>Can you write a program which takes the path of a file from the user and outputs the number of words used in the file? | **ESR:**<br>Yes!<br><br>There are a lot of places where there is a restriction on the number of words to be used - for example while writing essays....we can use |

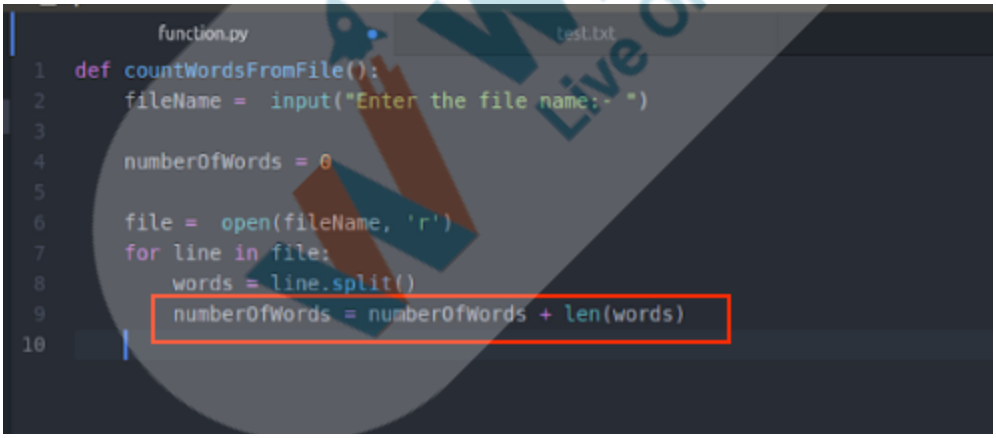
| | | |
|---|---|---|
| | Where could such a function be useful? | the program to count the number of words used. |
| | Let's get started. | |
| **Teacher Stops Screen Share** | | |
| | Now it's your turn. Please share your screen with me. | |
| • **Ask Student to press ESC key to come back to panel**<br>• **Guide Student to start Screen Share**<br>• **Teacher gets into Fullscreen** | | |
| **Step 3: Student-Led Activity (5 min)** | Great! Let's get started.<br><br>Let's start by creating a countWordsFromFile.py file. Inside the file, let's create a custom function countWordsFromFile() which takes the file path as input. | **ESR:**<br>We can define a function using def countWordsFromFile().<br><br>*<Student write code to define the function>* |
|  | | |
| | We can open the file using open() in read mode. | *The student writes code to open the file name.* |


| | | |
|---|---|---|
| | We can read all the lines in our file and store it in a list.<br><br>For each line in the file, let's split the line by whitespace to store all the words in the line as a list. | *The student writes code to read all the lines in a file.*<br><br>*The student uses a for loop to run for each line in the file.*<br>*For each line in the file, the student uses split() to split the line by whitespace and store the words in the words list.* |




| | | |
|---|---|---|
| | Great! Now how do we count the number of words in each line and in the file overall? | **ESR:**<br>We use a numberOfWords variable which starts from 0. We count the number of words in the words list for each file and keep adding to the numberOfWords. |


| | | |
|---|---|---|
| | There is a predefined function len() in python which will give you the length of the list. We can use it to find the length of words listed for each line. | *Student declares numberOfWords =0. Student uses numberOfWords = numberOfWords + len(words) to increment the value of count by the number of words in each line of the file. Student returns the count from the function.* |



| | | |
|---|---|---|
| | Now let's print the number of words and call the function. | The student writes the print statement to print the number of words and calls the countWordsFromFile function. |

function.py test.txt

```
1  def countWordsFromFile():
2      fileName =  input("Enter the file name:- ")
3
4      numberOfWords = 0
5
6      file =  open(fileName, 'r')
7      for line in file:
8          words = line.split()
9          numberOfWords = numberOfWords + len(words)
10     print("Number of words:")
11     print(numberOfWords)
12
13
14 countWordsFromFile()
15
```

| | | |
|---|---|---|
| | Let's run the code on the terminal . Open the terminal and go to the folder which contains your function.py file. Run the command **python3 function.py.** | *Student opens the terminal and goes to the folder which contains the function.py file and then runs the command **python3 function.py.*** |

```
~/D/gettingFilesList> python3 function.py
Enter the file name:- test.txt
Number of words:
10
```

| Teacher Guides Student to Stop Screen Share | | |
|---|---|---|
| **Step 4: Wrap-Up** | Amazing! I would encourage you to explore more about file objects. You can also use them to write to create a file and write to it through code!<br><br>Do you think we can also learn how to move our files from one location to another programmatically! | **ESR:** varied |
| | Of course we can! We can do anything on our system using python. In fact in the coming few classes, we will be creating a little program using which you will be able to back up your files in any folder automatically to a remote cloud! | - |
| **Project Overview** | **Functions**<br><br>**Goal of the Project:**<br><br>Today you have learned to write functions in python.<br><br>**Story:**<br><br>Rajini loves to play tricks on people. | |

| | | |
|---|---|---|
| | Her brother has written an essay in a computer file. Rajini wants to trick him by swapping the content of his file with some other file So she is seeking your help.<br>Create a program to swap the data inside the files.<br><br>I am very excited to see your project solution and I know you will do really well.<br><br>Bye Bye! | |
| **Teacher Clicks**   | | |
| **Additional Activities** | *Encourage the student to write reflection notes in their reflection journal using markdown.*<br><br>Use these as guiding questions:<br>● What happened today?<br>○ Describe what happened<br>○ Code I wrote<br>● How did I feel after the class?<br>● What have I learned about programming and developing games?<br>● What aspects of the class helped me?<br>● What did I find difficult? | *The student uses the markdown editor to write her/his reflection as a reflection journal.* |

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

| | | |
|---|---|---|
| Teacher Activity 1 | Reading and writing files doc | https://realpython.com/read-write-files-python/ |
| Teacher Activity 2 | Split method doc | https://www.bitdegree.org/learn/python-split |
| Teacher Activity 3 | reference code | https://github.com/whitehatjr/Functions |
| Student  Activity 1 | Reading and writing files doc | https://realpython.com/read-write-files-python/ |
| Student Activity 2 | Split method doc | https://www.bitdegree.org/learn/python-split |