A picture containing dark, night sky

Description automatically generated

Assignment 01 - Programming for Big Data

Hasan Mahamud Rana

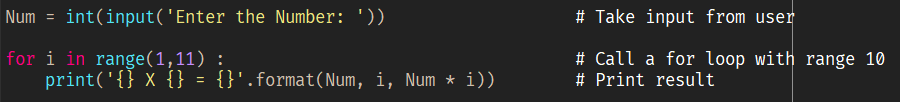
8732852

PROG8420 – Programming for Big Data

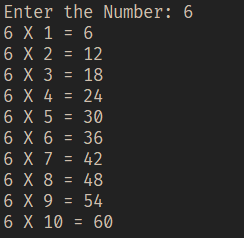
Radha Chawla

Due Date: Jun 25, 2021 11:59 PM

**Question 1: Write a python program to display the multiplication table with taking input from the user. [5 marks]**



**Output:**



(Radha Chawla, 2021, *Week 4\_Object type part 2 slide 3*)

**Question 2: What are functions in Python and what is the purpose of them? [2 marks]**

A function is a group of related organized statements that are writing to perform a particular specific task. A function is a block of code which only runs when it is called. We can pass data, known as parameters, into a function. There are inbuilt functions i.e print() and also user-defined functions.

Function purpose:

* We can use it repeatedly.
* Maximizing code reuse.
* The function also provides procedural decomposition. We can decompose multiple
* extensive procedures and divide them into various groups or considerable chunks.
* The function makes a program much easier to understand.

Creating a Function, In Python a user-defined function is defined using the def keyword:

Syntax

def <name>(arg1, arg2,... argN):

<statements>

return(<value>)

Defining a Function

def my\_function():

print('Hello from a function')

To call a function, use the function name followed by parenthesis:

my\_function()

Arguments, Information can be passed into functions as arguments. Arguments are specified after the function name, inside the parentheses. You can add as many arguments as you want, just separate them with a comma. The following example has a function with one argument (fname). When the function is called, we pass along a first name, which is used inside the function to print the full name:

Example

def my\_function\_with\_arguments(x, y):

z = x \*\* y

return(z)

x = 2

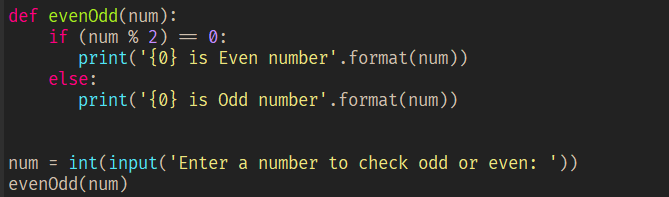
y = 3

result = my\_function\_with\_arguments(x,y)

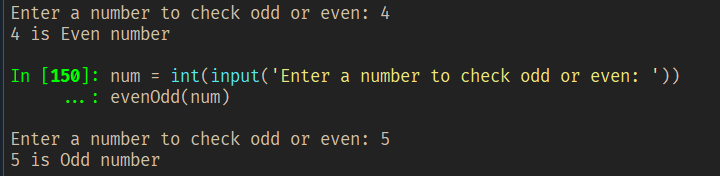
result

(Radha Chawla, 2021 PROG8420\_Week 6\_W6 - 02 – Func)

**Question 3: Write a program to check odd or even number. [5marks]**



Output:



**Question 4: What are lists and how are they different from tuples with example? [3 marks]**

List - To create a list in Python, enclose items in square brackets separated by commas and assign it a variable. Example:

weekdays = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday']

print(weekdays)

A tuple is an ordered sequence of items.

continents = ('Asia', 'Africa', 'Americas', 'Europe', 'Australia')

print(continents)

**Difference**

The only difference between it and a list is that a tuple is immutable. This means that once it has been created, it cannot be modified.

**Update List**

weekdays[0] = 'Updated'

print(weekdays) #List value updated

**Update Tuple**

continents[0] = 'Updated' # it will give (TypeError: 'tuple' object does not support item assignment)

print(continents)

However, It is possible add another Tuple

continents2 = ('Antarctica',)

all\_continents = continents + continents2

print (all\_continents)

Also, it is possible delete a Tuple

print (continents2)

del continents2

print (continents2) # It is no longer available (NameError: name 'continents2' is not defined)

(Radha Chawla, 2021 - Week 4\_CollectionObjectType slide 21), (Radha Chawla, 2021, Week 5\_W5 - 1 - CollectionObjectType slide 18)

**Question 5: What is \_\_name\_\_ in python and its purpose? [2 Marks]**

The \_\_name\_\_ variable (two underscores before and after) is a special Python variable. In Python, we can import that script as a module in another script.

In Python is no main() function; when the command runs a python program, the interpreter executes the code at level 0 indentation. However, before doing that, the interpreter defines a few special variables. \_\_name\_\_ is one of the particular built-in variables.

\_\_name\_\_ evaluates the name of the current module. \_\_name\_\_ can be used to check whether the current script is being run on its own or being imported. If the file is executed as the main program, the interpreter sets the \_\_name\_\_ variable value to “\_\_main\_\_”. Else if this file is being imported from another module, \_\_name\_\_ will be set to the module’s name.

Example: Write the following code in a file name mod1.py

def M1\_F1(x):

print('From mod1 file we see namespace...', \_\_name\_\_)

print('The passed value is', x)

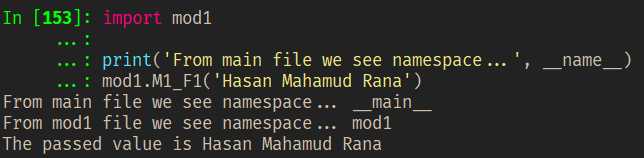
Write the following code in another file name say main.py

import mod1

print('From main file we see namespace...', \_\_name\_\_)

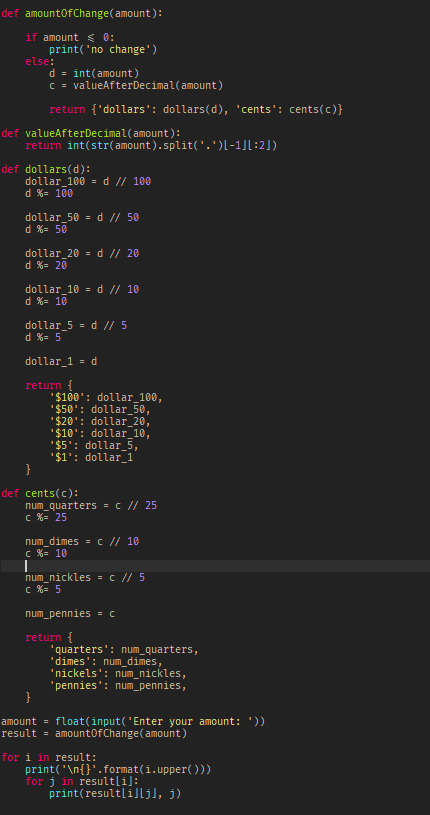
mod1.M1\_F1('Hasan Mahamud Rana')

Output:

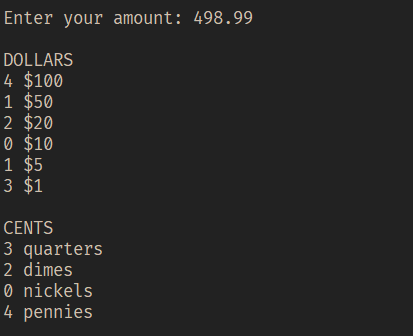


(Radha Chawla. 2021, Week 3\_Week 3 - 1 slide 3), (Python 3.9.5 Documentation, 2014)

**Question 6: Write a program that prompts user to enter a number and then prints the number of coins of each type required to make the given amount of change. [5 marks]**

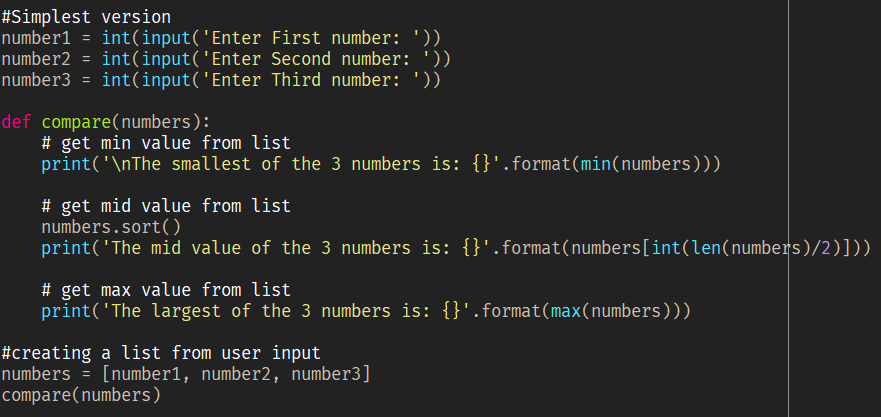


Output:

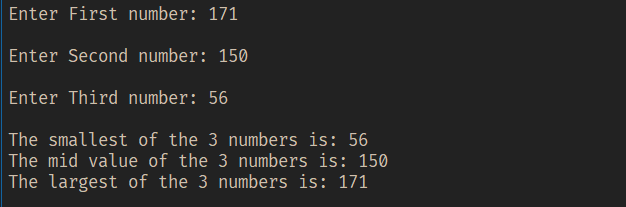


(Radha Chawla, 2021, W6 - 01 - Loops slide 7)

**Question 7: Write a program to get three inputs from user and compare these three numbers. [4 marks]**



Output:

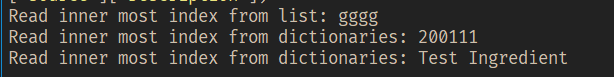


(Radha Chawla, 2021, Week 5 - 1 - CollectionObjectType slide 5)

**Question 8: Create an example of nested list, dictionaries and extract the value from inner most index. [2 marks]**



Output:



(Radha Chawla, 2021, Week 5 - 1 - CollectionObjectType slide 3, 11)

**Question 9: Is python an interpreted language. If yes, why if not explain. [2 marks]**

Python is considered an interpreted language because Python programs are executed by an interpreter. Interpreter - processes the program a little at a time, line by line.

SOURCE CODE -> INTERPRETER -> OUTPUT

(Radha Chawla, 2021, Week 2 - 1, slide 3,4)

**References**

Radha Chawla. (2021). *PROG8420\_Week 4\_Object type part 2 slide 3* [PowerPoint slides]. eConestoga.

Radha Chawla. (2021). *PROG8420\_Week 6\_W6 - 02 - Func* [PowerPoint slides]. eConestoga.

Radha Chawla. (2021). *PROG8420\_Week 4\_CollectionObjectType slide 21* [PowerPoint slides]. eConestoga.

Radha Chawla. (2021). *PROG8420\_Week 5\_W5 - 1 - CollectionObjectType slide 18* [PowerPoint slides]. eConestoga.

Radha Chawla. (2021*). PROG8420\_Week 3\_Week 3 - 1 slide 3* [PowerPoint slides]. eConestoga.

Python 3.9.5 Documentation, (August, 2012) *The Import system* https://docs.python.org/3/reference/import.html?highlight=\_\_name#\_\_name\_\_

Radha Chawla. (2021). *PROG8420\_Week 5 - 1 - CollectionObjectType slide 5* [PowerPoint slides]. eConestoga.

Radha Chawla. (2021). *PROG8420\_Week 5 - 1 - CollectionObjectType slide 3, 11* [PowerPoint slides]. eConestoga.

Radha Chawla. (2021*). PROG8420\_Week 2 - 1, slide 3,4* [PowerPoint slides]. eConestoga.

Radha Chawla. (2021). *PROG8420\_W6 - 01 - Loops slide 7* [PowerPoint slides]. eConestoga.