**CSF 2113 Lab 1.1: Python Basics Revision**

**Using Expressions Operators in Python**

1. **Variable Names**

Which of the following are valid/invalid variable names in python

|  |  |
| --- | --- |
| **Variable Name** | **Valid/Invalid** |
| abc | Yes |
| \_abc | Yes |
| abc\_ | Yes |
| 1\_abc\_2 | No |
| 1abc | No |
| \_1abc | Yes |
| if | No deserved word |
| False | No deserved word |
| false | yes |
| @abc | no |
| abc@ac | yes |
| abc$1 | no |
| while | No deserved word |

1. **Expressions and Operators**

Write down the output of the code segments.

|  |  |
| --- | --- |
| **Code Segment** | **Output** |
|  | 6 |
|  | 8 |
|  | -11 |
|  | 5 |
|  | 9.0 |
|  | 2.0 |
|  | 0 |
| 11111 11111 11111 11111 111 | 3 |
|  | 4.6 |
| int | 4 |
|  | 8 |
|  | 7 |
|  |  |
|  | False  False  True  True  False  True  True  False |
|  |  |
|  |  |
|  |  |
|  | Int |
|  |  |

1. **Write following small programs in Python:**

|  |
| --- |
| 1. Write a Python program that asks the user to enter his name then displays a message on the screen to welcome the user: (See Example Run of a program) |
| Paste the code/screenshot here |
| 1. Get the following inputs from the user: first name, last name, age. Then print a message such as:   Greetings Mr. <First name> <Last name>. You are <age> years old. Next year you will be <age>+1 years old. |
| Paste the code/screenshot here |
| 1. Write a program which take marks of three courses as input from user, then calculate and display the total marks of all quizzes. (See Example Run of a program) |
| Paste the code/screenshot here |
| 1. Write a python program which ask user to enter the length and width of a rectangle, then calculate and display the area of the rectangle. User can enter the value in real numbers like 2.15: (See Example Run of a program) |
| Paste the code/screenshot here |
| 1. Write a python program which ask user to enter the radius of a circle, then calculate and display the area and circumference of circle. User can enter the value in real numbers like 2.15:   Area = pi\*r\*r  Circumference = 2\*pi\*r (where pi = 3.14)  (See Example Run of a program) |
| Paste the code/screenshot here |
| 1. The volume of a sphere with radius r is 4/3 πr^3. What is the volume of a sphere with radius 5? The value of π is 3.14159 |
| Paste the code/screenshot here |
| 1. Make a switchboard for a calculator that looks like this:   ------------------------------------------------------------------------------  Type one of the following options:  1 for Addition 2 for subtraction  3 for multiplication 4 for division  ------------------------------------------------------------------------------ |
| Paste the code/screenshot here |

END

**CSF 2113 Lab ٢.1: Containers (Lists)**

# Using Lists in Python

## 1. Creation/Initialization

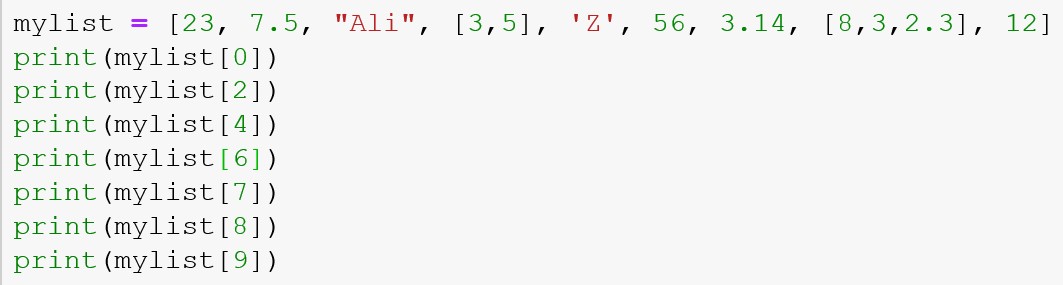
What is the output of following python code?

|  |
| --- |
|  |
| [6, 2, 8] <class 'list'>  [3.2, 5.4, 2.0] <class 'list'>  ['Ali', 'Mohammed', 'Sualeh'] <class 'list'>  [2.5, 2, 'UAE'] <class 'list'>  [2.5, 3, 'Sharjah', [3, 5, 6]] <class 'list'> |

## 2. Indexing

Write down the

output of following code segment:



23

Ali

Z

3.14

[8, 3, 2.3]

12

IndexError

## 3. Slicing

Write down the output of following code segment:

|  |
| --- |
|  |
| [23, 7.5, 'Ali', [3, 5]]  [23, 7.5, 'Ali', [3, 5], 'Z']  ['Z', 56, 3.14, [8, 3, 2.3], 12]  [3.14, [8, 3, 2.3], 12]  [[8, 3, 2.3], 12]  Negative  [3.14, [8, 3, 2.3]]  [3.14, [8, 3, 2.3], 12]  ['Ali', [3, 5], 'Z', 56]  [23, 7.5, 'Ali', [3, 5], 'Z']  [23, 7.5, 'Ali']  [23, 7.5]  [] |

# List Mutability

## 1. Changing Value of the existing element

What is the output of following python code?

|  |
| --- |
|  |
| ['Fatima', 7.5, 'Ali', [3, 5], 'Z']  ['Fatima', 7.5, 'Ali', [3, 5], 4.3]  ['Fatima', 7.5, 8.5, [3, 5], 4.3]  ['Fatima', [8.5, [3, 5]], 8.5, [3, 5], 4.3] |

**2. Appending, Inserting and Removing elements**

Write down the output of following code segment:

|  |
| --- |
|  |
| [6, 5, 8]  [6, 5, 8, 2]  [6, 5, 8, 2, 6, 5]  [6, 5, 8, 2, 6, 5, [12, 13]]  [6, 5, 8, 7.5, 2, 6, 5, [12, 13]]  [[3, 4], 6, 5, 8, 7.5, 2, 6, 5, [12, 13]]  [[3, 4], 6, 8, 7.5, 2, 6, 5, [12, 13]]  [[3, 4], 8, 7.5, 2, 6, 5, [12, 13]] |

## 3. Sorting and Reversing

Write down the output of following code segment:

|  |
| --- |
|  |
| [3, 8, 9, 2, 6]  [6, 2, 9, 8, 3]  [3, 8, 9, 2, 6] |
|  |
| [2, 3, 6, 8, 9]  [6, 2, 9, 8, 3]  [2, 3, 6, 8, 9]  [9, 8, 6, 3, 2] |

# Searching in List

## 1. Searching in List

What is the output of following python code?

|  |
| --- |
|  |
| True  False  True  True  False |

# Creating and populating list

## 1. Creating list by different methods

What is the output of following python code?

|  |
| --- |
|  |
| [] 0  [3] 1  [3, 31] 2  [3, 31, 9] 3 |

|  |
| --- |
|  |
| [] 0  [6] 1  [6, 9] 2  [6, 9, 3, 10] 4 |
|  |
| ['Ali', 'Faisal', 'Ahmed']  ['Ali', 'Faisal', 'Ahmed', 'Noora']  ['6', '5', '2', '3'] |
|  |
| [0, 0, 0] 3  [0, 0, 0, [3, 2]] 4 |

[0, 0, 0, [3, 2], 0, 0, 0, [3, 2]] 8

**End of Lab**

**CSF 2113 Lab 2.2: Containers (Tuple & Dictionaries)**

# Using Tuples in Python

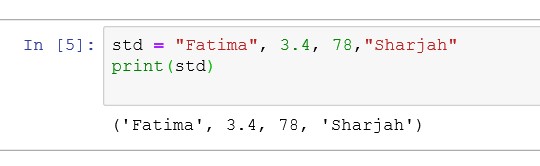
**Tuples are a collection of data items. They may be of different types. Tuples are immutable**

**(like strings). Python optionally uses brackets () to denote tuples**

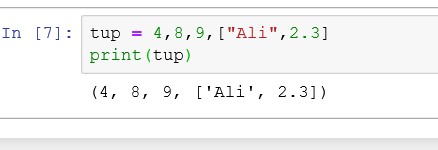
**We could have also used () for the above tuple**

**If we have only one item, we need to use a comma to indicate it's a tuple: e.g. (“Bat”,)**

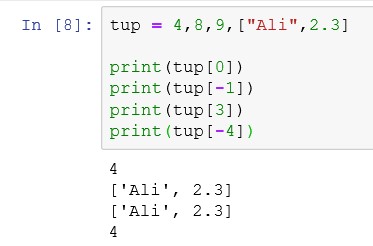
1. **Creation:** Create a touple of following information about a student. Name the tuple std. “Fatima”, 3.4, 78,”Sharjah”



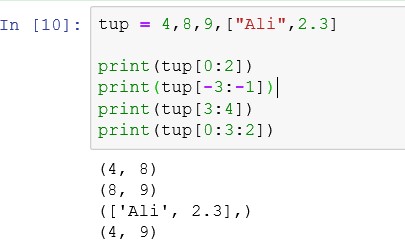
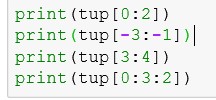
1. Create a tuple where one element is a list and rest are numbers:



1. **Indexing**: Tuple elements can be accessed by index: try out following element in above created lists: tup[0] tup[-1] tup[3] tup[-4]



1. **Slicing:** All slice operations return a new tuple containing the requested elements. This means that the following slice returns a new (shallow) copy of the tuple. Perform following slicing operations on above created tuple:

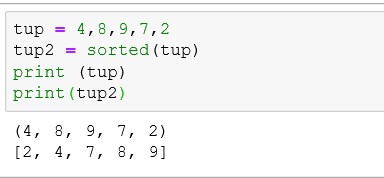
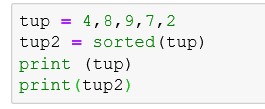


1. **Mutable**: Tuples are immutable. We cannot change the value of an index. Try out following.

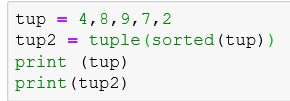


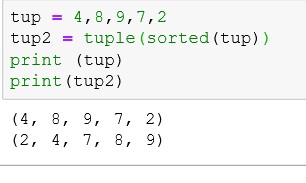


1. **Sorting a Tuple**: As tuple are immutable sorting a tuple is not possible however we can sort by using sorted function which not change the tuple will rather return a sorted list of elements of tuple.



1. **Sorting a Tuple**: We can use this sorted list to create another tuple.





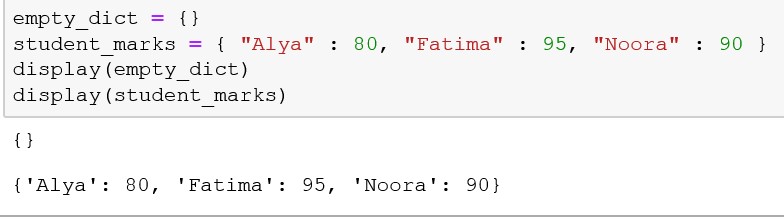
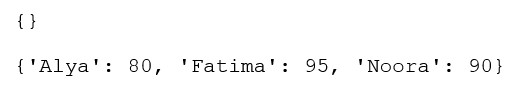
1. **Other functions of tuple**: We can use variety of functions with tuple.

|  |  |  |  |
| --- | --- | --- | --- |
| **Function** | **purpose** | **code** | **output** |
| length | Total length of tuple |  | 5 |
| concatena  tion | Concatena te two or  more  tuples |  | (4, 8, 9,  'sharjah', 'Dubai') |
| Repetition | Repeat the sequence  of  element |  | (4, 8, 9, 4,  8, 9, 4, 8, 9) |
| Members  hip | Check the element is present in tuple |  | True False |
| Max | Find  maximum  value in tuple |  | 9 |
| Min | Find  minimum  value in tuple |  | 4 |

Using Dictionaries in Python

**A dictionary is like a list, but more general. In a list, the indices have to be integers; in a dictionary they can be (almost) any type. Keys must be *unique* within a dictionary: No *duplicates.* Simply put, a dictionary is a list of key-value pairs.**

1. Create two dictionary variables: One an empty dictionary and one with student’s marks as shown in the image below. Than Display the contents of dictionaries.



1. Write down the output of the following python code segment.

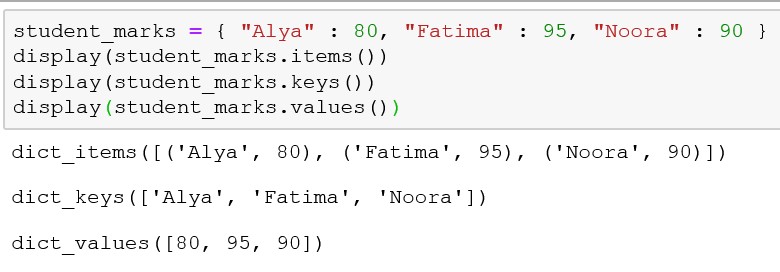
|  |
| --- |
|  |
| 80  95  KeyError (as key “Noora” is misspelled) |

1. Create a dictionary with three items as per given screenshot.



Display following:

* + List of all items in the dictionary “student\_marks”
  + List of all keys in the dictionary “student\_marks”
  + List of all the values in the dictionary “student\_marks”



1. Write down the output of the following code segment.

|  |
| --- |
|  |
| True  False  True  False |

1. Write down the output of the following code segment.

|  |
| --- |
|  |
| 80 90 95 |

1. Write down the output of the following code segment.

|  |
| --- |
|  |
| {'Alya': 80, 'Fatima': 95, 'Noora': 90} 3  {'Alya': 85, 'Fatima': 95, 'Noora': 90} 3 |

{'Alya': 85, 'Fatima': 95, 'Noora': 90, 'Noura': 75} 4

1. Write down the output of the following code segment.

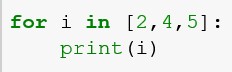
|  |
| --- |
|  |
| {'Alya': 80, 'Fatima': 95, 'Noora': 90} 3  {'Alya': 67, 'Fatima': 95, 'Noora': 90} 3  {'Alya': 67, 'Fatima': 95, 'Noora': 90, 'Sara': 82} 4 |

**End of Lab**

**CSF 2113 Lab 3.1 Iterations Solution**

# for loop statement in Python using a list of values

1. What is the output of following code segment?

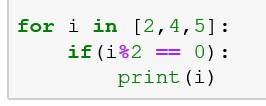


2

4

5

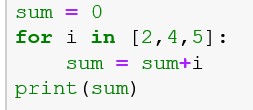
1. What is the output of following code segment?



2

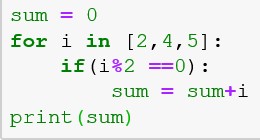
4

1. What is the output of following code segment?



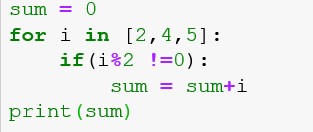
11

1. What is the output of following code segment?



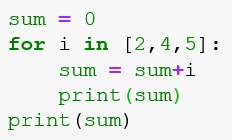
6

1. What is the output of following code segment?



5

1. What is the output of following code segment?



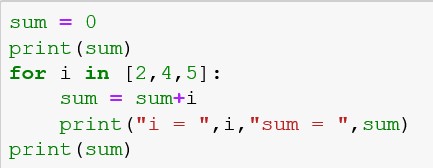
2

6

11

11

1. What is the output of following code segment?

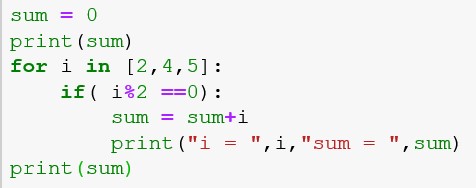


0 i = 2 sum = 2 i = 4 sum = 6

i = 5 sum = 11

11

h. What is the output of following code segment?

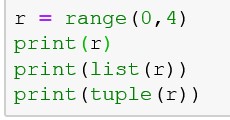


0 i = 2 sum = 2 i = 4 sum = 6

6

# for loop statement in Python using range

1. What is the output of following code segment?

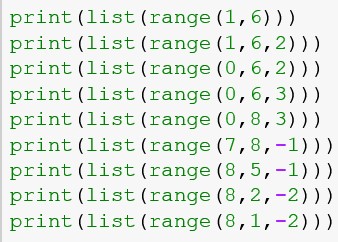


range(0, 4)

[0, 1, 2, 3]

(0, 1, 2, 3)

1. What is the output of following code segment?



[1, 2, 3, 4, 5]

[1, 3, 5]

[0, 2, 4]

[0, 3]

[0, 3, 6]

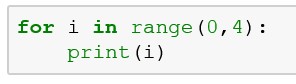
[]

[8, 7, 6]

[8, 6, 4]

[8, 6, 4, 2]

1. What is the output of following code segment?



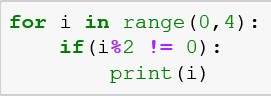
0

1

2

3

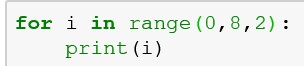
1. What is the output of following code segment?



1

3

1. What is the output of following code segment?



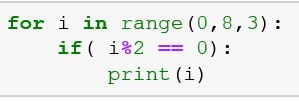
0

2

4

6

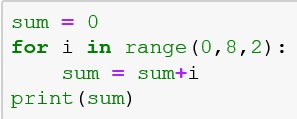
1. What is the output of following code segment?



0

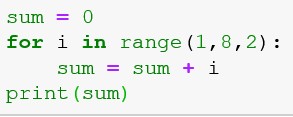
6

1. What is the output of following code segment?



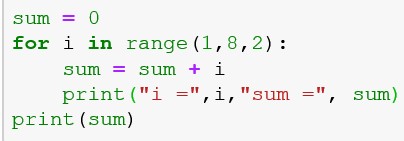
12

1. What is the output of following code segment?



16

1. What is the output of following code segment?

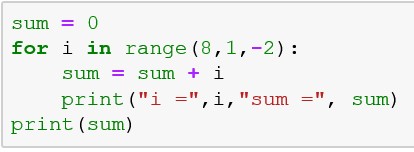


i = 1 sum = 1 i = 3 sum = 4 i = 5 sum = 9

i = 7 sum = 16

16

j. What is the output of following code segment?



i = 8 sum = 8 i = 6 sum = 14 i = 4 sum = 18

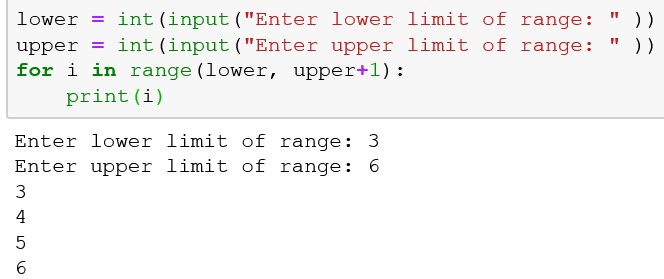
i = 2 sum = 20

20

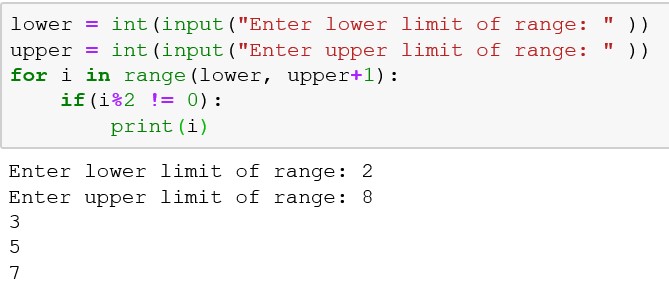
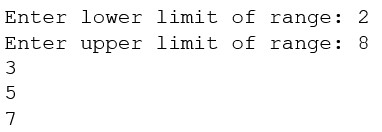
# for loop statement in Python using range

1. Write a program which take a range from the users and display all numbers in the range.

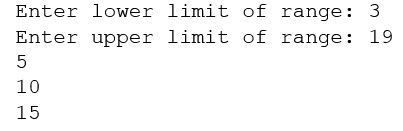


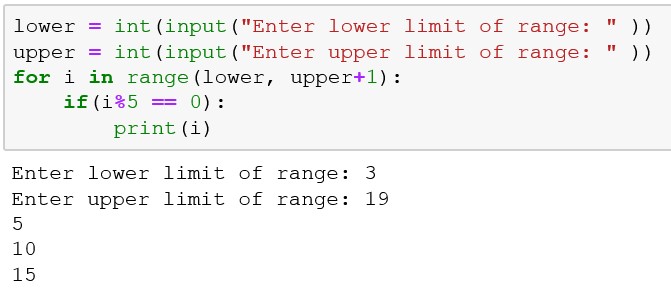


1. Write a program which take a range from the users and display all odd numbers in the range.

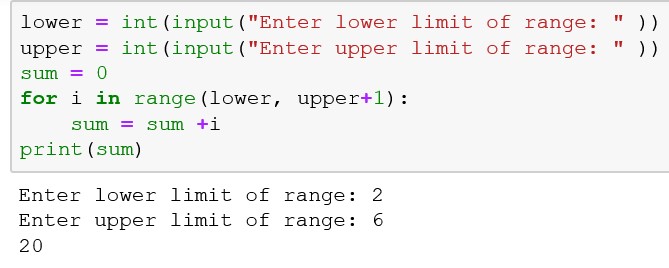
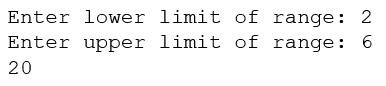


1. Write a program which take a range from the users and display all numbers which are multiple of 5 in the given range.



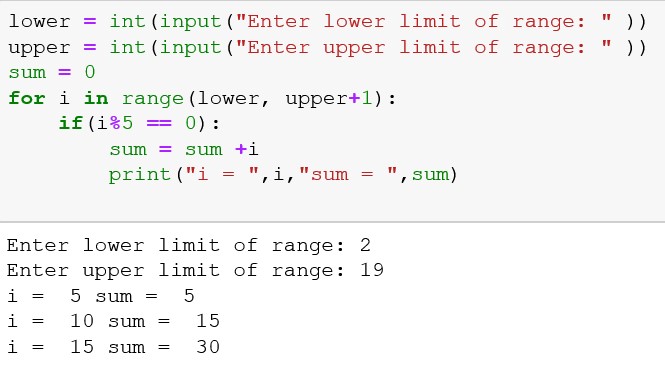


1. Write a program which take a range from the users and display sum of the numbers in the given range?



1. Write a program which take a range from the users and display sum of the all numbers which are multiple of 5 in the given range?

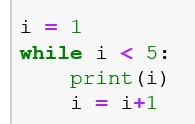




**End of Lab**

**CSF 2113 Lab 3.2 Iterations (while loops) Solution**

1. **Using while loop in python**
   1. What is the output of following code segment?



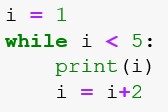
1

2

3

4

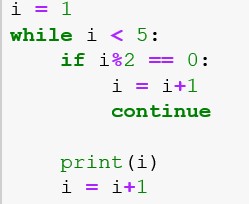
* 1. What is the output of following code segment?



1

3

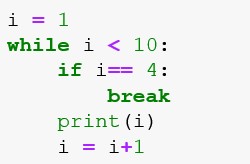
* 1. What is the output of following code segment?



1

3

* 1. What is the output of following code segment?

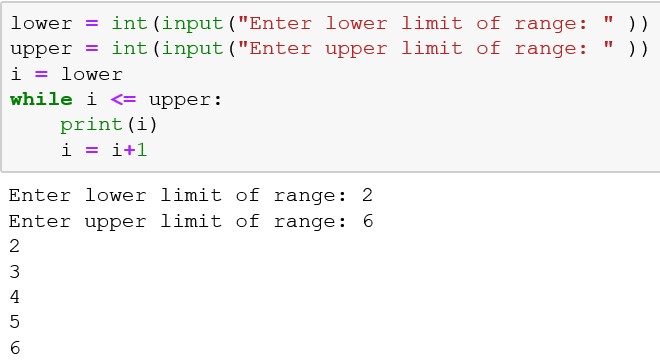


1

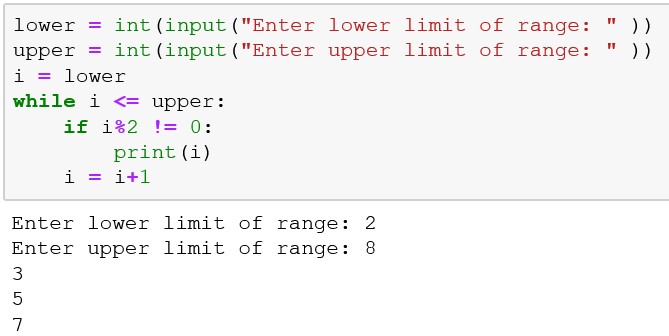
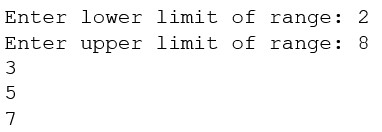
2

3

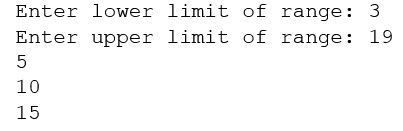
1. **while loop statement in Python**
2. Write a program (using while loop) which take a range from the users and display all numbers in the range.

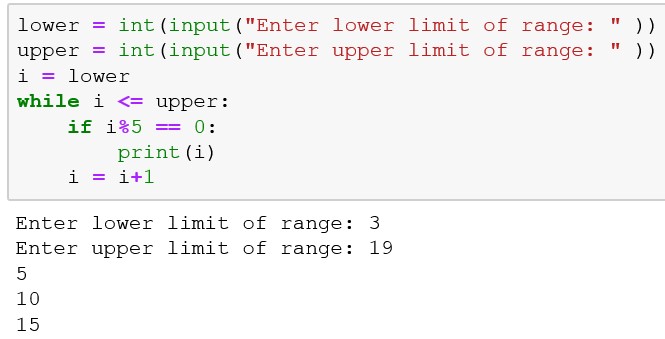


1. Write a program (using while loop) which take a range from the users and display all odd numbers in the range.

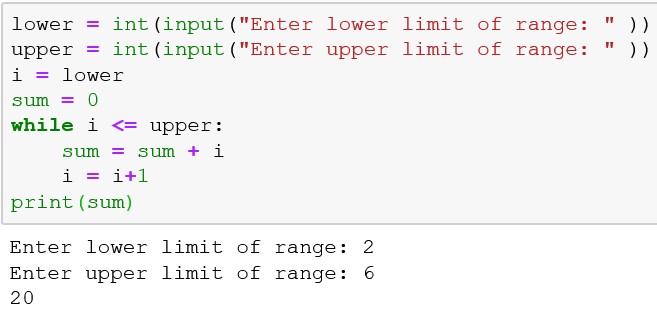
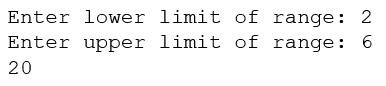


1. Write a program(using while loop) which take a range from the users and display all numbers which are multiple of 5 in the given range.

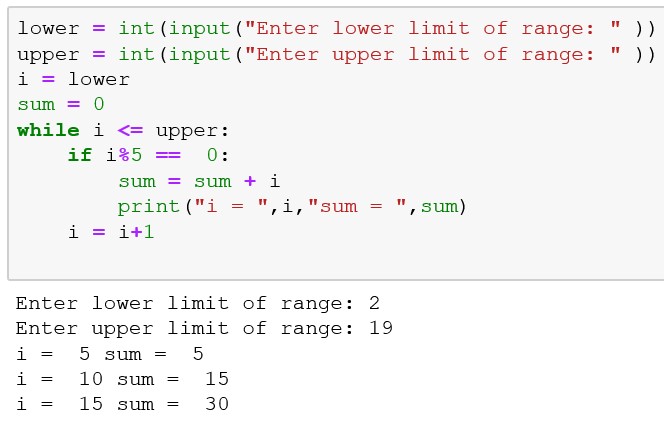




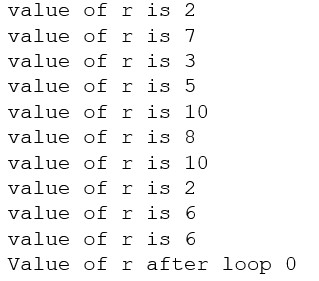
1. Write a program (using while loop) which take a range from the users and display sum of the numbers in the given range?

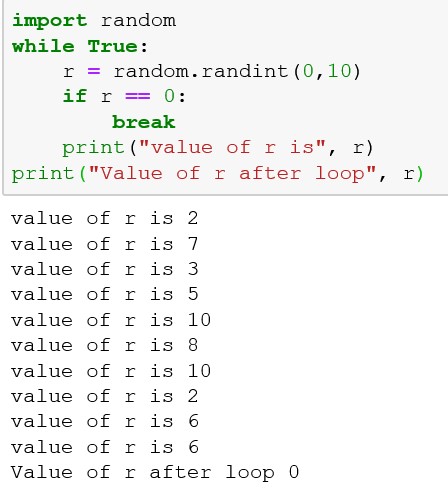


1. Write a program (using while loop) which take a range from the users and display sum of the all numbers which are multiple of 5 in the given range?

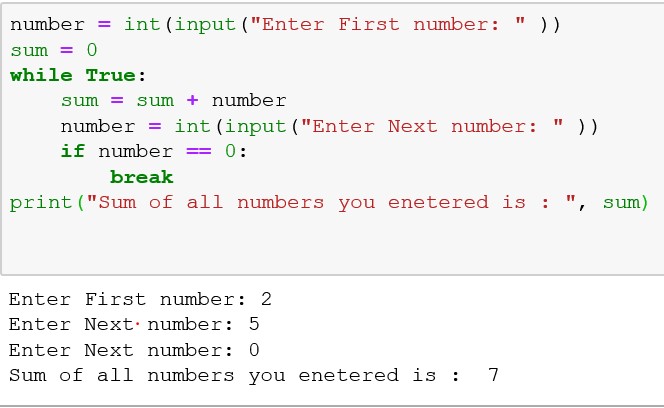
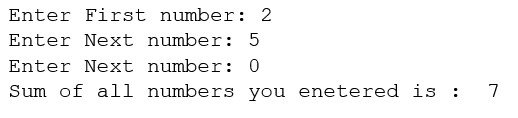


1. Write a program (using while loop) which generate integer random numbers between 0 and 10. Program should display the generated numbers and it should terminate when zero is generated as random number?





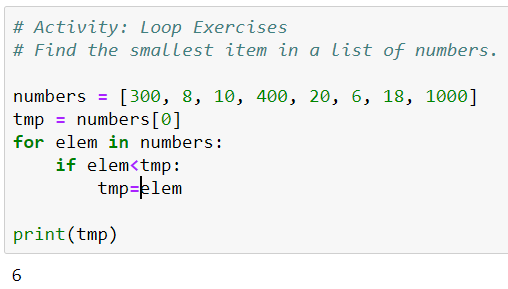
1. Write a program (using while loop) which takes a sequence of numbers from users. Program should stop taking input once user enter zero as input. Program should display the sum of all the numbers entered buy user.



**End of Lab**

# Loop Exercises

# Q 1: Find smallest item in a list of numbers



# Q 2: Guessing Game revisited

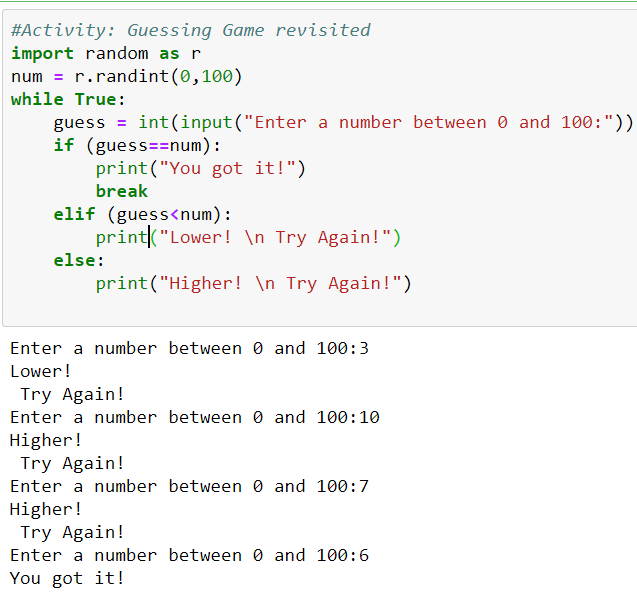
Write code that generates a random number between 0 and 100, and asks the user to guess what it is

Guess the number: 10  
 Higher!

Guess the number: 3  
 Lower!

Guess the number: 2  
 You got it!

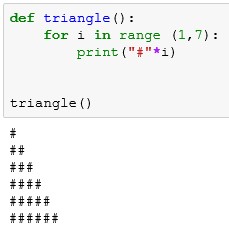
* If the guess is too low, tell the user to guess 'Higher!'
* If the guess is too high, tell the user to guess 'Lower!'
* End the program when the user has found the number



**CSF 2113 Lab 4.1 Functions**

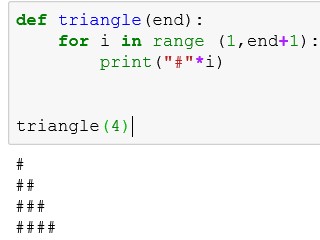
# Using Functions in python

1. Write a Python function which print a triangle of # with lines 1 to 6.



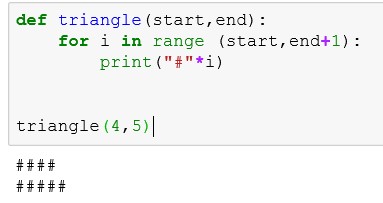
1. What Write a Python function which print a triangle of # with lines from **1 to given number**.

|  |  |
| --- | --- |
|  |  |



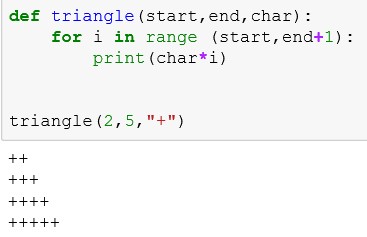
1. Write a Python function which print a triangle of # with lines **from given number** to **given number**.

|  |  |
| --- | --- |
|  |  |



1. Write a Python function which print a triangle of **given character** with lines from **given number** to **given number.**

|  |  |
| --- | --- |
|  |  |



# Using Functions in Python

a. Write the output of the code segment:

|  |  |
| --- | --- |
| Code | Output |
|  | Welcome to Python |
|  | Welcome to Sharjah |
|  | Welcome to Sharjah Welcome to Python |
|  | Welcome to Sharjah  Welcome to Sharjah |
|  | 4  4 |
|  | DubaiDubai |
|  | 12 |
|  | 6 |
|  | 13 |
|  | 12  5  2.6666665  12  Invalid operator |

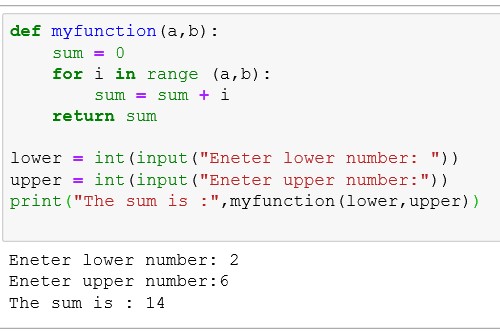
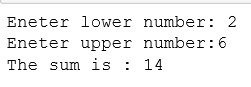
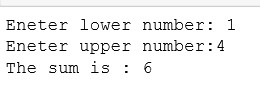
**End of Lab**

**CSF 2113 Lab 4.2 Functions**

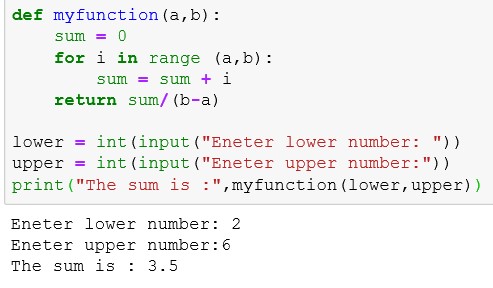
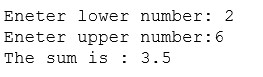
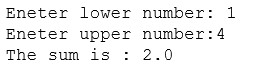
# Using Functions in python

1. Write a Python function which take a two numbers as input and return the sum of the

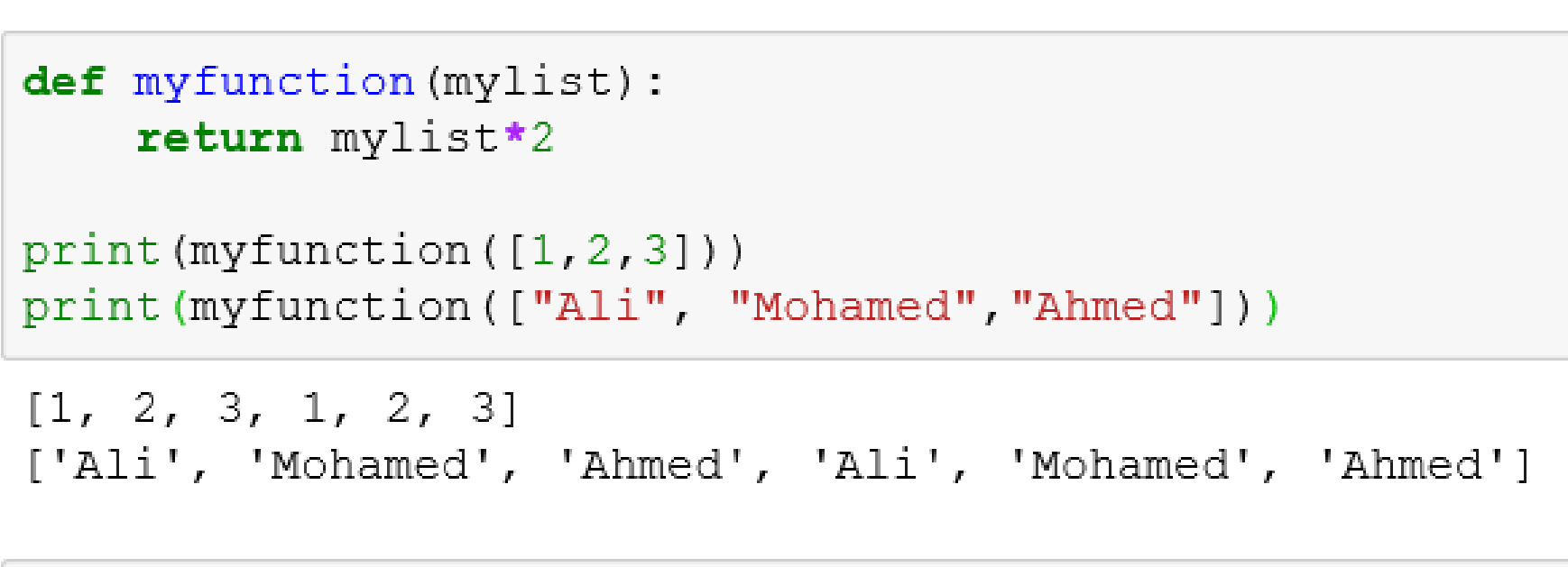
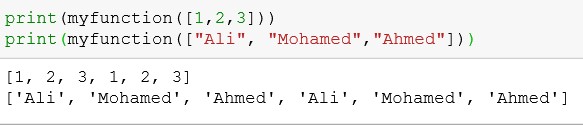
numbers between these two numbers.



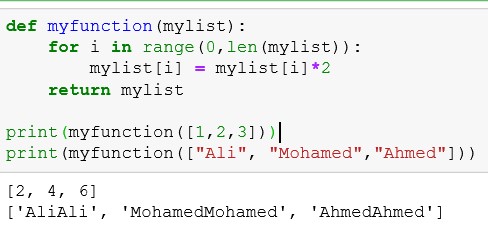
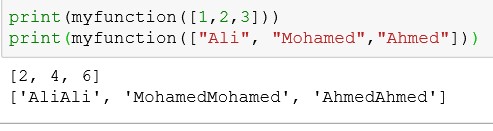
1. What a Python function which take a two numbers as input and return the average of the numbers between these two numbers.



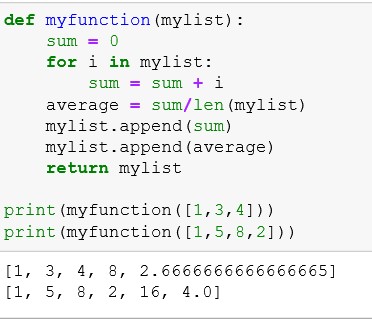
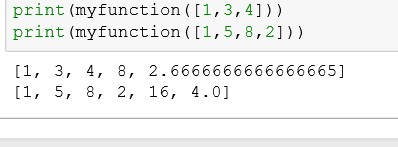
1. Write a Python function which take a list as input and return a list which have listed all element twice.



1. Write a Python function which take a list as input and return a list with element twice of original list.



1. Write a Python function which takes a list of numbers as input and calculate the sum and average of all elements of the list and add both (sum and average) at the end of the list and return the updated list.



# Using Functions in Python

a. Write the output of the code segment:

|  |  |
| --- | --- |
| Code | Output |
|  | 8  10  10 |
|  | -3  11  12 |

|  |  |
| --- | --- |
|  | 3  18  0  10 |
|  | 6  25  0  15 |
|  | 9  12  0  18 |
|  | [1, 3, 5, 1, 3, 5]  [1, 5, 4, 2, 1, 5, 4, 2]  []  [3, 9, 4, 2, 3, 9, 4, 2] |
|  | [2, 6, 10]  [2, 10, 8, 4]  []  [6, 18, 8, 4] |
|  | [1, 3, 5]  [1, 2, 4, 5]  []  [2, 3, 4, 9] |

**End of Lab**

|  |
| --- |
| Lab 4.3: Working with functions |

# Q 1: Caught speeding

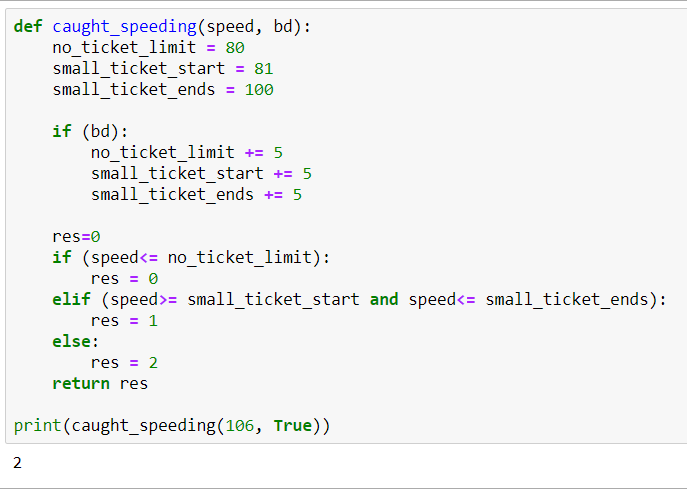
You are driving a little too fast, and a police officer stops you.

Write code to compute the result, encoded as an int value: 0=no ticket, 1=small ticket, 2=big ticket. If speed is 80 or less, the result is 0. If speed is between 81 and 100 inclusive, the result is 1. If speed is 101 or more, the result is 2. Unless it is your birthday -- on that day, your speed can be 5 higher in all cases.

caught\_speeding(80, False) → 0  
caught\_speeding(85, False) → 1  
caught\_speeding(85, True) → 0

Sol:

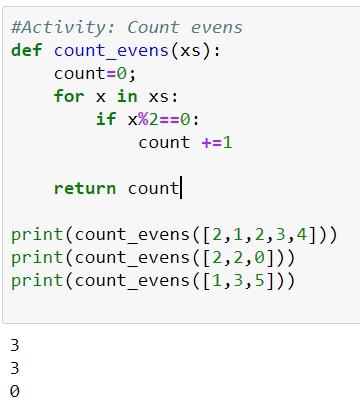
|  |
| --- |
| def caught\_speeding(speed, bd):  no\_ticket\_limit = 80  small\_ticket\_start = 81  small\_ticket\_ends = 100    if (bd):  no\_ticket\_limit += 5  small\_ticket\_start += 5  small\_ticket\_ends += 5    res=0  if (speed<= no\_ticket\_limit):  res = 0  elif (speed>= small\_ticket\_start and speed<= small\_ticket\_ends):  res = 1  else:  res = 2  return res  speed=106  birthday =True  fine = caught\_speeding(106, True)  print(fine) |



# Q 2: Count even elements

Return the number of even ints in the given list. Note: the % "mod" operator computes the remainder, e.g. 5 % 2 is 1.

* count\_evens([2, 1, 2, 3, 4]) → 3  
  count\_evens([2, 2, 0]) → 3  
  count\_evens([1, 3, 5]) → 0



**CSF 2113 Lab 5.1 Functions Advanced Concepts**

1. **Using Recursive Functions in python**
2. Write a recursive function, which takes a number and print its factorial.

|  |
| --- |
|  |

1. Define a recursive function that takes a number n and returns the Fibonacci number of that index from the sequence.

The Fibonacci numbers are the numbers of the following sequence of integer values: 0,1,1,2,3,5,8,13,21,34,55,89, ... . The Fibonacci numbers are defined by:

, with and

|  |
| --- |
|  |

1. Write a recursive function that takes a positive number as input and then, print all the numbers up to 0.

|  |
| --- |
|  |

1. **Using Anonymous functions (Lambda expression)**
2. Write the output of the code segment:

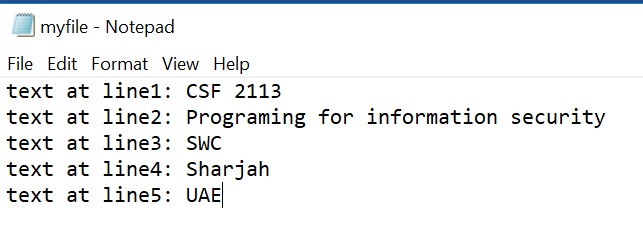
|  |  |
| --- | --- |
| Code | Output |
|  | 4 |
|  | 13 |
|  | 30 |
|  | 4 |
|  | Mr. John |
|  | Ms. Nancy |

**End of Lab**

**CSF 2113 Lab 6: File Handling**

1. Create a file in home directory and name it myfile.txt.

Write the code segment to read and display the information from the file. Read one line at a time and print each line after stripping the whitespaces. File name: myfile.txt



|  |
| --- |
| Code: |
| Output: |

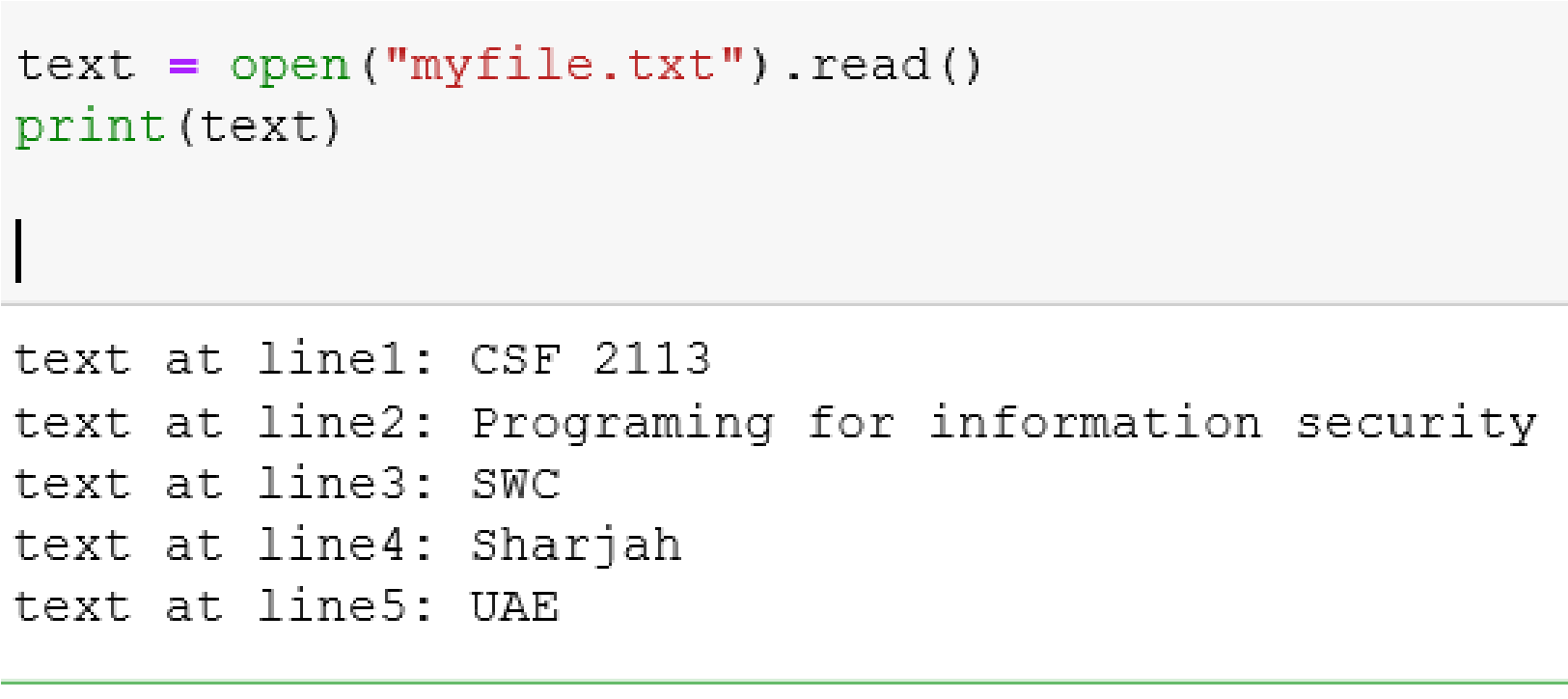
1. Write the code segment to read and display the information from the file. (Same file used in Question a). Read In one go using **readlines()** function.

Code:

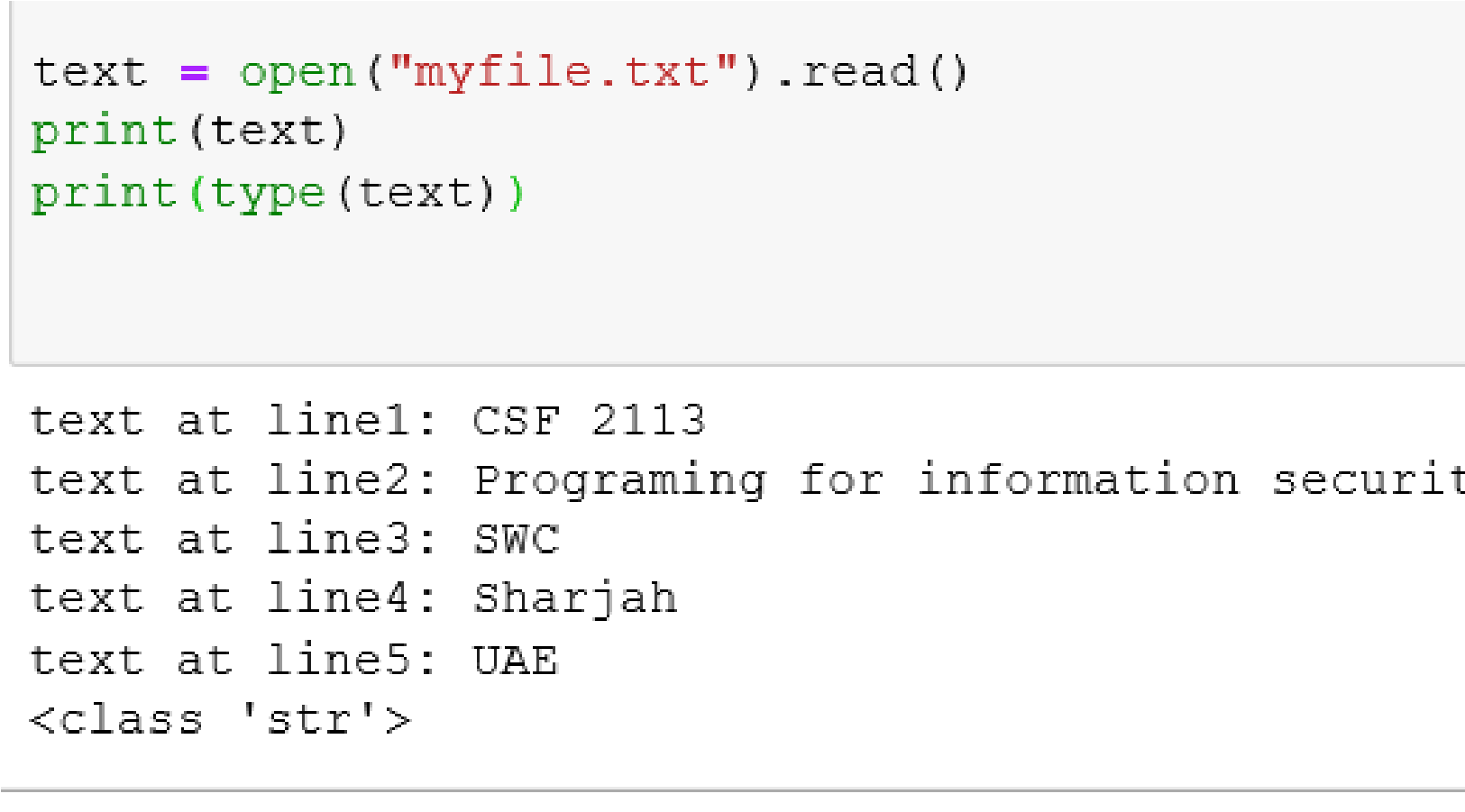
|  |
| --- |
|  |
| Output: |

1. Write the code segment to read and display the information from the file. (Same file used in Question a). Read In one go using **read()** function.

Code:



Output:



1. Create a file in home directory and name it name-city.csv.

Write the code segment to read and display the information from the file. Read one line at a time and store the values in list.

File name: name-city.csv

|  |
| --- |
| Code: |
| Output: |

1. Write the code segment to read and display the information from the name-city.csv file. Read one line at a time and store the values in list of records.

|  |
| --- |
| Code: |
| Output: |

1. Write the code segment to read and display the information from the name-city.csv file. Read using csv library.

|  |
| --- |
| Code: |
| Output: |

1. Write the code segment to write the following information in the file. (

File name: **lab9.txt**

Text to write**: yourname, your id and your college name**.

All three information should be on three lines in file. Your file should look like following:

|  |
| --- |
| Code: |
| Output: |

1. Write a python program to write the list of list in a csv file using csv library.

|  |
| --- |
| Code: |
| Output: |



**CSF 2113 Lab Using requests Library**

1. Write Python code to make a get request to the given web page, and print the response text.

<http://httpbin.org/ip>

|  |
| --- |
| import requests  url = "http://httpbin.org/ip"  r = requests.get(url)  print(r.text) |

|  |
| --- |
|  |

1. Write Python code to make a get request to the given web page, and print the url from response object.

<http://httpbin.org/ip>

|  |
| --- |
| import requests  r = requests.get("http://httpbin.org/ip")  print("URL is : ", r.url) |

|  |
| --- |
|  |

1. Write Python code to make a request to the given web page, and print the status-code from response object.

<http://httpbin.org/ip>

|  |
| --- |
| import requests  url = "http://httpbin.org/ip"  r = requests.get(url)  print("status code : ", r.status\_code) |

|  |
| --- |
|  |

1. Write Python code to make a get request to the given web page, and print the response headers from response object.

<http://httpbin.org/ip>

|  |
| --- |
| import requests  url = "http://httpbin.org/ip"  r = requests.get(url)  print(r.headers) |

|  |
| --- |
|  |

1. Write Python code to make a get request to the given web page, and print the elements of response headers using a loop from response object.

<http://httpbin.org/ip>

|  |
| --- |
| import requests  url = "http://httpbin.org/ip"  r = requests.get(url)  for i in r.headers:  print(i, ":", r.headers[i]) |

|  |
| --- |
|  |

1. Write Python code to make a get request to the given web page, and print response text from response object.

<http://httpbin.org/headers>

|  |
| --- |
| import requests  url = "http://httpbin.org/headers"  r = requests.get(url)  print("text:", r.text) |

|  |
| --- |
|  |

What is the value of the User-Agent?

|  |
| --- |
|  |

1. Write Python code to make a get request to the given web page, set the user agent value to IPhone 11. Print the response text from response object.

<http://httpbin.org/header>

|  |
| --- |
| import requests  newheader = {"user-agent":"IPhone 11"}  url = "http://httpbin.org/headers"  r = requests.get(url, headers=newheader)  print("text:", r.text) |

|  |
| --- |
|  |

1. Write Python code to make a get request to the given web page, set the user agent value to IPhone 11. Print the response text and url from response object.

<http://httpbin.org/get>

|  |
| --- |
| import requests  url = "http://httpbin.org/get"  r = requests.get(url)  print("text:", r.text) |

|  |
| --- |
|  |

1. Write Python code to make a get request to the given web page. Create the dictionary object as follow and pass it as an argument using the '*param*' keyword.



Print the response text and url from response object.

<http://httpbin.org/get>

|  |
| --- |
| import requests  url = "http://httpbin.org/get"  passwd = {"username": "admin", "password":"abc123"}  r = requests.get(url, params=passwd)  print("text:", r.text)  print("url: ", r.url) |

|  |
| --- |
|  |

Observe the output of the Question 8 and 9 and write down the differences between the outputs. Compare and explain the differences.

|  |
| --- |
| The parameters are passes ads args.  "args": {  "password": "abc123",  "username": "admin"  },  The parameters are added at the end of the url as query string  url: <http://httpbin.org/get?username=admin&password=abc123> |

1. Write Python code to make a **post** request to the given web page. Create the dictionary object as follow and pass it as an argument using the '*param*' keyword.



Print the response text and url from response object.

<http://httpbin.org/post>

|  |
| --- |
| import requests  url = "http://httpbin.org/post"  passwd = {"username": "admin", "password":"abc123"}  r = requests.post(url, data=passwd)  print("text:", r.text)  print("url: ", r.url) |

|  |
| --- |
|  |

Observe the url in the output. Can you see the login name and password in the url?

What is the reason?

|  |
| --- |
| No. The url does not contain the login name and password as query string. The data sent to the server with POST is stored in the request body of the HTTP request. |

**End of Lab**

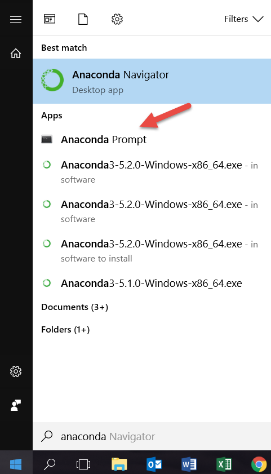
**Lab: Web crawling and scraping with Scrapy**

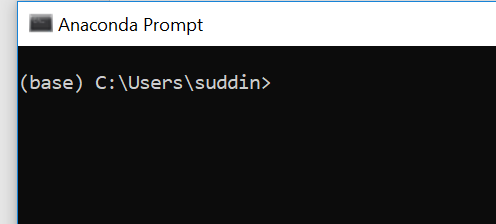
This Lab will take you through the following steps:

1. Installing Scrappy on Windows
2. Creating a new Scrapy project
3. Writing a spider to crawl a site and extract data
4. Exporting the scraped data using the command line
5. Changing spider to recursively follow links
6. **Installing Scrappy on Windows**

If you want to download Scrapy on your windows machine, you can install it using Anaconda as described below:

Click on search  and start typing “**anaconda prompt”:**





Now type the following in ‘Anaconda Prompt’:

> conda install -c anaconda scrapy

Press Y when prompted.

1. **Creating a new Scrapy project**

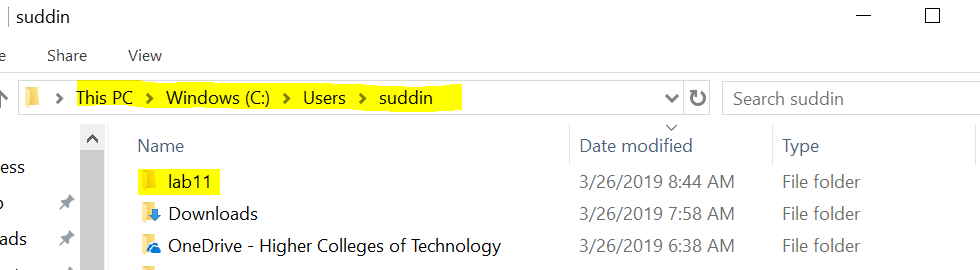
In order to set up a new Scrapy project. Enter a directory where you’d like to store your code and run.

> scrapy startproject lab11

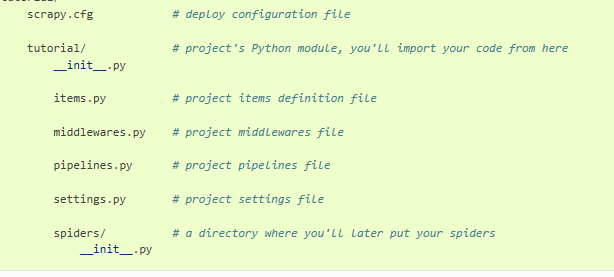
*Where lab11 is the name of the project*

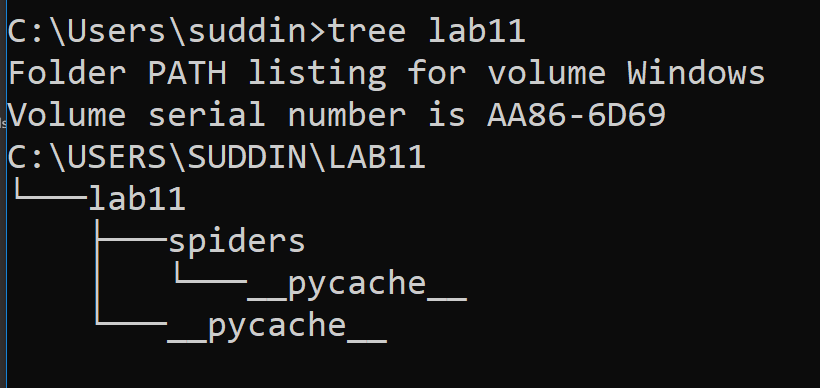


This will create a lab11 directory with some contents:



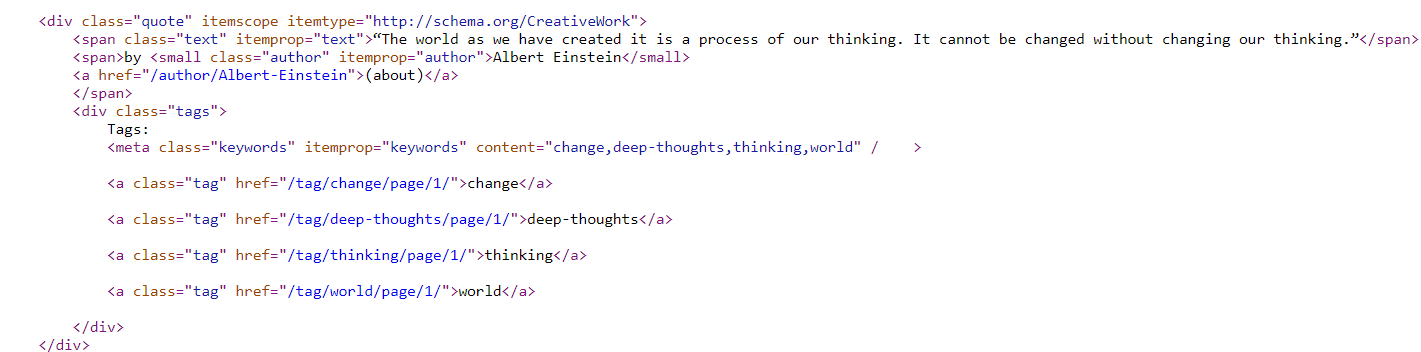
lab11 directory will have following contents:





1. **Writing a spider to crawl a site and extract data**

Here we create a spider by writing the code to extract the quotes from the web page ([http://quotes.toscrape.com](http://quotes.toscrape.com/)). In the source code HTML elements look like following:

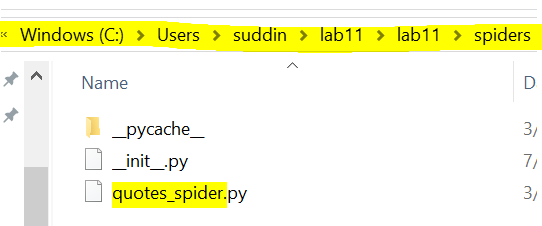


Here is our first code to extract the data from a given URL

|  |
| --- |
| import scrapy  class QuotesSpider(scrapy.Spider):  name = "quotes"  start\_urls = [  'http://quotes.toscrape.com/page/1/',  ]  def parse(self, response):  for quote in response.css('div.quote'):  yield {  'text': quote.css('span.text::text').get(),  'author': quote.css('small.author::text').get(),  'tags': quote.css('div.tags a.tag::text').getall(),  } |
|  |

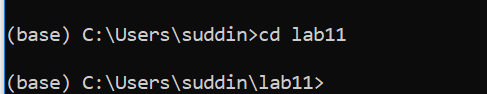
This is code for our first Spider. Save it in a file named **quotes\_spider.py** under the lab11/spiders directory in your project.

***Note:*** *Copy the code in notepad++ or notepad and save it as* ***quotes\_spider.py*** *under the lab11/spiders directory in your project:*



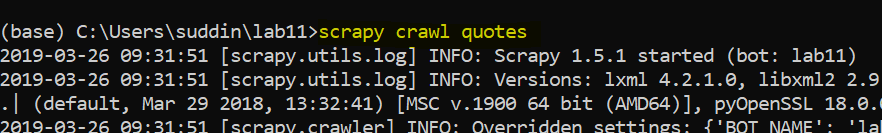
**To Run Spider:**

To run our spider, go to the project’s top level directory and run:

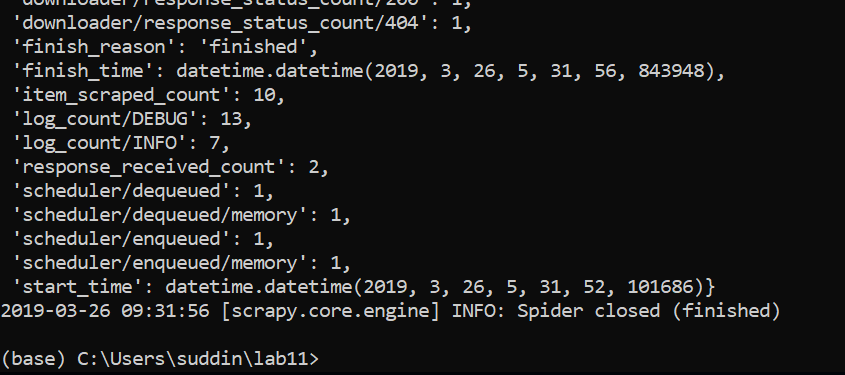


Run the following command:

scrapy crawl quotes



This command runs the spider with name quotes that we’ve just added, that will send some requests for the quotes.toscrape.com domain. You will get an output similar to this:

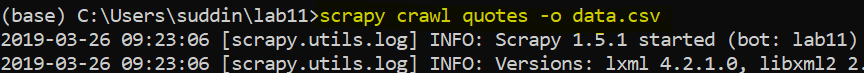


1. **Exporting the scraped data using the command line**

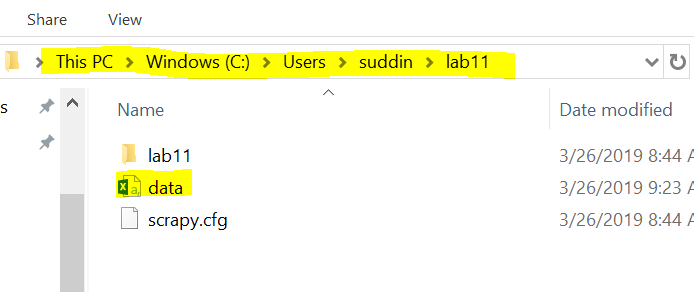
The simplest way to store the scraped data is by using csv file, with the following command:

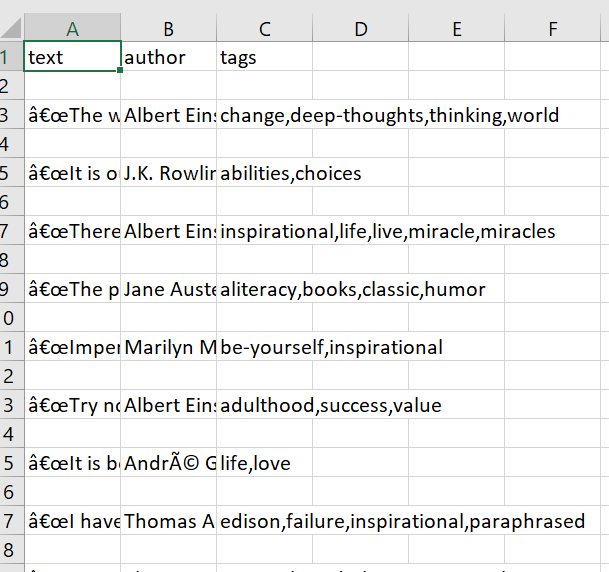
scrapy crawl quotes -o data.csv

Where data.csv is our file to store scrapped data



This will create a file data.csv in your lab11 directory with scrapped data.

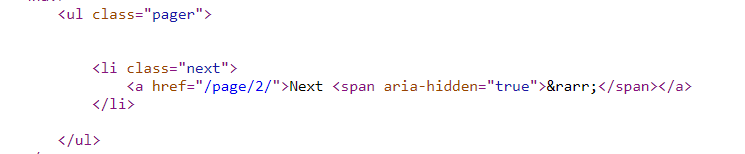




1. **Changing spider to recursively follow links**

Instead of just scraping the stuff from the first two pages from http://quotes.toscrape.com, you want quotes from all the pages in the website.

Now that we have already extracted data from pages, let’s see how to follow links from them. First thing is to extract the link to the page we want to follow. Examining our page, we can see there is a link to the next page with the following markup:

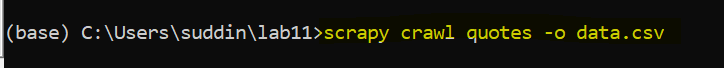


Update the code in quotes.spider.py file to follow the next page link as follow:

|  |
| --- |
| import scrapy  class QuotesSpider(scrapy.Spider):  name = "quotes"  start\_urls = [  'http://quotes.toscrape.com/page/1/',  ]  def parse(self, response):  for quote in response.css('div.quote'):  yield {  'text': quote.css('span.text::text').get(),  'author': quote.css('small.author::text').get(),  'tags': quote.css('div.tags a.tag::text').getall(),  }  next\_page = response.css('li.next a::attr(href)').get()  if next\_page is not None:  next\_page = response.urljoin(next\_page)  yield scrapy.Request(next\_page, callback=self.parse) |
|  |

Now run the following command again and see how many rows of data you have is csv file.

scrapy crawl quotes -o data.csv



1. **Writing a spider to crawl a site and extract data using XPath**

Create a project with name lab12 by repeating step 2 to 5. The only change you have to make is use following code to create quotes.spider.py file instead of the code used in previous example:

Update the code in quotes.spider.py file to follow the next page link as follow:

|  |
| --- |
| import scrapy  class ToScrapeSpiderXPath(scrapy.Spider):  name = 'quotes'  start\_urls = [  'http://quotes.toscrape.com/',  ]  def parse(self, response):  for quote in response.xpath('//div[@class="quote"]'):  yield {  'text': quote.xpath('//span[@class="text"]/text()').extract\_first(),  'author': quote.xpath('//small[@class="author"]/text()').extract\_first(),  'tags': quote.xpath('//div[@class="tags"]/a[@class="tag"]/text()').extract()  }  next\_page\_url = response.xpath('//li[@class="next"]/a/@href').extract\_first()  if next\_page\_url is not None:  yield scrapy.Request(response.urljoin(next\_page\_url)) |
|  |

**CSF 2113 Lab Recourse Discovery**

Note: If you have not installed MAMP. Install and configured it as described in the MAMP installation activity.

1. Download, unzip and copy the folder Lab\_RD from BBlearn and past it in C:\MAMP\htdocs folder:

|  |
| --- |
|  |

1. Copy past the following URL in the browser to confirm the MAMP configuration. <http://localhost:8123/Lab_RD/page1.html>

Note: If you have used a different port number, above URL will not work. Replace the 8123 port number with the number you used.

|  |
| --- |
|  |

1. Write python code in the Juypter notebook and send a get request to the webpage and display the contents and status\_code.

<http://localhost:8123/Lab_RD/page1.html>

|  |
| --- |
| import requests  url = "http://localhost:8123/Lab\_RD/page1.html"  r = requests.get(url)  print(url, r.status\_code)  print(r.text) |

|  |
| --- |
|  |

1. Write python code in the Juypter notebook and send a get request to the webpage and display only the status\_code.

<http://localhost:8123/Lab_RD/page3.html>

|  |
| --- |
| import requests  url = "http://localhost:8123/Lab\_RD/page3.html"  r = requests.get(url)  print(url, r.status\_code) |

|  |
| --- |
|  |

Answer the following questions:

* What is the difference in the status\_code of pag1 and pag3?

|  |
| --- |
|  |

* What does it mean to receive a status\_code of 200?

|  |
| --- |
|  |

* What does it mean to receive a status\_code of 404?

|  |
| --- |
|  |

* Whay are the status\_code different?

|  |
| --- |
|  |

1. Write python code in the Juypter notebook that store the following urls in a list send a get request to every webpage and display URL with the status\_code.

|  |
| --- |
| http://localhost:8123/Lab\_RD/page1.html  http://localhost:8123/Lab\_RD/page2.html  http://localhost:8123/Lab\_RD/page3.html  http://localhost:8123/Lab\_RD/page4.html  http://localhost:8123/Lab\_RD/page5.html  http://localhost:8123/Lab\_RD/page6.html  http://localhost:8123/Lab\_RD/page7.html  http://localhost:8123/Lab\_RD/page8.html  http://localhost:8123/Lab\_RD/page9.html  http://localhost:8123/Lab\_RD/page10.html |

|  |
| --- |
| import requests  urls= ["http://localhost:8123/Lab\_RD/page1.html",  "http://localhost:8123/Lab\_RD/page2.html",  "http://localhost:8123/Lab\_RD/page3.html",  "http://localhost:8123/Lab\_RD/page4.html",  "http://localhost:8123/Lab\_RD/page5.html",  "http://localhost:8123/Lab\_RD/page6.html",  "http://localhost:8123/Lab\_RD/page7.html",  "http://localhost:8123/Lab\_RD/page8.html",  "http://localhost:8123/Lab\_RD/page9.html",  "http://localhost:8123/Lab\_RD/page10.html"]  for url in urls:  r = requests.get(url)  print(url, r.status\_code) |

|  |
| --- |
|  |

Answer the following questions:

* How many pages have status\_code of 200?

|  |
| --- |
| 8 |

* How many pages have status\_code of 404?

|  |
| --- |
| 2 |

* How many pages are present in the folder Lab\_RD?

|  |
| --- |
| 8 |

* List the URLs of the pages present in Lab\_RD

|  |
| --- |
| * http://localhost:8123/Lab\_RD/page1.html * http://localhost:8123/Lab\_RD/page2.html * http://localhost:8123/Lab\_RD/page4.html * http://localhost:8123/Lab\_RD/page5.html * http://localhost:8123/Lab\_RD/page6.html * http://localhost:8123/Lab\_RD/page7.html * http://localhost:8123/Lab\_RD/page9.html * http://localhost:8123/Lab\_RD/page10.html |

1. Copy and paste the URLS given below in a text file and save it in your Juypter home directory.

Write python code in the Juypter notebook that read these URLs from the file and send a get request to every webpage to display URL with the status\_code.

|  |
| --- |
| http://localhost:8123/Lab\_RD/page1.html  http://localhost:8123/Lab\_RD/page2.html  http://localhost:8123/Lab\_RD/page3.html  http://localhost:8123/Lab\_RD/page4.html  http://localhost:8123/Lab\_RD/page5.html  http://localhost:8123/Lab\_RD/page6.html  http://localhost:8123/Lab\_RD/page7.html  http://localhost:8123/Lab\_RD/page8.html  http://localhost:8123/Lab\_RD/page9.html  http://localhost:8123/Lab\_RD/page10.html |

|  |
| --- |
|  |

|  |
| --- |
| import requests  #reading input file  urls = open("labRD.txt").readlines()  #cleaning data  for url in range(0,len(urls)):  urls[url] = urls[url].rstrip()    for url in urls:  r = requests.get(url)  print(url, r.status\_code) |

|  |
| --- |
|  |

**End of Lab**

**CSF 2113 Lab Pen testing**

1. Write python code in the Juypter notebook and send a get request with basic authentication to the webpage and display the status\_code.

url = <https://httpbin.org/basic-auth/admin/admin123>

login = "mabdalla"

password = "Srtb@24e"

|  |
| --- |
| import requests  url = "https://httpbin.org/basic-auth/admin/admin123"  login = "mabdalla"  password = "Srtb@24e"  cred = (login,password)  #print(cred)  r = requests.get(url, auth= cred)  print(cred,"\t",r.status\_code) |

|  |
| --- |
|  |

1. Write python code in the Juypter notebook and send a get request with basic authentication to the webpage and display the status\_code.

url = <https://httpbin.org/basic-auth/admin/admin123>

login = "admin"

password = "admin321"

|  |
| --- |
| import requests  url = "https://httpbin.org/basic-auth/admin/admin123"  login = "admin"  password = "admin321"  cred = (login,password)  #print(cred)  r = requests.get(url, auth= cred)  print(cred,"\t",r.status\_code) |

|  |
| --- |
|  |

1. Write python code in the Juypter notebook and send a get request with basic authentication to the webpage and display the status\_code.

url = <https://httpbin.org/basic-auth/admin/admin123>

login = "admin"

password = "admin123"

|  |
| --- |
| import requests  url = "https://httpbin.org/basic-auth/admin/admin123"  login = "admin"  password = "admin123"  cred = (login,password)  #print(cred)  r = requests.get(url, auth= cred)  print(cred,"\t",r.status\_code) |

|  |
| --- |
|  |

Answer the following questions:

* What is the difference in the status\_code for Q1, Q2 and Q3?

|  |
| --- |
|  |

* What does it mean to receive a status\_code of 200?

|  |
| --- |
|  |

* What does it mean to receive a status\_code of 401?

|  |
| --- |
|  |

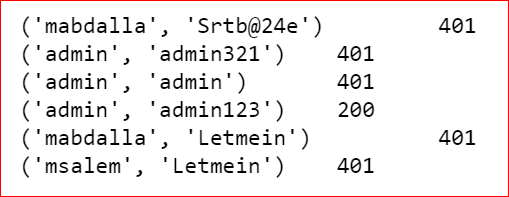
* What is the password of user with login “admin”?

|  |
| --- |
|  |

1. Write python code in the Juypter notebook that store the following login and password combination in list of tuples and send a get request with basic authentication to the following URL and display the status\_code.

url = <https://httpbin.org/basic-auth/admin/admin123>

Expected output:



|  |
| --- |
| mabdalla,Srtb@24e  admin,admin321  admin,admin  admin,admin123  mabdalla,Letmein  msalem","Letmein |

|  |
| --- |
| import requests  url = "https://httpbin.org/basic-auth/admin/admin123"  data = [("mabdalla","Srtb@24e"),  ("admin","admin321"),  ("admin","admin"),  ("admin","admin123"),  ("mabdalla","Letmein"),  ("msalem","Letmein")  ]  for cred in data:  r = requests.get(url, auth=cred)  print(cred,"\t",r.status\_code) |

|  |
| --- |
|  |

Answer the following questions:

* How many combination of login and password have status-code of 200?

|  |
| --- |
| 1 |

* How many combination of login and password have status-code of 401?

|  |
| --- |
| 5 |

* How many combination of login and password are successful?

|  |
| --- |
| 1 |

1. Download the test file **pentestdata.txt** containg the combination of login and password information. Read the data from file and send a get request with basic authentication to the following URL and display the status\_code.

url = <https://httpbin.org/basic-auth/admin/admin123>

|  |
| --- |
| import requests  #reading input file  data = open("pentestdata.txt").readlines()  #cleaning data  for x in range(0,len(data)):  data[x] = data[x].rstrip()  #printing data for varification  #for i in data:  # print(i)  for item in data:  log\_pas = item.split(",")  cred = (log\_pas[0],log\_pas[1])  r = requests.get(url, auth=cred)  print(cred,"\t",r.status\_code) |

|  |
| --- |
|  |

1. Copy and Past the url used in above question and try to login using the successful login and password information. Take the screenshot of the successful login and paste it in the answer section.

url = <https://httpbin.org/basic-auth/admin/admin123>

|  |
| --- |
|  |

**End of Lab**