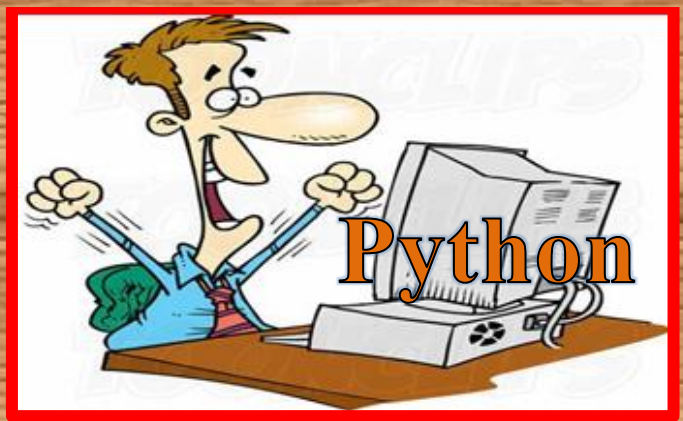
Wright College + Chapter 4

Loops and Iteration: Using 'for' and 'while' Statements, Math functions and Random Numbers

CIS 103 Python Programming Language + Introduction to Computer Programming



'Hands-On' Mastering Computer Logic, Design and Programming Using Python Language

Written By:

OGAR HAJI

Master's Degree in Computer Science

DePaul University ✚ Chicago, Illinois + USA

Date Published: February 5, 2023

Wright College +

Chapter 4 + Part A

Loops and Iteration: Using 'for' and 'while' Statements, Math functions and Random Numbers

CIS 103 Python Programming Language + Introduction to Computer Programming

'Hands-On' Mastering Computer Logic, Design and Programming Using Python Language

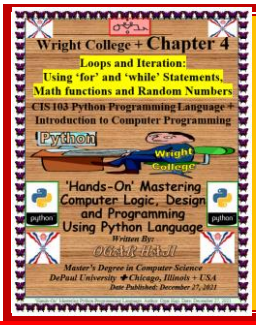
Written By:

OGAR HAJI

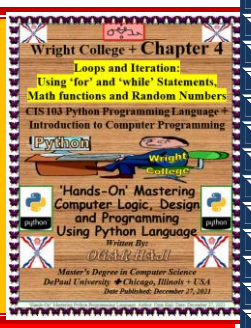
Master's Degree in Computer Science

DePaul University ✚ Chicago, Illinois + USA

Date Published: February 5, 2023



Instructor Hands-On Mastering Python Programming Language Chapters to Cover:



1) Chapter 01 + Introductions to Computers and Coding in Python Programming Language

2) Chapter 02 + Declaring Variables in Python and Turtle

3) Chapter 03 + Conditional Statements Using If Statement

4) Chapter 04 + Loops using For and While Statements

5) Chapter 05 + Introduction to Functions

6) Chapter 06 + Advanced Functions

7) Chapter 07 + Files and Exceptions

8) Chapter 08 + Python Lists and Tuples

9) Chapter 09 + Python Lists and 2D Lists

10) Chapter 10 + Python Strings and String Manipulation

11) Chapter 11 + Python Dictionaries and Sets

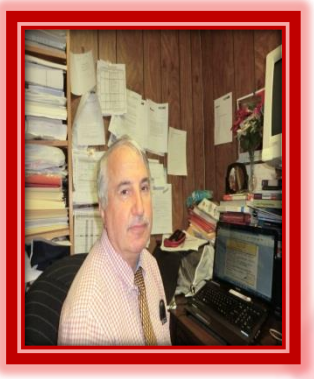
12) Chapter 12 + Object-Oriented Programming (OOP) in Python

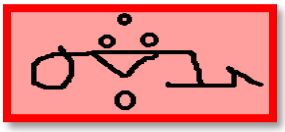
13) Chapter 13 + Recursion Functions and Fibonacci Numbers

14) Chapter 14 + Graphical User Interface (GUI)

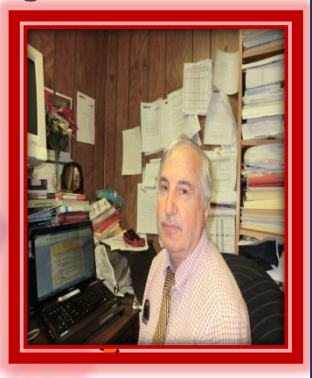
15) Chapter 15 + Data Base and Python

Page B



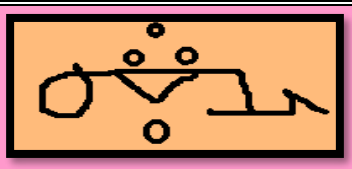


Computer Text Books Published by the Author: Ogar Haji

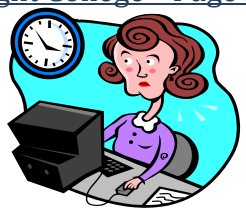


The Following is a List of Computer Text Books Published by the Author: Ogar Haji. He has an MS Degree in Computer Science from DePaul University, Chicago, Illinois + USA. Mr. Ogar Haji has over 30 Years of teaching experience at: The College of Office Technology, Oakton College, Washington College, Truman College, Wright College, Triton College, ITT Technical Institute, Phoenix University and East+West University in Chicago, Illinois + USA.

1) **'Hands-On' Mastering Microsoft Windows 10 and 11**
2) **'Hands-On' Mastering Microsoft Excel 2016 and 2019**
3) **'Hands-On' Mastering Microsoft Word 2016 and 2019**
4) **'Hands-On' Mastering Microsoft Access 2016 and 2019**
5) **'Hands-On' Mastering Microsoft PowerPoint 2016 & 2019**
6) **'Hands-On' Mastering Microsoft Publisher 2010**
7) **'Hands-On' Mastering MS Visual Basic .Net Language**
8) **'Hands-On' Mastering C# Programming Language**
9) **'Hands-On' Mastering Html5 and CSS3 Web Page Design**
10) **'Hands-On' Mastering JavaScript Programming Language**
11) **'Hands-On' Mastering Ruby Programming Language**
12) **'Hands-On' Mastering Java Programming Language**
13) **'Hands-On' Mastering QBasic Programming Language**
14) **'Hands-On' Mastering DOS (Disk Operating System)**
15) **'Hands-On' Mastering Python Programming Language**



Ogar Haji (Computer Instructor)
Harry S. Truman College
1145 West Wilson Avenue
Chicago, Illinois 60640
USA



Computer Labs Rules

1- **No Drinks, Food, Headphones allowed in Computer labs. And Please Turn Off the Cell Phones.**

2- **When Lecturing is in progress, you are Not allowed to work on the computer. Please Pay Attention and Take Notes.**

3- **Attendance and Punctuality are very important. If you are absent, it is your responsibility to make up for the missing work and assignments. Attendance will be taken daily.**

4- **Students should have a USB Flash Drive and Save Projects to it.**

5- **Practice makes perfect. Please keep practicing the new features or steps repeatedly until the instructor tells you to stop.**

6- **You have to Concentrate on what you are doing. Talking is Not Allowed in the computer Lab.**

7- **Please Study the Lessons in your Python Handout and Text Book Daily and review your notes before class. There will be a Quiz Once a Week.**

8- **Please Check Mark ☑ the Lessons in the Handout that you have completed.**

9- **You must do All Python works, Assignments and Tests located at the End of each Chapter on Time.**

CIS 103 Python Programming

Instructor: Ogar Haji

Chapter 4 + Part A

Loops and Iteration:
Using 'for' and 'while' Statements,
Math functions and Random Numbers
Print your Full Name 'Ogar Haji' 10 Times Project,
Sum a Range of Numbers Project,
Print Multiplication Table of 10 Rows and 10 Columns Project

You will learn the following in Chapter 4 Part A:

- Making Decision using Single if statement
- Using if --else statement to check if a condition is true or false
- Calculate the Average of 3 Tests and using if statement to check if student is 'Passing' or 'Failing'
- Using if – else statement to find Final Grade for the Average
- Calculate the Average of 3 Tests and Using if statement
- Using if statement to check if a condition is true or false
- Calculate Overtime of GrossPay exercise
- Using the Conditional And Operator &&
- Using the Conditional Or Operator ||
- Using the Switch …. Case statement
- Using the Ternary Conditional Operator (? :)
- Using the Not ! operator to Negate the Result (Opposite)
- Do Chapter 4 Instructor Handout Homework #5
- Do Lab Assignment 4 + Hotel Room Charges Calculator Project

Input/Output

Flowchart Symbols

Processing

+++ Review +++

Lesson 150 Review + What are the Flowchart Symbols used in Python Language?

You should always **Draw a Flowchart** when you Design, Code and Solve a problem in Python language.

Before you Code a program in Python Language, you have to **Draw a Flowchart** to solve the problem of the program you want to code.

The following symbols are used with Python Programming Language:

Symbol	Symbol Name	Usage
Start or End	**Oval (Beginning and Terminal) symbol**	Use **Oval (Beginning and Terminal) Symbol** at the Beginning of the Flowchart and at the End of the Flowchart. Use with Start and End statements.
Print and Read	**Parallelogram (Input/Output)** Symbol	Use **Parallelogram (Input /Output)** or **I/O** symbol to Input Data, Read Input or Print Output
gross_pay = hours * rate	**Rectangle** Symbol	Use the **Rectangle** Symbol for Calculating, Assigning Values
	**FlowLine** Symbol	Use **FlowLine** Symbols to show the Flow or Sequence of the flowchart.
if hours>40:	**Diamond (Decision)** Symbol	Use **Diamond (Decision)** Symbol with the If or Select statements when deciding if Hours is > 40. The Result will be either True or False.
A	**Connector** Symbol	Use **Connector** Symbol to Connect the Flowchart rather than draw a long Arrow. Use A
calculateGrossPay()	**Function or Method (Predefined Process)** Symbol.	Use **Function** or **Method (Predefined Process)** Symbol to call another Function or Method that contains coding statements.

Calculate Gross Pay of Employees Project

+++ (Do Lab Exercise) +++

+++ Review +++

Lesson 151 Review + How to Calculate Gross Pay of Employees Project?

Problem or Project: Design and Code in Python Language the project to Calculate Gross Pay of Employees in a company.

Do the following 12 Must Steps to Design, Code and Solve a project using Python Language.

Do Steps 1 thru 7 in your Note Book or on Paper.

Step 1) Purpose of the Program: State what Program will do: (5 Points)

a) This Program will calculate Gross Pay of Employees.
b) It will ask the User to Enter Employee's Full Name:
c) It will ask the User to Enter Number of Hours Worked and
d) It will ask the User to Enter Hourly Rate.
e) The program will then calculate Gross Pay.

Gross Pay = Hours * Rate

f) Display the Gross Pay

Step 2) Input: You should know how the Input looks like: (5 Points)

Enter Employee's Full Name: Ogar Haji
Enter Hours Worked: 40
Enter Hourly Rate: 10

Step 3) Processing and Calculation: The program will process each record and Calculate Gross Pay: (5 Points)

Gross Pay = Hours * Rate

Step 4) Output: You should know how output should look like: (5 Points)

********** Calculate Gross Pay Project **********

The Employee Ogar Haji Gross Pay is = $ 400

********** End of Program **********

Step 5) Flowchart: Draw a Flowchart for Gross Pay program.(5 Points)

Start

1) Declare the Variables to be used in the project

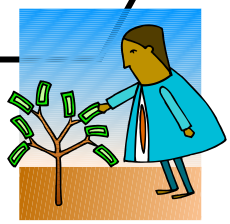
2) print to output ('Title/Heading')

3) Prompt user 'Enter Employee's First Name: '
Prompt user ('Enter Employee's First Name: ')

4) Read the first_name, last name and store it in variables

5) Prompt the user ('Enter Hours Worked: ')

6) Read hours, store it in hours variable, convert it to int()



7) Prompt the user ('Enter Hourly Rate: ')

8) Read Rate, store it in a rate variable, convert to float()

9) Calculate Gross Pay = hours * rate

10) print to output ('Your Gross Pay is = $', grossPay)

11) print to output ('******* End of Program *******')

End

Step 6) PseudoCode: print a PseudoCode for the Program.(5 Points)

1) Declare the variables to be used in the program
2) print to Console 'The Title or Heading of the program '
3) Prompt the user 'Enter Employee's Full Name: '
4) Read from Console full_name and store the String in a variable
5) Prompt the user 'Enter Hours Worked: '
6) Read from console hours worked and store in a String variable
7) Convert String hours variable to integer value
8) Prompt the user 'Enter Hourly Rate: '
9) Read from console the rate and store in a String variable
10) Convert String rate variable to float value
11) Calculate Gross Pay = Hours * Rate
12) print to Console 'The Employee Gross Pay is = $', gross_pay
13) print to Console '***** End of Program *******'**

Step 7) Code the Program in Python by referencing the Flowchart or Pseudocode you designed above and Save it as CalculateGrossPay to USB.

a) Type the following Python code in IDLE IDE: c File, New File

Code for the first part of 'CalculateGrossPay' project follows:

```
# 1) Add Comments about the Calculate Gross Pay Project
'''
 ************************************************************
   Purpose of the Project:
   a) This Interactive Project will prompt the user to enter his/her (First Name,
   Last Name, and Hours Worked and Rate) then it will read the text entered
    and  store it in its variables. Then it Calculates the Gross Pay.
   b) Project Name: CalculateGrossPay
   c) Date: Saturday, December 28, 2016
   d) Programmer: Instructor – Ogar Haji
 ************************************************************
'''
# 2) Prompt the user to Enter his/her First Name and store it in a variable
first_name = input ('Enter your First Name: ')
# 3) print or echo First Name to the output
print ('Your First Name is: ', first_name)
# 4) Prompt the user to Enter his/her Last Name and store it in a variable
last_name = input ('Enter your Last Name: ')
# 5) print or echo Last Name to the output
```

print ('Your Last Name is: ', last_name)

6) Prompt the user to Enter Hours Worked and Store it in a variable
hours = input ('Enter Number of Hours Worked: ')

7) Convert the string hours to integer using int() function
hours = int (hours)

8) print or echo Hours Worked to the output
print ('Hours Worked: ', hours)

9) Prompt the user to Enter Hourly Rate, read it and store it
rate = input ('Enter Hourly Rate: ')

10) Convert the rate to float using float() function
rate = float (rate)

11) print or echo Hourly Rate to the output
print ('Hourly Rate: ', rate)

12) Calculate Gross Pay
gross_pay = hours * rate

13) print grossPay to the output screen using .format () function
print ('Gross Pay is = $ {:,.2f}'.format(gross_pay))

Step 8) Click **Run Project**  button to Start Running the program

The following output appears on the Left side of the screen with the Input you entered and the correct calculated GrossPay $400.

If any Syntax Errors Found Do Next Step 9:

```
Enter your First Name: Ogar
Your First Name is:  Ogar
Enter your Last Name: Haji
Your Last Name is:  Haji
Enter Number of Hours Worked: 40
Hours Worked:  40
Enter Hourly Rate: 10
Hourly Rate:  10.0
Gross Pay is = $ 400.00
```

Step 9) Debug the Program: Debug or Correct any Syntax Errors until you have a clean Compiled program. (5 Points) (Clean compiled program means No Errors in the program).

Step 10) Test the Program: Test the Program with Test Data. (5 Points)

Repeat Step 10) Test the program many Times and Test the Program again and again until All conditions are tested:

```
file:///C:/Users/Ogar/Documents/-- 00- 2 Spring 2014/CIS142 Visual C#
Enter your First Name: Mary
Your First Name is: Mary
Enter your Last Name: Smith
Your Last Name is: Smith
Enter Number of Hours Worked: 30
Hours Worked: 30
Enter Hourly Rate: 11.23
Hourly Rate: 11.23
Gross Pay is = 336.9
Mary Smith 30 11.23 336.9
Your Gross Pay is = $336.90
```

Step 11) Documentation (5 Points): You have to add more comments to the Program (like Comments about the Purpose of the Program, Your Name and the Date the Program was written.)

```
##################################################################
# Purpose of the Program:
#   a) This Program will calculate Gross Pay.
#   b) It will ask the User to Enter Employee's Full Name:
#   c) It will ask the User to Enter Number of Hours Worked
#   d) It will ask the User to Enter Hourly Rate.
#   e) The program will calculate Gross Pay.
#       Gross Pay = Hours * Rate
#   f) Display Gross Pay
##################################################################
```

Step 12) Print a Copy of Python Code along with screen printout of the Running program. Submit to your Instructor the Print Copy and the screen Printout (Snaps) along with the following: (Which you did on Paper)

Copy the Python Code and the result of the program and Paste it in Microsoft Word program:

1) Purpose of the Program.
2) Input: how the Input looks like
3) Processing and Calculations
4) Output: how the Output will look like
5) Flowchart 6) Pseudocode
7) Python Code and
8) Print out copy of Python code and Output after running the program.

Submit the Programs on Time.
Remember Points will be deducted (20%) for Programs submitted Late.

Important Note:
1) Do Steps 1 thru 7 on Paper.
2) Then Get into Python IDLE IDE Code Editor
3) Type the Python code.
4) Save All the Files
5) Run the Program and Test it with Test Data for All Conditions.

Modify the Project and add the following to print to the Console using Format Specifiers (%s, %d, %,.2f, %n) with print() method.

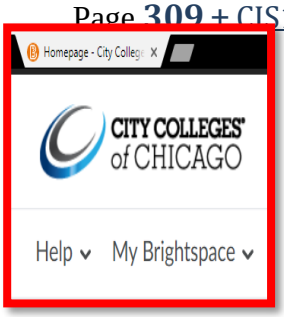
To print the Results using Format Specifiers: %s %d %f %.2f %n

```
#Using Format Specifiers (%s, %d, %f, %,.2f) to print out to output
  print ('%s       %s       %d    %f   $%,.2f ' % (
          first_name, last_name, hours, rate, gross_pay)  )
```

or

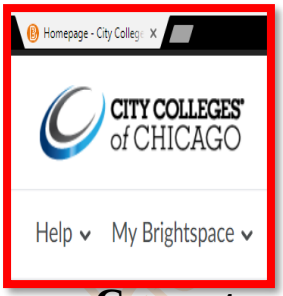
```
# print the Formatted Gross Pay to output console using .format () function

print ('Gross Pay is = $ {:,.2f}'.format(gross_pay) )
```

When Modifying a Project
Do Only 1 Modification at a Time



Upload Your Weekly Assignments: To Brightspace Correct Weekly Assignments Folder

+++ Review +++



Lesson 152 + How to Upload Your Weekly Assignments to Brightspace Correct Weekly Assignments Folder?

You must Upload your Weekly Assignments and Homework to Brightspace correct Weekly Assignments Folder as following:

1) Copy the Python Code from Python IDLE to Word document:

1. Copy the **Python Code from** the **Python IDLE IDE** and **Paste it into** the **Microsoft Word Document.**
2. In IDLE IDE, press **Ctrl+A (select All)** to **select All** the **Python code**.
3. Press **Ctrl+C (Copy)** to **Copy** the **selected Python code** into **computer memory RAM.**

2) Paste the Python Code into Microsoft Word:

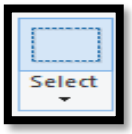
1. Get into **Microsoft Word document then** press **Ctrl+V (paste) to Paste** the **copied Python code** from **memory** into **Word document.**
2. Press **Ctrl+Home (go to the Top of Document)** and **type** your **Full Name** at the **top of document followed by** the **Python File Name** in **size 20 and bold.**

3) Print the Screen of the Output of Python Python IDLE:

1. Run the **Python project** and **make sure the program** is **running** with **correct output.**
2. Press **PrintScreen button** 

to **capture** the **output screen shot.**

4) Paste the Print Screen of Python output into Paint progra

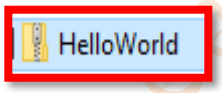
1. Get into **Paint program** and press **Ctrl+V (paste) to Paste** the **screen shot** in **Paint** program.
2. Inside the **Paint program,** Click **Select** 

icon and then **Select only** the **Output** of the **Python project.**
3. Press **Ctrl+C (Copy)** to **Copy** the **selected output image.**

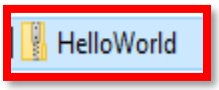
5) Get back into Microsoft Word program:

1. Go back to **Word Document,** press **Ctrl+End (End of Document)** to **go to End of document.**
2. In the **Word document,** press **Ctrl+V (Paste)** to **paste** the **Python output** there.
3. Save the **Word Document** as the **Name of the Python project** and in this **example (Save File as HelloWorld project)**

6) To Compress or Zip the Python project:

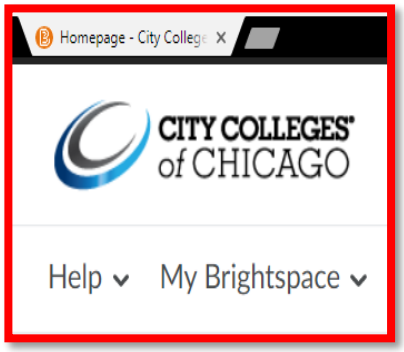


1. Right-Click on the **Python project (HelloWorld)** that is **saved on your computer**.
2. Point to **Send to,** then **click on Compressed (Zipped) Folder** and you will have **another File** which is Compressed or Zipped.



7) To Upload the 2 Files to Brightspace:

1. Log on Brightspace with your **User Name** and **Password**
2. Click on your course **CIS 103 Python** to **Select it**.
3. Click on **Assignments**
4. Click on **Assignments**
5. Click on **Week 03 Assignments Folder**
6. Click on **'Add a File'** button
7. Click on **'My Computer'**
8. Click on **Upload** button



9. Go to the **location** where you **saved the Python project 'HelloWorld'.**
10. Click on the **File** or **Folder (HelloWorld)**
11. Click on **Add** button and the **File** or **Folder** will be **added** to the **Week 01 Assignments Folder.**

Note: Always **Upload** to **Brightspace the Modified Python Project:**
1) The **Microsoft Word Document** of the **Python Project Code along with the Python Output Screen shots.**

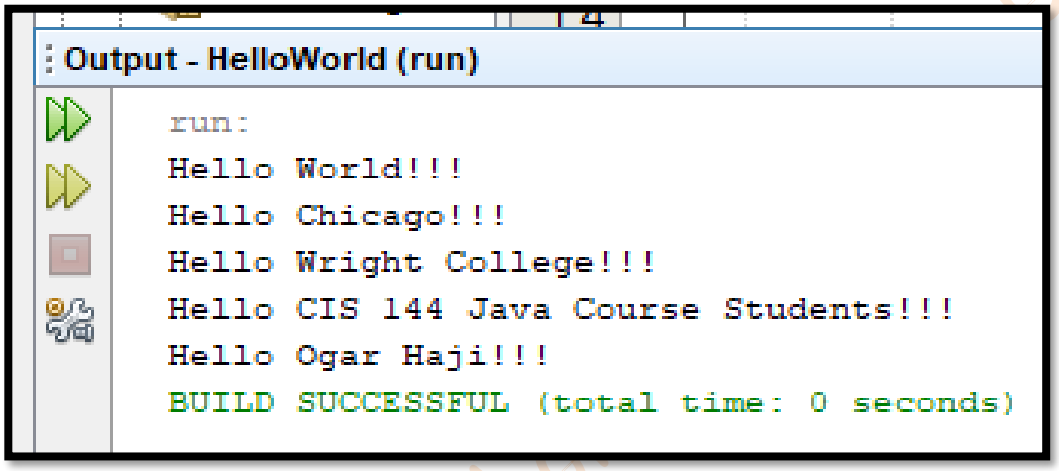
The Python project 'HelloWorld' code in Word Document along with the Output Screen Shots appear as following:

Ogar Haji (Your Full Name)

CIS 103 Python

Project Name: HelloWorld

(Always Place the Screen Output at the Top Before Python Code)



'''

Project Name: HelloWorld

This Python project will print the message 'Hello World' to screen

Programmer: Instructor + Ogar Haji (Type your Full Name)

Date: June 01, 2017

'''

This project will print the Literal String 'Hello World' to output screen

print ('Hello World!!!')

print ('Hello Chicago!!!')

print ('Hello Wright College!!!')

print ('Hello CIS 103 Python Course Students!!!')

print ('Hello Ogar Haji!!!')

help (len)

Using help (len)
To get help on len() length function

len('Python')

Lesson 153 + How to use help (len) to get help on Length function?

```
>>> help (len)
Help on built-in function len in module __builtin__:
len(...)
    len(object) -> integer
    Return the number of items of a sequence or mapping.
```

help (len)

len ('Python') built-in Python function will return the **Length (Number of Characters)** of the **string 'Python'** which is **6 characters.**

```
>>> len ('Python')
6
>>> len ('Wright College')
14
>>> name = 'Mary Smith'
>>> len (name)
10
```

len ('Python')

{0}, {1}

Using format and Place Holders {0}
To Format and print Strings & Numbers

.format(n1,n2)

Lesson 154 + How to use Format and Place Holder to print Strings & Numbers?

Use Python **.format() method** and **Place Holders {0}, {1}, {2}** to **Format Strings** and **Numbers**.

```
# 1) Declare the Local variables to be used in this method
number1 = 4
number2 = 5
result = 0
# 2) Calculate the result of multiplying number1 and number2
result = number1 * number2
# 3) Use print() function to print the result of Multiplication
print ('Multiplication of {0} and {1} is = {2}'.format
                        (number1, number2, result) )
```

or using the **f** string format (This is very easy)

```
print (f'Multiplication of {number1} and {number2} is = {result}' )
```

Sequence
Selection
Iteration
Functions

Control Structure Forms used in Python:
1. Sequence 2. Selection 3. Iteration 4. Functions
+++ (Read and Study This Lesson) +++

Sequence
Selection
Iteration
Functions

Lesson 155 + What are the Control Structures or Forms used in Python language?

You can **Control the Order of Python Code** to be **executed** in Python Language.

The following are the Control Structures used in Python Language:

1. **Sequencing Structure:** **Sequence Structure** in programming means that the **computer will run or execute your code in Sequence**, **one Line after another from** the **top** to the **bottom** of your **program**. So far, we have been doing **Sequence Structure** Programming. For example, **these Python Statements** will be **executed** in the **Order Listed one Line after another Line** .

```
average = input ('Enter Student Average: ')
average = float (average)
print ('Student Average is = ', average)
```

```
Enter Student Average: 97
Student Average is =  97.0
```

2. **Selection Structure:** In **Selection Structure**, you will use the **if Selection Statement** to **Check** for **certain Conditions** to be **True** or **False** and then Python will execute those **True** or **False Statements**. Use **if** statement for **selection** as shown below:

```
average = 97
if average >= 70:
    print ('The Student is Passing')
else:
    print ('The Student is Failing')
```

```
Output - JavaApplication82 (run)
run:
The Student is Passing
BUILD SUCCESSFUL (total
```

3. Iteration Structure: **Iteration Structure** is when you want **to Iterate, Repeat** or **Loop Many Times**. You can use '**for**' statements **when you know in advance** the **Number** of **Times to loop**.
Use '**while**' statement to **prompt the user** if he/she **wants to continue looping** or **the Number** of **Times to loop**.
There are **Two types of Loops**: **'for' loops and 'while' loops**.

```
# To print your Name 100 Times, using 'for' statement
for i  in range (100) :
      print ('Ogar Haji')
# To print your Name 100 Times, using 'while' statement
counter = 1       # declare a variable 'counter' outside of 'while' loop
while  counter <= 100:
      print ('Ogar Haji')
      counter += 1       # increment counter by 1
```

4. Functions Structure: A **Function() Structure** is a **Block of Code grouped** together to perform a specific operation and the **Function only runs when it is called by the Function Name**.

A **Python Function must be Terminated** with a **:** and **all the statements within** the **Function Must be indented** with the **same number** of **spaces like 4 spaces**. (**default indentation is 4 Spaces**).
You can **pass Data as Parameters** into a **function()**.
Functions() are used to perform certain actions, like the multiplications of 2 Numbers: Ex, **multipy_2_numbers()**.
A **Function() has () Parenthesis after** it with No Parameters which **means** that this is a **Function and a function** can **have Parameters between ( )** or with **No Parameters**. For example,
The following Function is an **Empty Function** with **No Parameters**.
def multiply_2_numbers(): # An empty function with No Parameters
While the following is a **Function** with **2 Parameters (n1 and n2)** that will be **passed to it.**

def multiply_2_numbers (n1, n2):

Use **Functions()** to **reuse code as many times as you wish**.
You **define** the **Function** and **code** it only **Once** and you c**an call** and **use** the **Function** as **many times** as you wish.

You **call** the **multiply_2_numbers() Function** from the **main()** Function like the following:

 # Call the Function **multiply_2_numbers()** Function
multiply_2_numbers()
The following line will be printed out when Python returns from
executing multiply_2_numbers() Function
print ('Return from Function call')

To Define & Code a Function called multiply_2_numbers():

def multiply_2_numbers():
    # 1) Declare the **Local variables** to be used in this Function
    number1 = 4
    number2 = 5
    # 2) Calculate the result of **multiplying number1 and number2**
    result = number1 * number2
    # 3) Use **print()** function and **.format** to print the result of Multiplication.

Note that **{0}** and **{1}** are called Place Holders and use start numbering them from 0 {0} and increase by 1 {1} and {2}. So **{0}** will be replaced with **number1** and **{1}** will be replaced by **number2** and so on.

```
print('The Multiplication of {0} and {1} is = {2}'.
                format (number1, number2, result) )
```

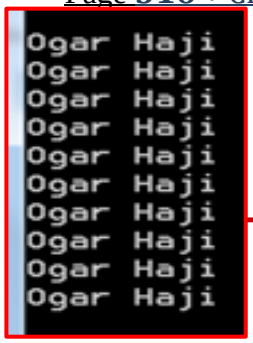
or Better and Use This: Note: no space after f

```
print(f'The Multiplication of {number1} and
{number2} is = {result}')
```

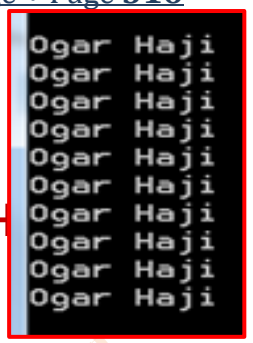
The output will look like the following:

```
The Multiplication of 4 and 5 is = 20
```

```
The Multiplication of 4 and 5 is = 20
```



Using 'for' statement to: Print your Full Name 'Ogar Haji' to Output Screen Many Times (10 Times) ++ (Do Lab Exercise 1) 100 Points ++



Do Lab Exercise 1

Lesson 156 Ex + How to Use For statement to Print Your Full Name 10 Times to output Console?

Problem or Project: Design and Code in Python Language the project to Print Your Name (Ogar Haji) to output screen 10 Times.

There are 2 types of Loops in Python Language:

1) **Count-Controlled Loops:** In Python, use '**for**' statement for **Count-Controlled Loops** when you **know** in **Advance** the **Number** of **Times** a **Loop will occur** or **happen**.
2) **Condition-Controlled Loops:** In Python, use '**while**' statement for **Condition-Controlled Loops** when you **Test a Condition**, and the **condition** is **True**.

You can use **'for' statement** to **loop** and **print number of times**. Use **'for'** statement **when** you **know in advance how many times to loop (like 10)**.

You can **increment a counter** by **1. For Example:** **i = i +1** **or** **i += 1**

To Increment a **counter by 2,** you can do: **i = i + 2** **or** **i += 2**

To Increment a **counter by 3,** you can do: **i = i + 3** **or** **i += 3**

For example, If I want to print my Name 10 Times, I can do the following: Using print() function 10 times.

```
print ('Ogar Haji')
print ('Ogar Haji')
print ('Ogar Haji')
print ('Ogar Haji')
```

```
print ('Ogar Haji')
print ('Ogar Haji')
print ('Ogar Haji')
print ('Ogar Haji')
print ('Ogar Haji')
print ('Ogar Haji')
```

But what if I want to print my name 50 or even 100 times, then that with be tiresome and ridiculous.

You should use 'for' statement when you know in advance how many times to loop (This is Controlled Loop)

With **for** statement use a **variable name** like **'i'** which stands for **index**, the **range ()** **function** and the **value** to **loop** to **(10)** followed by a **:**

Remember to Indent the statements Inside the for statement.

```
for i in range (10) : # Use for statement to loop and print your full name 10 time and
# Indent the Block of statement within the for statement
for  i in range (10) :
    print ('Ogar Haji')
```

+ Do Lab Exercise 1 + Do Lab Exercise 1 +

1) Ex. Type the following Python Code and **Do Lab Exercise 1**

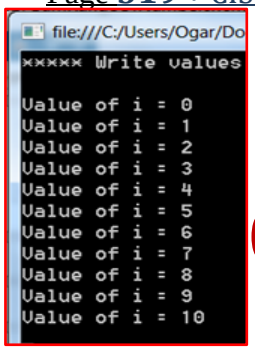
2) Save Python project as Chapter4LabEx-ForStatement.py

```
'''
Write your Comments here
Date: Wednesday, September 08, 2021
Python Programmer: Ogar Haji
'''
# 1) using for statement to Print your Full Name to output 10 Times
print ('** Print Your Full Name to Console 10 Times ***')
print ()          # print a blank line
# 2) Use for statement to loop and print your full name 10 times
for i in range (10) :          # use for in range (10) to loop 10 times
    print ('Ogar Haji')
# 3) Print End of Job Lines of Footer
print ()
print ('********** E n d  O f  J o b **************')
print ('******** Programmer: Ogar Haji ********')
```

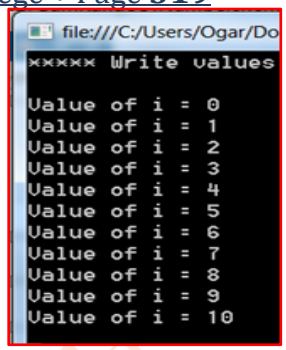
Do Lab Exercise 1

The Output of the For Python Project will look like the following:

```
= RESTART: C:/Users/ogarh/AppData/Local/Programs
** Print Your Full Name to Console 10 Times ***

Ogar Haji
Ogar Haji
Ogar Haji
Ogar Haji
Ogar Haji
Ogar Haji
Ogar Haji
Ogar Haji
Ogar Haji
Ogar Haji

********** E n d  O f  J o b **************
******** Programmer: Ogar Haji ********
```



Using 'for' statement to: Print out the Numbers from 0 till 10 to Output Console Screen (Continue Lab Exercise 1) 100 Points



Continue Lab Exercise 1

Lesson 157 Ex + How to Use For statement to print the Numbers from 0 till 10?

Problem or Project: Design and Code in Python Language the project to Print out the Numbers from 0 thru 10 to the output console screen.

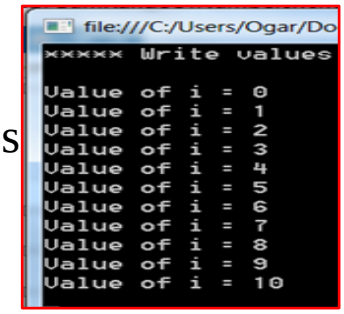
Using range with for statement:

You can **use for statement** to **loop** and **write Numbers** from **0 to 10**.
Use **for** statement when you **know in advance how many times to loop**.
It is very **common** to use **'i' variable (i means index)** with **'for'** statement in all programming languages including Python language.
To **use for** statement, you must **use the range 11)** as shown below:

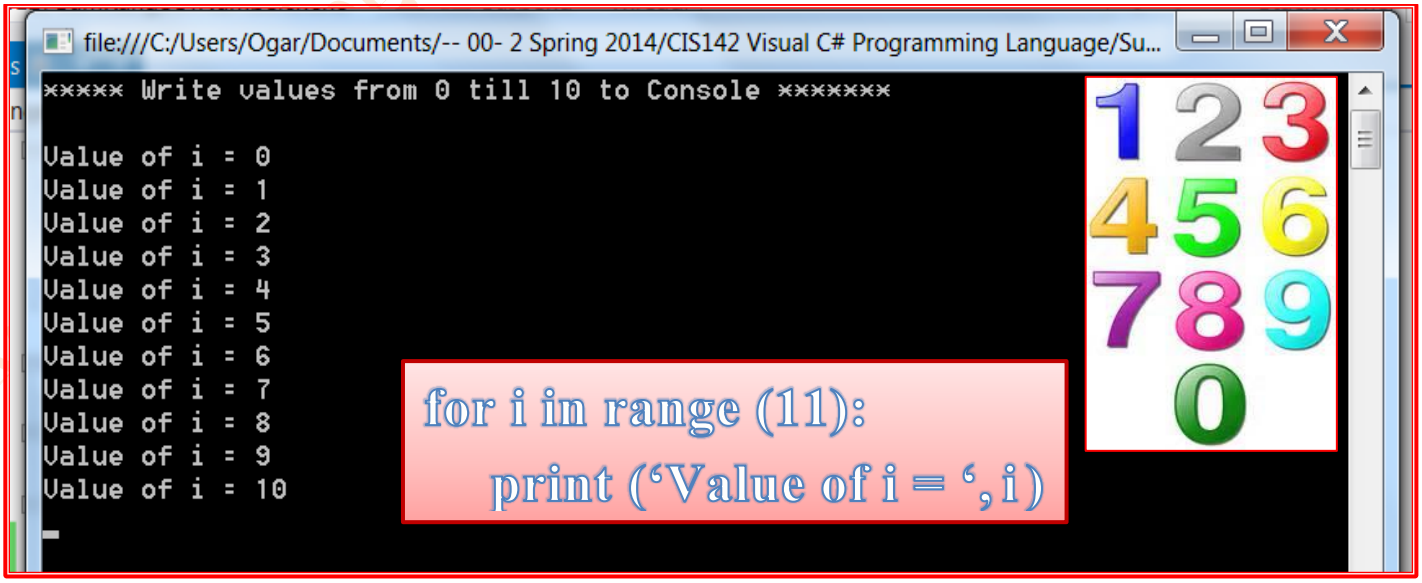
for a variable i in range (11): # this will loop from **0 till 10** (**11 times**)

```
for i in range (11) :      # from 0 till 10 which is 11 times
    print ('Value of i = ', i )
```

Indent



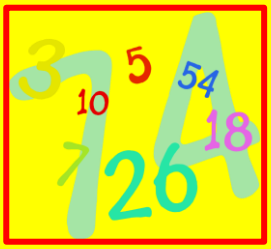
Use For statement to Print to output console the Numbers 0 to 10:



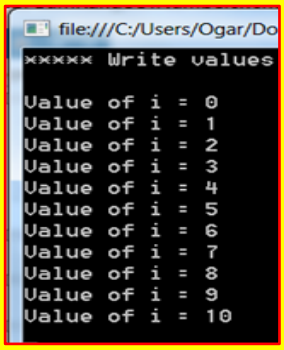
+ Continue Lab Exercise 1 + Continue Lab Exercise 1 +

1) Ex. Type the following Python Code and **Continue Lab Exercise 1**

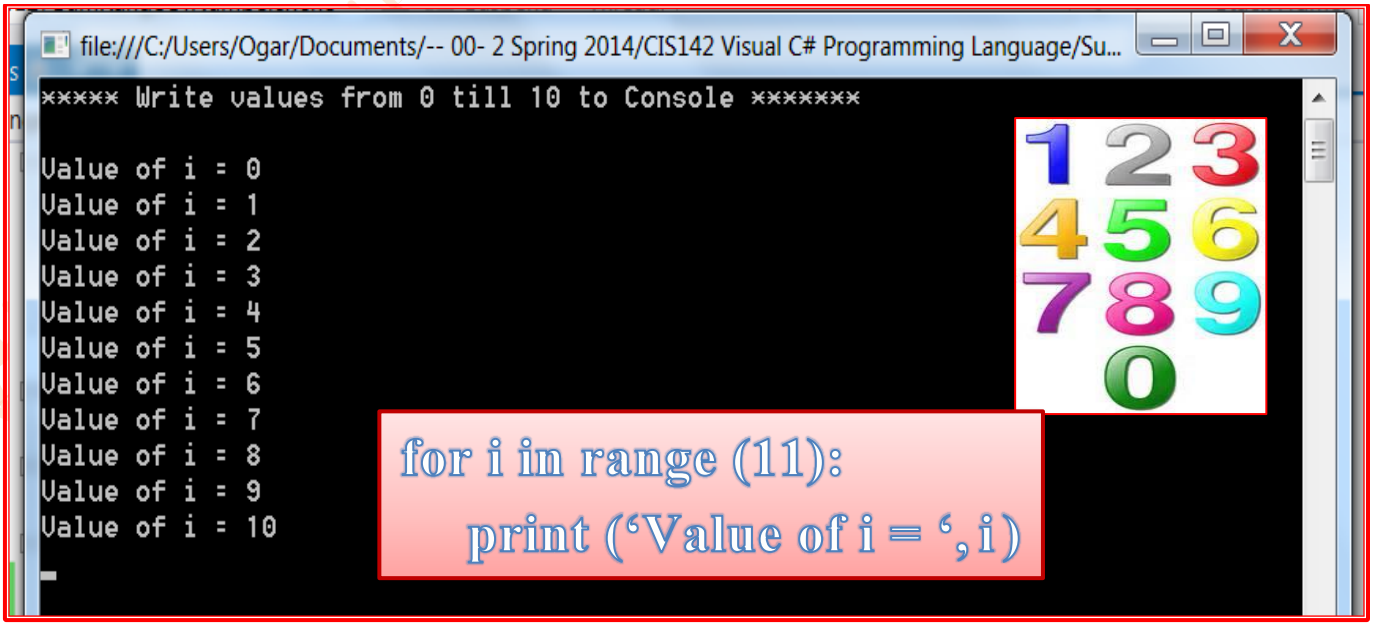
2) Save Python project as Chapter4LabExe-PrintNumbers.py




```
'''
Write your Comments here
Date: Wednesday, September 08, 2021
Programmer: Ogar Haji
'''
# 1) Print out the Headings of the project to the Console
print ('** Print out the values from 0 till 10 to output **')
print ()
# 2) Using for statement to print the Numbers from 0 till 10
for i in range (11) :    # from 0 till 10 which is 11 times
    print ('Value of  i = ',  i )
```



The Output of the Python program to Print the Numbers 0 thru 10:



Using 'for' statement to: Sum a Range of Numbers From 0 till 100 + (Continue Lab Exercise 1) 100 Points

Continue Lab Exercise 1

Lesson 158 Ex + How to Use For statement to Sum Up a Range of Numbers from 0 till 100 in Python language?

Problem or Project: Design and Code in Python Language the project to Sum up a Range of Numbers from 0 till 100.

You can use for statement to Sum a Range of Numbers from 0 to 100.

You can increment a counter by 1. For Example: i = i +1 or i += 1

To Increment a counter by 2, you can do: i = i + 2 or i += 2

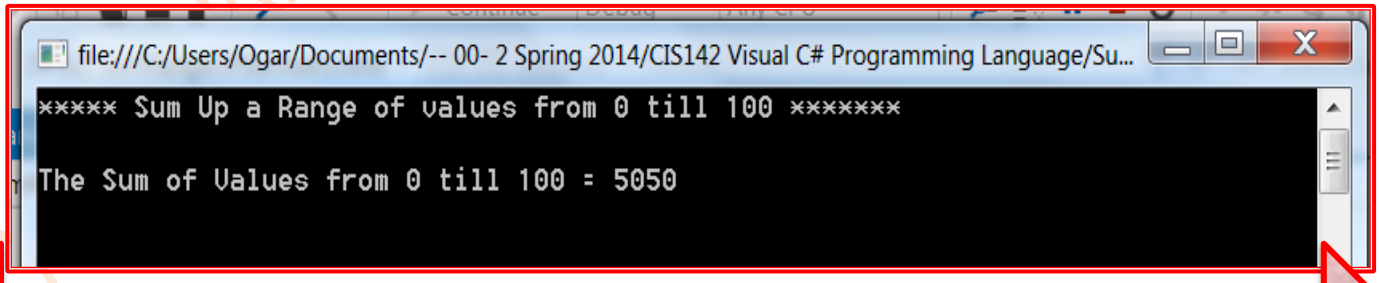
To Increment a counter by 3, you can do: i = i + 3 or i += 3

Declare the **Local variables** to be used in this project & **initialize to 0**

```
sum = 0

for i in range (0, 101):
    sum += i          # or sum = sum + i
    print ('Value of Sum = {0} \n'. format(sum) )

print ('The Sum of values from 0 till 100 = {0} \n'. format (sum) )
```

```
for i in range (0, 101):
    sum += i
```

```
sum = 0
sum = 1
sum = 3
sum = 6
sum = 10
sum = 15
.............
sum = 55
```



+ Continue Lab Exercise 1 + Continue Lab Exercise 1 +

1)Type the following Python Code and Do Lab Exercise 1

2) Save Python project as **Chapter4LabExe-SumRangeOfNumbers.py**

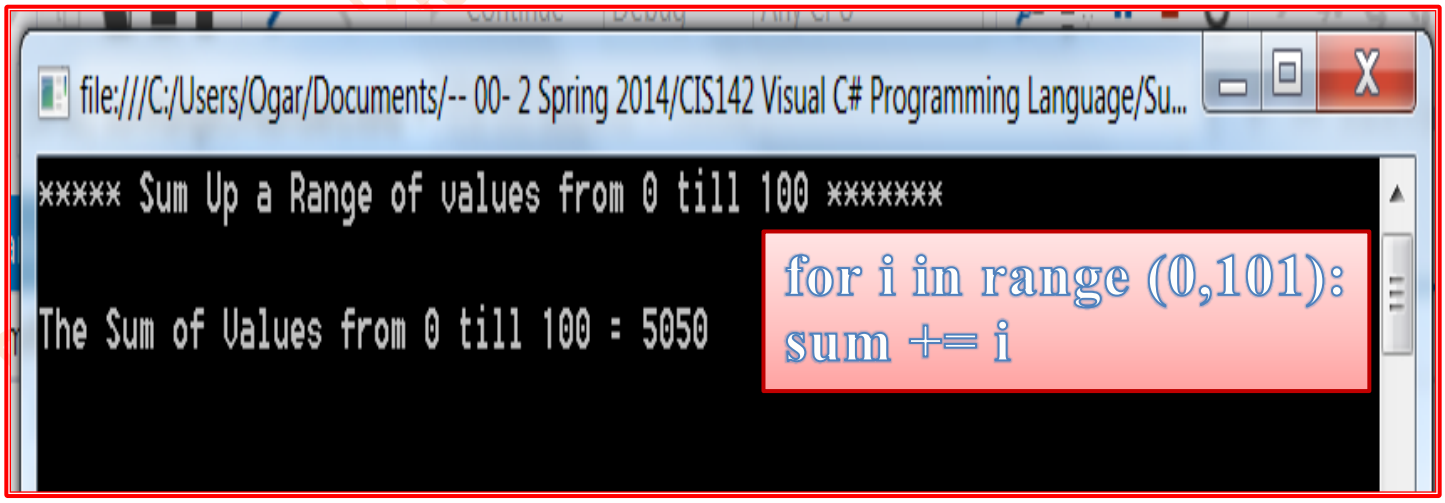

```
# 1) Add Comments about the project on the top of the project.
'''
      Write your Comments here
       Date: Sunday, February 05, 2023
       Programmer: Ogar Haji
'''
```

Do Lab Exercise 1

```
#1) using 'for' statement to Sum Up values from 0 till 100
print ('*** Sum up a Range of values from 0 till 100 ***')
print ()

# 2) Initialize to 0 the Local Variables to be used in the Project
sum = 0

# 3) Use 'for' statement with range to loop 100 times
for i in range (101) :
      sum = sum + i          # or sum += i

# ) Print the sum of numbers from 1 to 100
print ('The Sum of values from 0 till 100 = ', sum)
```

```
sum = 0
sum = 1
sum = 3
sum = 6
sum = 10
sum = 15
.............
sum = 55
```

The Output of the Python program is shown below:



Modify This Project to Do the Following Modifications: 30%

1) **Modify the Python project to Sum the Numbers from 0 till 10 (remember to go to 11 to include 10).**
2) **Modify the Python project to Sum the Numbers from 0 till 50.**
3) **Sum** the **Odd Numbers (1, 3, 5… 99)** from **1 to 100 in increment of 2.**

```
for i in range (start_value, end_value, step)
for i in range (1, 101, 2) :   # step 2 or increment by 2
    print ('Value of i = ', i)
    sum = sum + i  # or sum += i
```

```
for i in range (1, 101, 2):
```

4) **Sum** the **Even Numbers (2, 4, 6…. 100)** from **2 to 100**.

Modify Lab Exercise 1

```
# This program demonstrates a simple for loop
# that uses a list of numbers.
print ('I will display the numbers 1 through 7.')
for num in [1, 2, 3, 4, 5, 6, 7]:      # include the list in Brackets [ ]
    print (num)
```

```
= RESTART: C:/Users/ogarh/AppData/Local/
I will display the numbers 1 through 7.
1
2
3
4
5
6
7
>>>
```

```
for num in [1,2,3,4,5,6,7]:
```

Using colors_list to store a list of colors:

This program also demonstrates a simple for
loop that uses a **list of strings**.

colors_list = ['red', 'green', 'blue', 'purple', 'yellow']

red	purple	blue	green	orange	yellow
0	**1**	**2**	**3**	**4**	**5**

in the above colors list:
red has an **index i value** of **0,**
purple has **index i value** of **1**
blue has **index i value** of **2**
green has **index i value** of **3**
orange has **index i value** of **4**
yellow has **index i value** of **5**

```
# 1) Create a List of colors and assign it to list 'color'
colors_list = ['red', 'green', 'blue', 'purple', 'yellow']

# 2) Use 'for  color in' to loop and print all items in the list 'colors'
for color in (colors_list):
    print (color)
```

The Output of Create a List of Colors

```
red
green
blue
purple
yellow
>>>
```

```
Colors_list = ['red', 'green', 'blue']
for color in (colors_list):
    print (color)
```

To Check for Invalid Data in Python Language:

```
# Get a test score and convert to integer using int() function
score = input ('Enter a Test Score (between 0 and 100): ')
score = int (score)
# Check if score is less than 0 or greater than 100 (Invalid Input),
# then prompt the user for a Valid Input (between 0 and 100).
```

while score < 0 or score > 100:

```
while score < 0 or score > 100:    # check for Invalid data
    print ('ERROR: The score cannot be Negative')
    print ('or Greater than 100.')
  # Then prompt user to enter a Valid number (between 0 and 100)
    score = int (input ('Enter the test score: (between 0 and 100): '))
```

Using 'for' statement to:

Sum a Range of Numbers Prompted By User

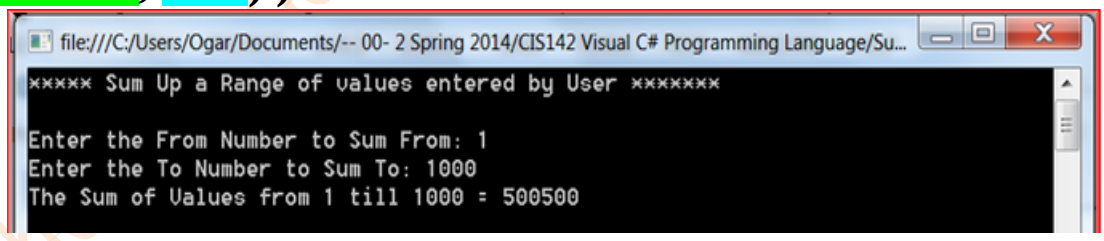
+++ (Read and Study This Lesson) +++

Lesson 159 + How to Use For statement to Sum Up a Range of Numbers Prompted by User in Python language?

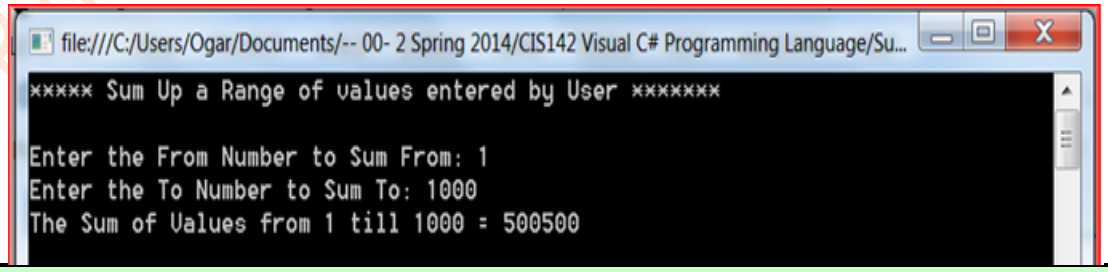
Problem or Project: Design and Code in Python Language the project to Sum up a Range of Numbers Prompted by User from a Number till the Number to Sum Up to.

You can prompt the user to Enter the 'fromNumber' to start summing from and prompt to enter 'toNumber' to Sum to that Number.

```
# 3) Prompt the User to Enter the From Number to Sum From
from_number = input ('Enter the From Number to Sum From: ')
from_number = int (from_number)
# 4) Prompt the User to Enter the To Number to Sum To
to_number = input ('Enter the To Number to Sum To: ')
to_number = int (to_number)
# 5) Use for statement to Sum up the Range of Numbers
sum = 0
for  i  in range (from_number, to_number) :
      sum += i       # or sum = sum + i
print ('The Sum of Values from {0} to {1} = {2}' .format(from_number, to_number, sum) )
```



1)Ex. Modify project SumRangeOfNumbersPromptByUser



if i==7:
break

Using 'break' keyword to: Break out or Exit from Loops ('for' or 'while') +++ (Read and Study This Lesson) +++

if i==7:
break

Lesson 160 + How to Use Break keyword to Break or Exit from a Loop in Python?

Use 'break' keyword to **Break out** or **Exit** from a **Loop** like **for** or **while Loops**.

- The **break** statement, when **executed in a while, for, do…while** or **switch**, causes **immediate exit** from that statement.

break

- **Execution continues** with the **first statement after the control statement**.
- **Common uses** of the **break** statement are **to escape early from** a **loop** or to **skip** the **remainder** of a **switch**.

```
for i in range (11):      # you can use any variable for 'i'
    # Check if 'i' is equal to 7, then break out of the for loop
    print ('Value of  i = ', i )

    if i == 7 :
        print ('I am breaking out when i =  ', i )
        break
# End of for loop statement
print ('Value of  i = ',  i )      # End of 'for' statement
```

break

```
Value of  i =  0
Value of  i =  1
Value of  i =  2
Value of  i =  3
Value of  i =  4
Value of  i =  5
Value of  i =  6
Value of  i =  7
I am breaking out when i =  7
Value of  i =  7
```

The Output of the 'break' statement:

```
Value of  i =  0
Value of  i =  1
Value of  i =  2
Value of  i =  3
Value of  i =  4
Value of  i =  5
Value of  i =  6
Value of  i =  7
I am breaking out when i =  7
Value of  i =  7
```

break

if i==13:
    continue

Using 'continue' keyword to: Continue for Next Iteration in the Loop ++(Read and Study This Lesson)++

if i==13:
    continue

Lesson 161 + How to Use 'continue' keyword to continue for the next for iteration in a Loop in Python?

Use 'continue' keyword to **continue** for the **Next Iteration** in the **head Loop**.

The '**continue**' keyword is used in Python language if you want to **skip** the **remaining statements** in the **loop body** and **proceeds** with the **next iteration** of the **loop**.

continue

- In **while** and **do**…**while** statements, the program evaluates the loop-continuation test **immediately after** the **continue** statement executes.
- In a **for** statement, the **increment expression executes**, then the **program evaluates** the **loop-continuation test**.
- **Here below, I want to use for statement to print numbers from 0 up to 16 but Not printing the Number 13:**

```
for i in range (16):
    # Check if 'i' is equal to 13, then continue from the top of for loop
    if i == 13:
        print ('I am Continuing from Top when i =', i )
        continue
    else:
        print ('Value of  i =', i )
    # End of the if statement
# End of for statement
print ('Value of  i =', i )
```

continue

```
Value of  i = 0
Value of  i = 1
Value of  i = 2
Value of  i = 3
Value of  i = 4
Value of  i = 5
Value of  i = 6
Value of  i = 7
Value of  i = 8
Value of  i = 9
Value of  i = 10
Value of  i = 11
Value of  i = 12
I am Continuing from Top when i = 13
Value of  i = 14
Value of  i = 15
Value of  i = 15
```

continue

The Output of the 'continue' statement:

Notice that Value of i = 13 is Skipped and Not Printed.

```
Value of  i = 0
Value of  i = 1
Value of  i = 2
Value of  i = 3
Value of  i = 4
Value of  i = 5
Value of  i = 6
Value of  i = 7
Value of  i = 8
Value of  i = 9
Value of  i = 10
Value of  i = 11
Value of  i = 12
I am Continuing from Top when i = 13
Value of  i = 14
Value of  i = 15
Value of  i = 15
```

continue

for num in range (1, 10, 2)

Using range with 3S (start:stop:step) To Print Odd Numbers

+++(Read and Study This Lesson)+++

for num in range (1, 10, 2)

Lesson 161 + How to Use range with (start:stop:step) to Print Even Numbers from 1 to 10 in a for Loop in Python?

We can also use with range (start,stop,step) the 3 s.
For example,
start = 1 means the for loop will **start with number 1.**
stop = 10 will **stop at number 10**
step = 1 (default) will **increment by 1**

```
for i in range (1, 100, 1):
    print (i)
```

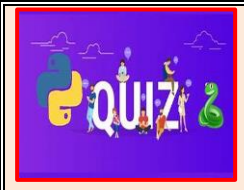
```
# remember 3 S with range (start,stop,step)
for num in range (1, 10, 2):
    print (num)    # Starts with 1, increments by 2, till 10
```

The Odd Numbers are the following:

```
= RES
n311/f
1
3
5
7
9
```

To Print Even Numbers, I will start with 0 or 2:

```
for num in range (2, 10, 2):
    print (num)    # Starts with 2, increments by 2, till 10
```



Chapter 4 + Quiz 1 + Test Your Python Language Knowledge:

---

1. What is the output after executing the following Python code?

```
for i in range (5):          # this for in range loop will loop from 0 till 4
    print (i, end = '  ')
```

A. 12345 B. 0 1 2 3 4 5 C. 0 1 2 3 4 D. 1 2 3 4 5 E. none

---

2. What is the output after executing the following Python code?

```
for num in range(1, 10, 2):    # remember 3 S with range (start:stop:step)
    print(num, end = ",")      # Starts with 1, increments by 2, till 10
```

A. 2,4,6,8,10 B. 2,4,6,8 C. 1,3,5,7,9, D. 1,3,7,9 E. none

---

3. What is the output after executing the following Python code?

```
number = 13
if (number % 2 == 0):   # % returns remainder if 0, means number is Even
    print ('Even')
else:
    print ("Odd")
```

A. even B. odd C. Even D. Odd E. error

---

4. What is the output after executing the following Python code?

```
a = 'Apple';  b = 'Orange'
x, y = b, a       # swap and assign b 'Orange' to x, assign 'Apple' to y
print (x, y, sep= "::")
```

A. Apple:Orange B. Apple::Orange C. Orange:Apple D. Orange::Apple

---

5. What is the output after executing the following Python code?

```
course = 'CIS 103 Python'
print(course [::] )        # here [::] means from start to end and step is 1
```

A. CIS 103 Python B. 'CIS 103 Python' C. CIS D. Python E. nothing

---

---

6. What is the output after executing the following Python code?

```
for i in range (10, 12, 1):      # start = 10, end = 12 (not inclusive), step = 1
    if i % 2 == 0:               # 10 % 2 is = 0 which means True
        print (f'{i} is Even')   # prints 10 is Even, and 11 is Odd
    else:
        print (f'{i} is Odd')
```

A. 10 is Even
11 is Odd
B. 10 is Odd
11 is Even
C. 10 is Even
D. 11 is Odd
E. error

---

7. What is the output after executing the following Python code?

```
a = 1 ;  b = 1
while a < 2:          # here, a = 1 which is less than 2 (True)
    while b < 2:      # here, b = 1 which is less than 2 (True)
        print (a, ":", b)      # prints 1: 1
        b += 1                 # b = 2
        a += 1                 # a = 2
```

A. 1 : 2
B. 2 : 2
C. 2 : 1
D. 1 : 1
E. error

---

8. What is the output after executing the following Python code?

```
language = 'Python'
i = 0
while i < len (language):      # length of 'Python' is 6
    print (i, end = '*')       # i is = 0, so it will print 0*
    i += 1
                               # Increment i by 1 which is = 1
```

A. 0*1*2*3*4*5*
B. 1*2*3*4*5*6*
C. 6
D. 5
E. error

---

9. What is the output after executing the following Python code?

```
print (2 * 4 % 7)   # from left to right, 2 * 4 = 8, then 8 % 7 = 1 remainder
```

A. 0
B. 1
C. 4
D. 8
E. error

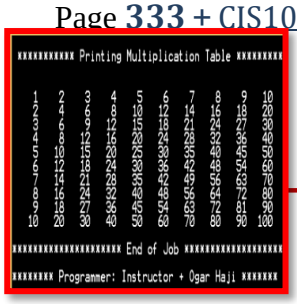
---

10. What is the output after executing the following Python code?

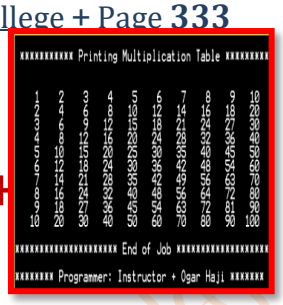
```
for i in range (11):          # you can use any variable you want for 'i'
   if i == 0 :                # i = 0 and 0 == 0, this is True
      print ('Breaking out of for, i =', i )     # prints this and break out
      break
   else:
      print ('Still working in for.')
```

A. Breaking out of for, i = 0
B. Still working in for.
C. True
D. False

Answers Are Found at The End of This Chapter 04.



Nested 'for' Statement To Print a Multiplication Table ++(Do Lab Exercise 2) 100 Points ++



Do Lab Exercise 2

Lesson 162 Ex + How to Use Nested For Statement to Print Multiplication Table from 1 to 10?

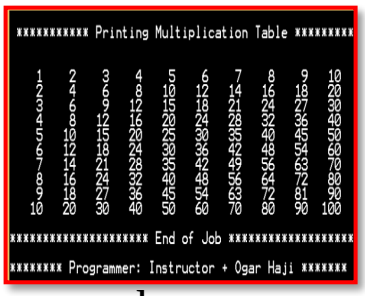
Problem or Project: Design and Code in Python Language the project to Print a Multiplication Tables of Numbers using Nested For statement.

Do the 12 Must Steps to Design, Code & Solve a project in Python Language

You can **use Nested For statements** to **print** a **Multiplication Table.**
A **Nested For loop** is a **for loop inside** the **body** of the **outer for loop.**
With **Nested for statement** here, we will **have an Outer for Statement** to **print the numbers of Rows.**
Then **within** the **Outer for statement,** we will **add an Inner for statement** to **print** or **process** the **Columns.**

1) **The Execution will start with the Row for statement, and it will set its value to 1.**
2) **Then it will start executing the Column for statement which will be executed 10 times for each row loop.**

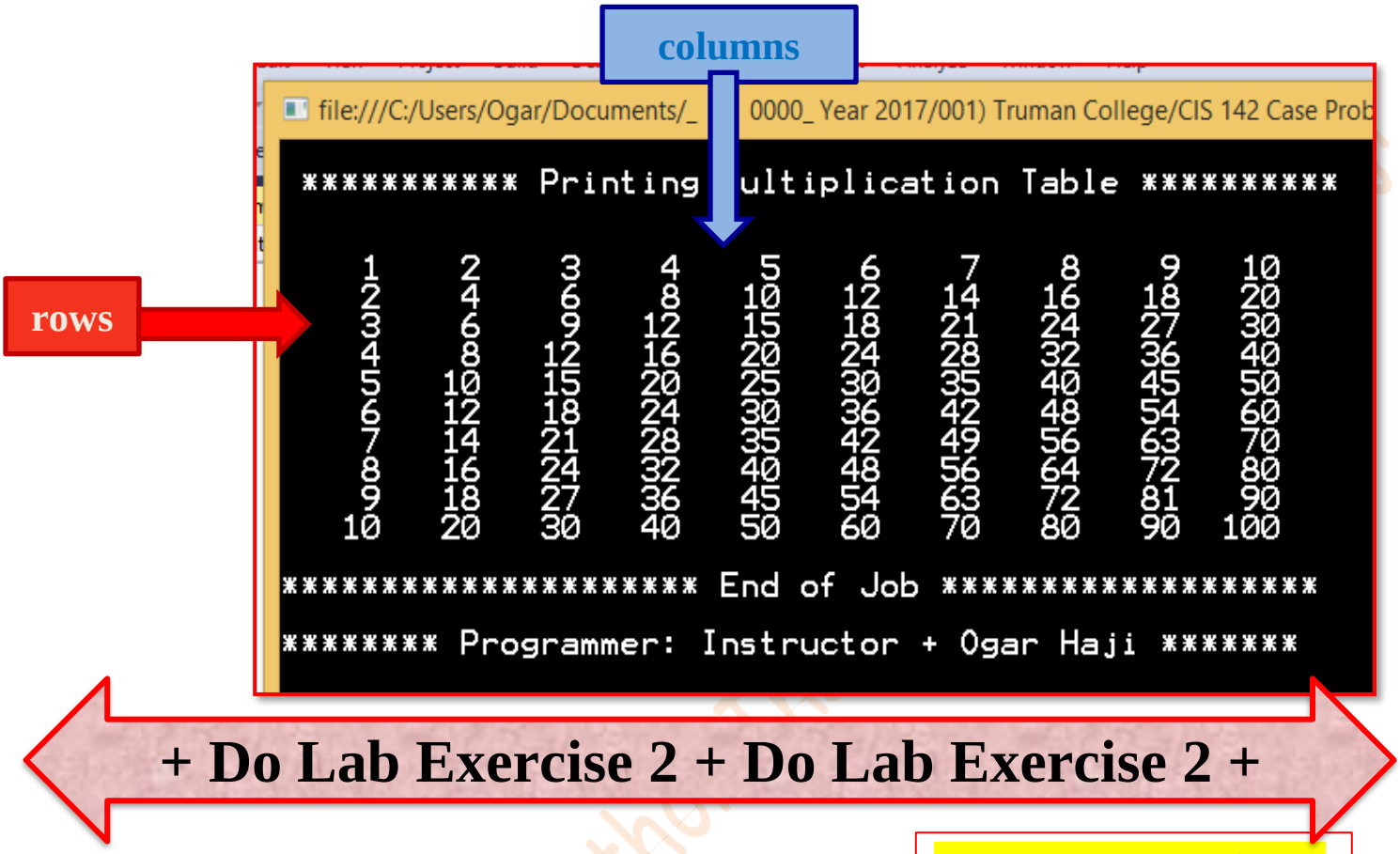
The **Inner For Columns** statement **will Execute 10 Times** and after the **Column count** is **10,** then it will **start again** to **execute** the **Row for statement.**




```
# 2) Use Nested for statement to print Multiplication Table
# 3) Print Rows from 1 to 10
for row in range (1, 11):  # to include row 10
    # 4) Print Columns from 1 to 10
    for column in range (1, 11) : #to include column 10
        # 5) print multiplication table with a tab \t spaces between them
        print (  row * column, end = '\t'  )
    # End of Second Inner for column loop
    # 6) Print a Blank Line after printing the 10th Column
    print ()  # print a Blank line
# End of First Outer for row loop
```

Do Lab Exercise 2

Below is a Multiplication Table of 10 Rows and 10 Columns:



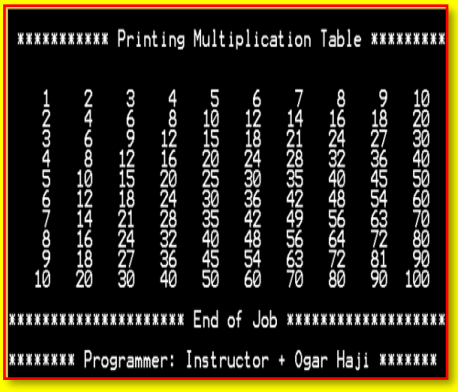
+ Do Lab Exercise 2 + Do Lab Exercise 2 +

Do Lab Exercise 2

1)Type following Lab Exercise in **Python IDLE**.

2) Save the Python project as **PrintMultiplicationTable.py**

Do Lab Exercise 2



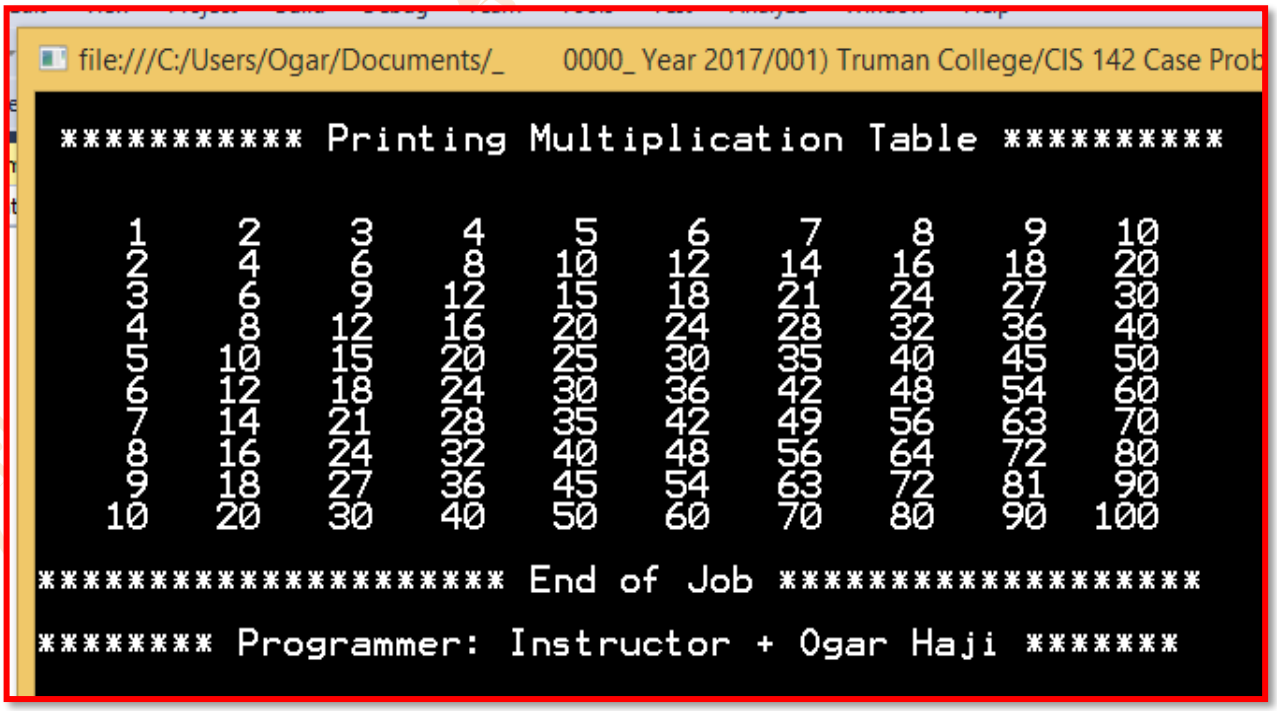

```
'''
 * This project uses Nested For statement
 * to Print Multiplication Table
 * Date written: March 22, 2017
 * Programmer: Instructor + Ogar Haji
'''

# 2) Print a simple Heading of the project
print ('\n **** Printing Multiplication Table ****\n')
```

```
# 3) Use Nested for statement to print Multiplication Table
# Print Rows from 1 to 10
for row in range (1, 11):    # to include row 10
    # 4) Print Columns from 1 to 10
    for column in range (1, 11): # to include column 10
        # 5) print multiplication table with a tab \t spaces between them
        print ( row * column, end = '\t'  )
    # End of the Second Inner for column statement
    # 6) Print a Blank Line after printing 10th Column
    print ()     # print a Blank line
# End of the First Outer for row statement
# 7) Print the Footers of the project
print ('\n********** End of Job **********\n')
print ('*** Programmer: Instructor + Ogar Haji **\n\n')
```

Column

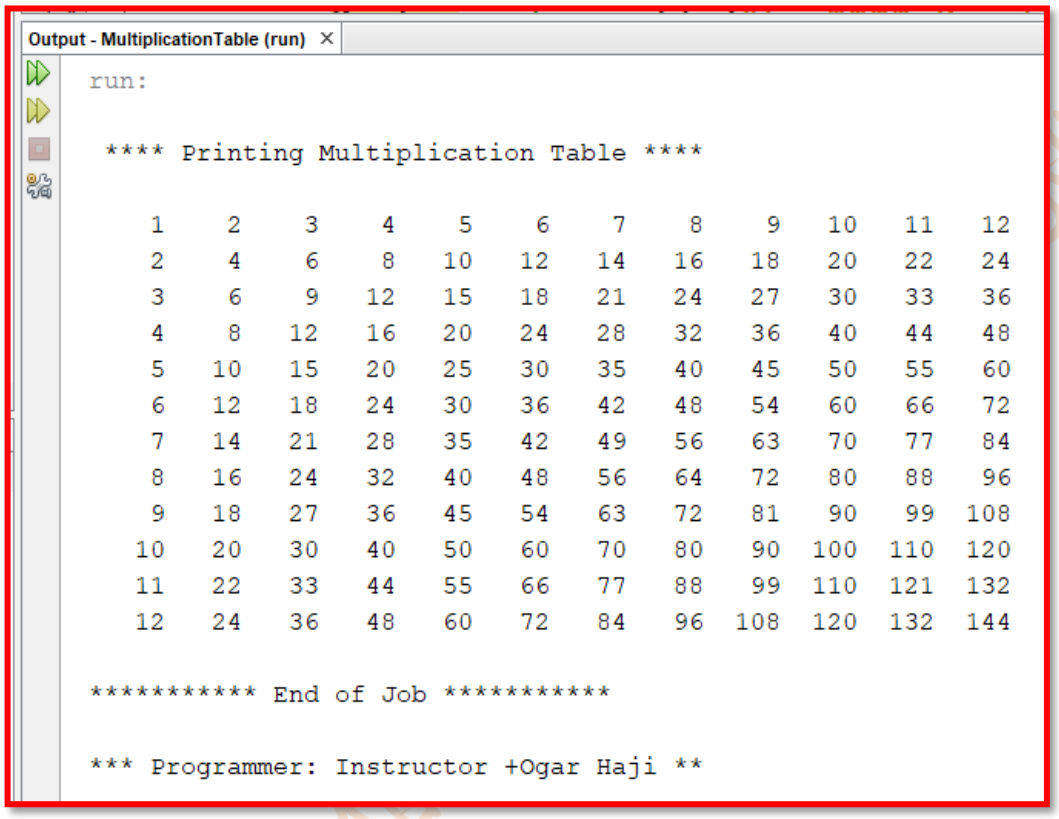
Row

The Multiplication Table of 10 by 10 will look like the following:



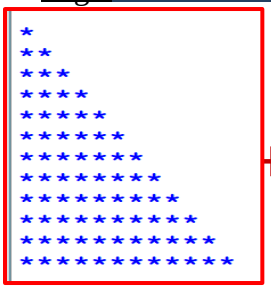
Modify This Project to Do the Following Modifications: 30%

1) **Modify the project to Add Date and Time on the top.**
2) **Modify the project to Print Multiplication Table of 12 by 12.**

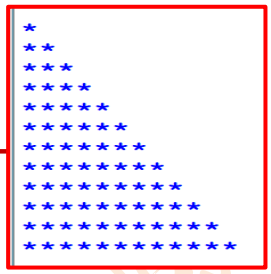


```
Output - MultiplicationTable (run)
run:

 **** Printing Multiplication Table ****

    1    2    3    4    5    6    7    8    9   10   11   12
    2    4    6    8   10   12   14   16   18   20   22   24
    3    6    9   12   15   18   21   24   27   30   33   36
    4    8   12   16   20   24   28   32   36   40   44   48
    5   10   15   20   25   30   35   40   45   50   55   60
    6   12   18   24   30   36   42   48   54   60   66   72
    7   14   21   28   35   42   49   56   63   70   77   84
    8   16   24   32   40   48   56   64   72   80   88   96
    9   18   27   36   45   54   63   72   81   90   99  108
   10   20   30   40   50   60   70   80   90  100  110  120
   11   22   33   44   55   66   77   88   99  110  121  132
   12   24   36   48   60   72   84   96  108  120  132  144

*********** End of Job ***********

*** Programmer: Instructor +Ogar Haji **
```

3) **Modify the project to Print Multiplication Table of 20 by 20.**



Nested For Statement to Print a Triangle Pattern of Stars ++(Do Lab Exercise 3) 100 Points +++



Do Lab Exercise 3

Lesson 163 + How to use Nested For statement to Print a Triangle Pattern of Stars?

Problem or Project: Design and Code in Python Language the project to Print a Triangle Pattern of Stars '*' using Nested For statement.

Do the 12 Must Steps to Design, Code & Solve a project in Python Language

You can use Nested For statements to print a Triangle Pattern of Stars. With Nested for statement here, we will have a for statement within another for statement.

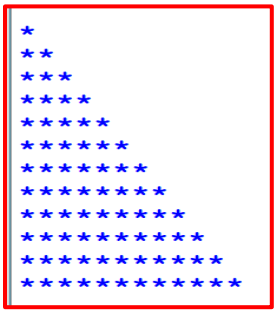
The first for statement is called the Outer for statement to print the numbers of Rows.

Then within the Outer for statement, we will add an Inner for statement to print or process the Columns.

3) The Execution will start with the Row for statement and it will set its value to 1.

4) Then it will start executing the Column for statement which will be executed 10 times for each row loop.

The **Inner For Columns** statement **will Execute 10 Times** and after the **Column count** is **10,** then it will **start again** to **execute** the **Row for statement**.



2) **Use Nested for statement** to print Triangle Pattern of Stars.

3) **Print Rows** in range **10**

```
for row in range (10):      # This is the Outer for
    # 4) Print Columns in range row + 1
    for column in range (row + 1) :  # This is the Inner for
        # 5) print Triangle Pattern of Stars with no spaces between them
        print ( '*', end = ' ' )
    # End of Second Inner for column loop
    # 6) Print a Blank Line after printing the 10th Column
    print()   # print a Blank line
# End of First Outer for row loop
```

The Output of the project: Print a Triangle Pattern of Stars

```
*********************************************************
***  Python project to print a Tringle Pattern of Stars  ***
*********************************************************

*
**
***
****
*****
******
*******
********
*********
**********
***********
************
*************
**************
***************
****************
*****************
******************
*******************
********************

*************************************************
*** End of the project + Programmer: Ogar Haji ***
*************************************************
```

+ Do Lab Exercise 3 + Do Lab Exercise 3 +

Do Lab Exercise 3

1)Type following Lab Exercise in **Python IDLE**.

2) Save the Python project as **TrianglePatternOfStars.py**

1) This program displays a triangle pattern of Stars (*).

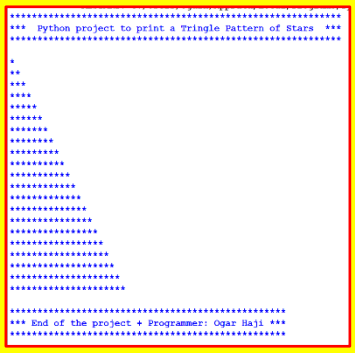
2) Print a nice Header for the project

```
print ( '*' * 60)          # print a line of 50 stars
print ('*** Python project to print a Tringle Pattern of Stars ***')
print ( '*' * 60)          # print a line of 50 stars
print ('')                 # print a blank line
```

2) Define and initialize a **Named Constant** for the number of rows to print

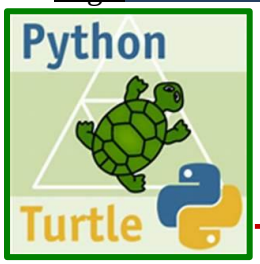
```
BASE_SIZE = 21
```

```
# 3) Use a Nested for statement to print the Triangle Pattern
# 4) Use outside for row loop to print the number of Row stars
for row in range (BASE_SIZE):  # This is the Outer for
    # 5) Use Inner for column loop to print the number of Column stars
    for column in range (row + 1):  # This is the Inner for
        # 6) Print a star and make the cursor stay on the same line
        print ('*', end = '')
    # 7) Print a Blank Line for the end of Inner for Column loop
    print ()
# 8) This is the end of the Outer for Row loop
print ('')                  # print a blank line
print ( '*' * 50)          # print a line of 50 stars
print ('*** End of the project + Programmer: Ogar Haji ***')
print ( '*' * 50)          # print a line of 50 stars
```

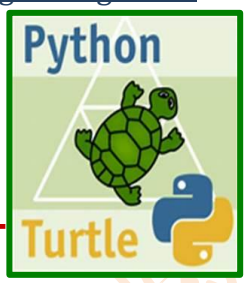


The Output of the project: Print a Triangle Pattern of Stars

```
**************************************************
***  Python project to print a Tringle Pattern of Stars  ***
**************************************************

*
**
***
****
*****
******
*******
********
*********
**********
***********
************
*************
**************
***************
****************
*****************
******************
*******************
********************

**************************************************
*** End of the project + Programmer: Ogar Haji ***
**************************************************
```



Python Language
Introduction to Python Turtle
To Draw Shapes (Review)
++(Read and Study This Lesson)+

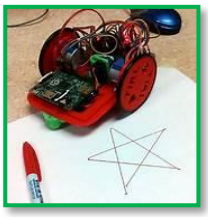


Lesson 164 + How to Draw Shapes using Python Turtle program?

Turtle is a **Python Module** or **Library** which is used to **Create** and **Design Shapes, graphics, pictures, games** and **Animations**.

Python Turtle was developed by **Wally Feurzeig, Seymour Parpet** and **Cynthina Slolomon** in **1967**.

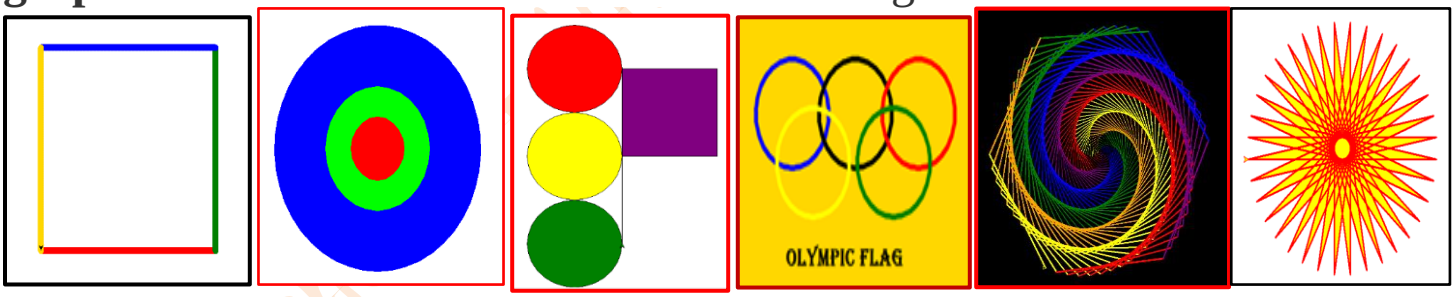
It was a part of the **original Logo programming language**.



In the **late 1960s**, **MIT professor Seymour Papert** used a **Robotic 'turtle'** to **teach programming to students**.

'turtle' is a **Python module** or **feature** like a **drawing board**, which lets us **command** a **turtle** to **draw all over the paper**.

By using the **commands** in **Python turtle**, you can **draw nice shapes**, **graphics** and **do animations** like the following:

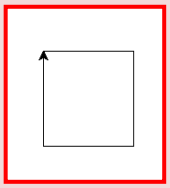


- **Python turtle** is a **Module** or **Library** which is used to **create shapes, patterns,** and **Games** on the Screen.
- **Python turtle** is initially **positioned** in the **Center** of the **Graphic Window** and the **turtle** looks like an **Arrow Head ▶.**
- **Python turtle** default heading is **0 degrees** and facing **East**.

```
import turtle
turtle.forward(100)
```

Some of Python turtle() Functions:

Turtle is a **python feature** like a **drawing board**, which **allows** you **to command** a **turtle** to **draw Lines** and **make Shapes** and **Games**.

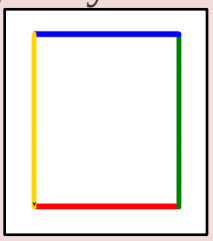


1) To **use turtle**, we must **import turtle** module first.

```
import turtle
```

2) Use **turtle.forward (100)** function to **move forward** and **draw** a **line** of **length 100 pixels**. **(96 pixels are equal to 1 inch)**

3) Use **turtle.left (90)** function to **turn the turtle left** by **90 degrees (left is Up the window screen)**.



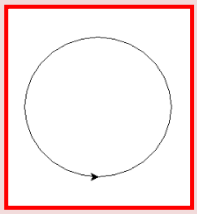
4) Use **turtle.right (90)** function to **turn the turtle Right** by **90 degrees (right is Down the window screen)**.

5) Use **turtle.setheading (90)** function to **set the turtle heading** to **90 degrees**.

6) Use **turtle.heading ()** function to **get and display the turtle current heading**.

7) Use **turtle.penup ()** function to **raise** the **turtle pen up** so **No Drawing will take effect**.

8) Use **turtle.pendown ()** function to **lower the turtle pen down and will start Drawing**.

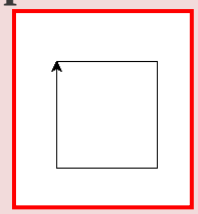


9) Use **turtle.circle (25)** function to **Draw a Circle** of **radius 25 pixels**.

10) Use **turtle.pensize (5)** function to **set** the **turtle pensize** to **size 5 pixels**. **(a little bit thicker than 1 pixel the default pensize)**

11) Use **turtle.dot ()** function to **draw a Dot (.)** on window screen.

12) Use **turtle.pencolor ('red')** function to change the **pen color** to **red color**. (**Black** color is the **default pencolor**)



13) Use **turtle.bgcolor ('yellow')** function to change the **background color** of the window screen to **yellow color**.

14) Use **turtle.setup (640,480)** function to **set the size** of the **window screen** to **Width 640 pixels** and **Height** to **480 pixels**.

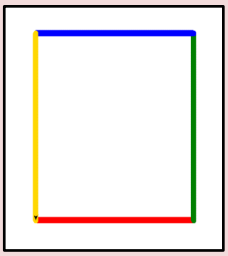
15) Use **turtle.goto (0, 100)** function to **move** the **turtle pen** to **location 0, 100** which **is x and y coordination's**.

16) Use **turtle.xcor ()** function to **return** the **x coordination** of the **current position** of **turtle pen**.

17) Use **turtle.ycor ()** function to **return** the **y coordination** of the **current position** of **turtle pen**.

18) Use **turtle.speed (0)** function to change turtle **Animation speed** of the **turtle pen drawing** to **0**. (**0** is the **Fastest speed with No Animation**). Turtle **Speed settings** can be set between **1 (Slowest)** to **10 (Fastest)**.

turtle.speed (0) is the Super Fastest and No Animation.
turtle.speed (10) is the Fast speed.
turtle.speed (6) is the Normal speed.
turtle.speed (3) is Slow speed.
turtle.speed (1) is the Slowest speed.



19) Use **turtle.hide ()** function to **hide** the **turtle arrow head**.

20) Use function **turtle.showturtle ()** to **show turtle arrow head**.

21) Use **turtle.write ('Ogar Haji')** function to **write the literal string 'Ogar Haji'** on the Window screen.

22) Use **turtle.begin_fill ()** function to **begin** to **fill** a **shape** with **color**

23) Use **turtle.fillcolor ('green')** function to **fill the shape** with **color green**.

24) Use **turtle.end_fill ()** function to **end** to **fill** a **shape** with **color**

25) Use **turtle.shape ('turtle')** to **display** the **turtle shape instead** of **Arrow shape.**

26) Use **turtle.home()** function to **move** the **turtle** to the **origin coordinates (0,0)**.

27) Use **turtle.done ()** function to Inform turtle that **we are done** and **leave the window open on screen**.

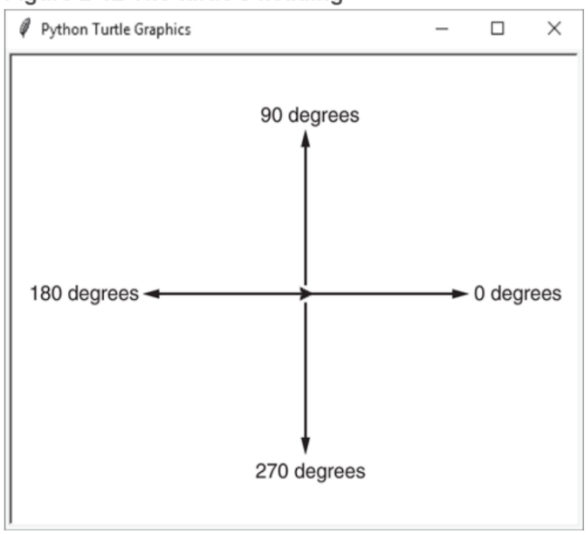
turtle.done()

The **python turtle Module or Library contains** all the **methods** and **functions** that we need to **create Shapes and Animation on screen**.

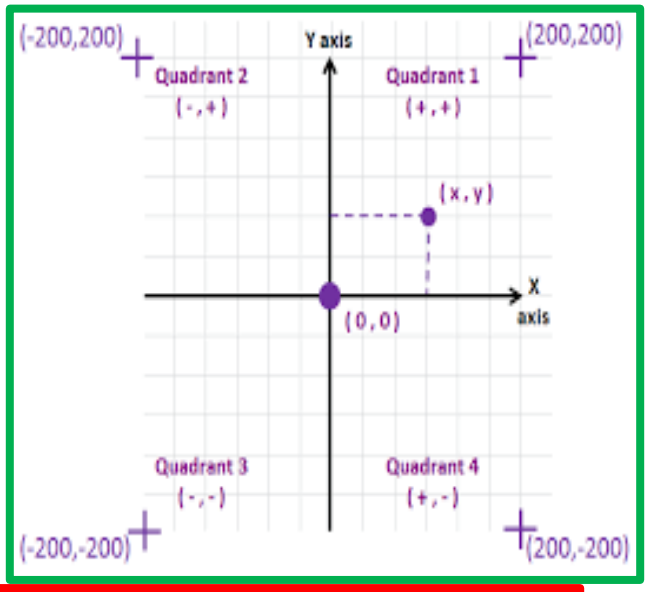
This shape is the **Turtle Heading ▶ which is Facing toward East**.

Turtle Left and Right Degrees

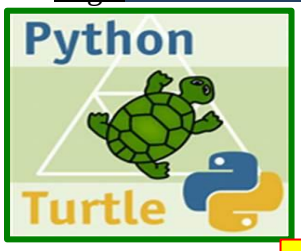
Figure 2-12 The turtle's heading



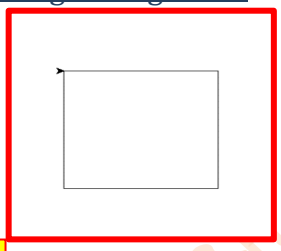
Turtle X and Y Coordinates



```
import turtle
turtle.forward(100)
```



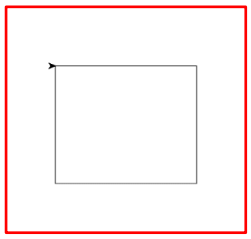
Python Turtle Language
Use for statement
To Draw the Square
+++(Do Lab Exercise 4)+++



Do Lab Exercise 4

Lesson 165 Ex: How to Draw a Square using For statement in Python Turtle?

To Draw a Square using Python turtle methods.
Then use 'for line in range (4):' statement to draw the 4 Sides or Lines of Square.



```
for line in range (4):
    turtle.forward (200)
    turtle.right (90)
```

Use a **for** statement to **Draw** the **4 Equal Lines** or **Sides** of the **Square**:

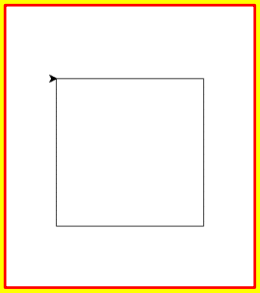
```
import turtle
for line in range (4):
    turtle.forward (200)    # Draw the First Line or Side
    turtle.right (90)       # Turn Right or Down by 90 Degrees
```

+ Do Lab Exercise 4 + Do Lab Exercise 4 +

1) Type following **Python Lab exercise,**

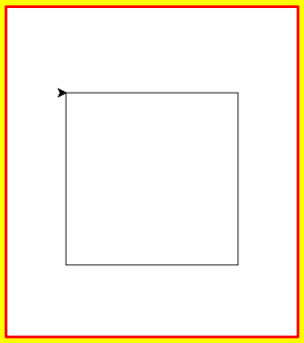
Do Lab Exercise 4

2) **Save** Python File as **TurtleUseForToDrawSquare.py.**



```
# Use a for statement to Draw a Square
# Programmer: Ogar Haji
# Date: October 22, 2020
# 1) Import turtle module library
import turtle
# 2) Draw First line and turn right or down 90 degrees
turtle.forward (200)
turtle.right (90)
```

```
# 3) Draw Second line and turn right 90 degrees
turtle.forward (200)
turtle.right (90)
# 4) Draw Third line and turn right 90 degrees
turtle.forward (200)
turtle.right (90)
# 5) Draw Fourth line and turn right 90 degrees
turtle.forward (200)
turtle.right (90)
# 6) Use turtle.done() function to inform turtle you are done and
# leave the window open
turtle.done ()
```

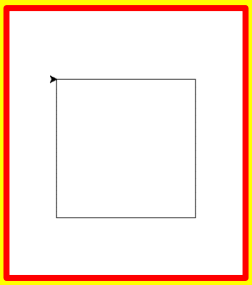


Using 'for line in range (4):' statement to loop 4 Times to Draw the 4 Lines or Sides of the Square:

It is very easy to use a for statement to draw the Square:

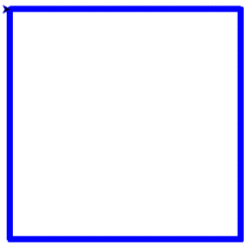
1) **Save** the Python program as **TurtleUseForToDrawSquare.py**

```
import turtle
for line in range (4):
    turtle.forward (200)
    turtle.right (90)
turtle.done()
```



Modify This Project to Do the Following Modifications: 30%

1) Change the turtle pen color to Blue color
2) Change the turtle pen size to size 8
3) Move the turtle forward by 300 pixels
4) Draw the Square
5) Upload the Modified project to Brightspace



Chapter 4 + Python Homework #4 Part A (Due This Sunday) 100 Points

Name:_____ **CIS103 Python** Programming Language + Wright College

Do the Following Chapter 4 Part A Python Homework:

Instructor: Ogar Haji

1) What does the **len()** function do?

2) Write the Python code to get **Help** on **len()** function.

3) Write the Python code to **declare** a **variable** called **'College Name'** and **assign** the **string 'Wright College'** to it.

4) Write the Python code to find the **length** or number of characters of the string variable '**college_name**'.

5) Write the Python code to **Declare 2 variables** called **test1** and **test2** and **assign** the value **'100'** to **test1** and **'96'** to **test2** and **find** the **average** of the **2 tests**?

6) Write the Python code to use **f String format** and place holders to print out the **test1** and **test2** and the **average**.

7) What are the **4 Control Structures or forms** used in Python language?

8) What is **Sequencing Structure** used in python language code, explain briefly?

9) What is **Selection Structure** used in python language code, explain briefly?

10) What is **Iteration Structure** used in python language code, explain briefly?

11) What is **Function Structure** used in python language code, explain briefly?

12) What are the **2 types of Loops** used in Python Language, explain briefly?

13) What does the **'for'** statement do, give an example?

14) What is the output after executing the following Python code?

```
for i in range (5):          # this for in range loop will loop from 0 till 4
    print (i, end = ' ')
```

A. 12345 B. 0 1 2 3 4 5 C. 0 1 2 3 4 D. 1 2 3 4 5 E. none

15) What is the output after executing the following Python code?

```
for num in range(1, 10, 2):     # remember 3 S with range (start:stop:step)
    print(num, end = ",")       # Starts with 1, increments by 2, till 10
```

A. 2,4,6,8,10 B. 2,4,6,8 C. 1,3,5,7,9, D. 1,3,7,9 E. none

16) What does the **'while'** statement do, give an example?

17) What is the output after executing the following Python code?

```
language = 'Python'
i = 0
while i < len (language):       # length of 'Python' is 6
    print (i, end = '*')        # i is = 0, so it will print 0*
    i += 1                      # Increment i by 1 which is = 1
```

A. 0*1*2*3*4*5* B. 1*2*3*4*5*6* C. 6 D. 5 E. error

18) Write the Python code to **Declare** a **variable** called **'counter'** and **initialize it** to **zero**.

19) Write the Python code to **Increment** the variable **'counter'** by **1**.

20) What is the output after executing the following Python code?

```
print (2 * 4 % 7)       # from left to right, 2 * 4 = 8, then 8 % 7 = 1 remainder
```

A. 0 B. 1 C. 4 D. 8 E. error

21) Write the Python code to **print** your **name 20 times** using **for** statement.

22) Write the Python code to **Sum** the **numbers** from **0** till **51** using **for** statement and **print** the **sum**.

23) Write the Python code to **declare a list** called **'colors_list'** and **assign 4 colors** to it.

24) Write the Python code to **Traverse** or **loop** the **colors_list** and **print** out the **colors** in the **list**.

25) What is the output after executing the following Python code?

```
number = 13
if (number % 2 == 0):   # % returns remainder if 0, means number is Even
    print ('Even')
else:
    print ("Odd")
```

A. even B. odd C. Even D. Odd E. error

26) What is the output after executing the following Python code?

```
course = 'CIS 103 Python'
print(course [::] )          # here [::] means from start to end and step is 1
```

A. CIS 103 Python B. 'CIS 103 Python' C. CIS D. Python E. nothing

27) What is the **'break'** Python keyword, explain briefly?

28) What is the **'continue'** Python keyword, explain briefly?

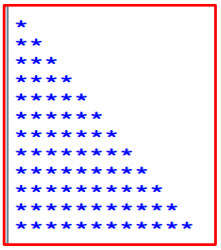
29) What is the **Nested for statement**, explain briefly?

30) Write the Python code to code the **Outer for** statement to **loop** the **rows 10 times**.

31) Write the Python code to code the **Inner for** statement to **loop** the **columns 10 times**.

32) Write the Python code to **print** the **Multiplications** of **rows and columns**.

33) Write the Python code to **print** a **Triangle pattern** of **Stars**.



34) What is the output after executing the following Python code?

```
for i in range (10, 12, 1):          # start = 10, end = 12 (not inclusive), step = 1
    if i % 2 == 0:                   # 10 % 2 is = 0 which means True
        print (f'{i} is Even')       # prints 10 is Even, and 11 is Odd
    else:
        print (f'{i} is Odd')
```

A. 10 is Even
11 is Odd

B. 10 is Odd
11 is Even

C. 10 is Even
11 is Even

D. 11 is Odd

E. error

35) What is the output after executing the following Python code?

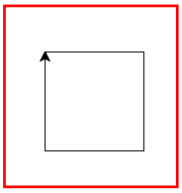
```
a = 'Apple';  b = 'Orange'
x, y = b, a      # swap and assign b 'Orange' to x, assign 'Apple' to y
print (x, y, sep= "::")
```

A. Apple:Orange B. Apple::Orange C. Orange:Apple D. Orange::Apple

36) What is **turtle module**, explain briefly?

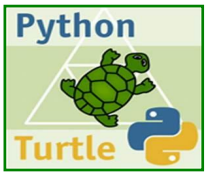
37) Write the Python turtle command to **include or import turtle module** so you can work with turtle commands in Python.

38) Write the Python turtle command to **Move** the **turtle forward by 300 pixels**.

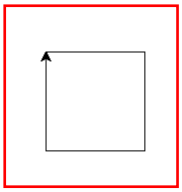


39) Write the Python turtle command to **Turn** the **turtle Left by 90 degrees**.

40) Write the Python turtle command to **Turn** the **turtle Right by 120 degrees**.

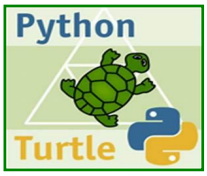


41) Write the Python turtle code to **Draw** a **Square** of **length of 350 pixels** using **for** statement.



42) Write the Python turtle code to **Change** the **turtle pen color** to **'red' color**.

43) Write the Python turtle code to **Change** the **turtle pen size** to **size 7**.



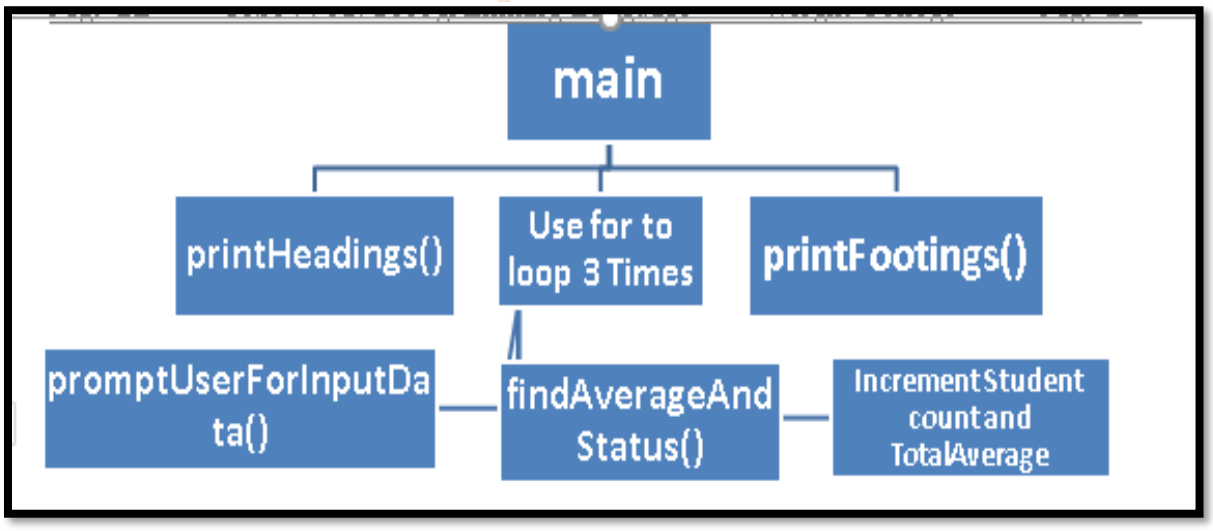
44) Write the Python turtle code to **inform the turtle that you are done and leave the shape drawn on the screen**.

45) What is the output after executing the following Python code?

```
for i in range (11):        # you can use any variable you want for 'i'
    if i == 0 :             # i = 0 and 0 == 0, this is True
        print ('Breaking out of for, i =', i ) # prints this and break out
        break
    else:
        print ('Still working in for.')
```

A. Breaking out of for, i = 0 B. Still working in for. C. True D. False

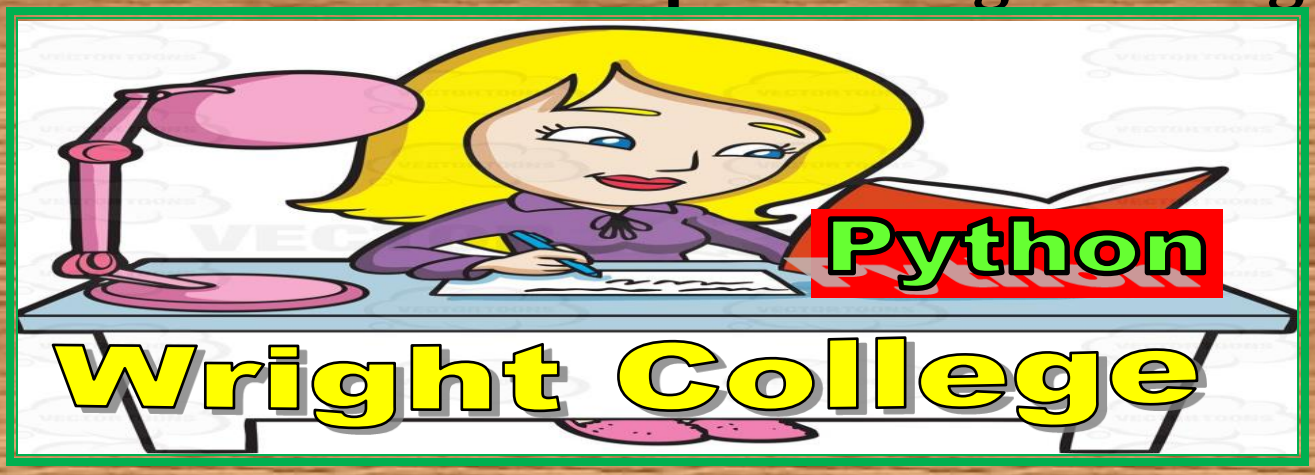
46) Write the Python code to **Design a Structure programming using Functions** and **call** the following **functions** from **main() method**.



Wright College + Chapter 4B

Loops and Iteration: Using 'for' and 'while' Statements, Math functions and Random Numbers

CIS 103 Python Programming Language + Introduction to Computer Programming



'Hands-On' Mastering Computer Logic, Design and Programming Using Python Language

Written By:

OGAR HAJI

Master's Degree in Computer Science

DePaul University ✚ Chicago, Illinois + USA

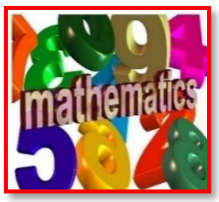
Date Published: February 5, 2023

Chapter 4 + Part B

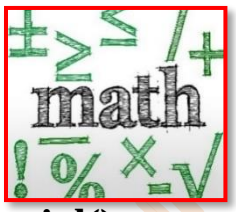
Loops and Iteration:
Using 'for' and 'while' Statements,
Math functions and Random Numbers
Pick 3 Lotto Project,

You will learn the following in Chapter 4 Part B:

- Making Decision using Single if statement
- Use if --else statement to check if a condition is True or False
- Calculate the Average of 3 Tests and using if statement to check if student is 'Passing' or 'Failing'
- Using if – else statement to find Final Grade for the Average
- Calculate the Average of 3 Tests and Using if statement
- Using if statement to check if a condition is true or false
- Calculate Overtime of GrossPay exercise
- Using the Conditional And Operator &&
- Using the Conditional Or Operator ||
- Using the Ternary Conditional Operator (? :)
- Using the Not ! operator to Negate the Result (Opposite)
- Do Chapter 4 Instructor Handout Homework #4
- Do Lab Assignment 4 + Hotel Room Charges



Using the math. Module in Python: Important math. Methods()

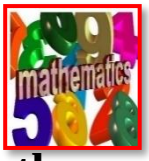


+++(Read and Study This Lesson)+++

Lesson 170 Ex + How to use math module in Python to Use math.factorial(), math.pie(), math.round(), math.ceil() and math.floor() methods?

The **math** module is used to **access mathematical methods** in Python. All methods of this functions are used for integer or real type objects.

To use the '**math**' module in Python, you should **import** the **math module**.



import math

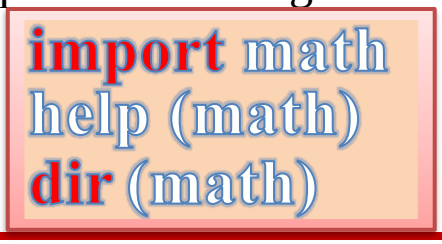
Getting help from Python

There is documentation built into Python. To get **help** on the **math** module, go to the Python shell and type the following two lines:

>>> import math

>>> help (math)

>>> dir (math)




```
IDLE Shell 3.11.0
File  Edit  Shell  Debug  Options  Window  Help
    Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 bi
    t (AMD64)] on win32
    Type "help", "copyright", "credits" or "license()" for more i
    nformation.
>>> import math
>>> dir (math)
    ['__doc__', '__loader__', '__name__', '__package__', '__spec_
    _', 'acos', 'acosh', 'asin', 'asinh', 'atan', 'atan2', 'atanh
    ', 'cbrt', 'ceil', 'comb', 'copysign', 'cos', 'cosh', 'degree
    s', 'dist', 'e', 'erf', 'erfc', 'exp', 'exp2', 'expm1', 'fabs
    ', 'factorial', 'floor', 'fmod', 'frexp', 'fsum', 'gamma', 'g
    cd', 'hypot', 'inf', 'isclose', 'isfinite', 'isinf', 'isnan',
    'isqrt', 'lcm', 'ldexp', 'lgamma', 'log', 'log10', 'log1p', '
    log2', 'modf', 'nan', 'nextafter', 'perm', 'pi', 'pow', 'prod
    ', 'radians', 'remainder', 'sin', 'sinh', 'sqrt', 'tan', 'tan
    h', 'tau', 'trunc', 'ulp']
>>>
```

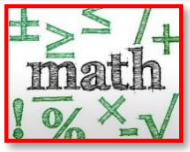
import math

To Find the Factorial of 5, use the **.factorial()** method of **math** function

print (math.factorial (5)) # will print 120

math.factorial(5)

```
>>> import math
>>> print (math.factorial (5) )
120
```



The **math module contains methods() for performing basic numeric operations** such as find the **power of a number**, find **Maximum** and **Minimum** of 2 Numbers, **exponential, logarithm, square root,** and **trigonometric methods**.

The **math Module** is part of Python **API (Application Programming Interface).**

math.pi is a **Constant** that is **assigned** the **value 3.141592653589793**.

print ('The Value of math.pi = ' , math.pi)

math.pi

```
>>> print (math.pi)
3.141592653589793
```

```
The Value of Math.PI = 3.141592653589793
BUILD SUCCESSFUL (total time: 0 seconds)
```

Python math module provides several **methods** to work on **math calculations** like **min(), max(), math.avg(), math.sin(), math.cos(), math.tan(), math.round(), math.ceil(), math.floor(), math.abs()** and more.

1) **max(*x*,*y*): max()** function returns maximum number

max (x, y)

The max(*x*,*y*) method can be used to **find the highest** value of *x* and *y*:

max(5, 10, 7) # returns **10** as the **Maximum Number**

min (x, y)

2) **min(*x*,*y*) min()** function returns minimum number

The min(*x*,*y*) method can be used to **find the lowest** value of of ***x*** and ***y***:

min(5, 10, 7) # returns **5** as the **Minimum Number**

math.fabs(n)

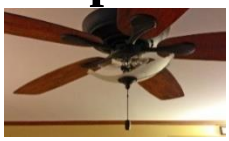
3) **math.fabs(*n*)** method

The math.fabs(n) method returns the **absolute (positive)** value of ***n***:

math.fabs(-4.7) # returns the **Positive** Number **+4.7**

4) **math.ceil (n)** method returns the **value** of **n Rounded Up** to its **nearest integer**.

math.ceil(n)



math.ceil (7.23) method returns the number **8.0**

5) **math.floor (n) method** returns the value of **n Rounded Down** to its **nearest integer**.

math.floor(n)

math.floor (7.23) method returns the number **7.0**

```
import math
#5) math.ceil (n) method returns the value of n Rounded Up to its nearest integer.
print (f'The math.ceil of 7.23 is {math.ceil (7.23)}')

#6) math.floor (n) method returns the value of n Rounded Down to its nearest integer.
print (f'The math.floor of 7.23 is {math.floor (7.23)}')
```

```
The math.ceil of 7.23 is 8
The math.floor of 7.23 is 7
```

6) **math.pow (x, y) method** returns the value of **x raised** to **power** of **y**.

math.pow (2, 3) returns the value **8.0 means 2 * 2 * 2 = 8**

math.pow(n, p)

7) **round(n) function** returns the number **n Rounded** to its **nearest Integer**.

math.round(n)

round (7.23) function returns the value **7** which is **rounded Down**
round (7.53) function returns the value **8** which is **rounded Up**

8) **math.sqrt (n)** method returns the **Square Root** of the number **n**.
math.sqrt (9) method returns the number **3.0**

math.sqrt(n)

import math

9) **math.remainder (x, y)** method returns remainder after dividing x by y.
remainder (7, 3) will return **1.0**

math.remainder(x,y)

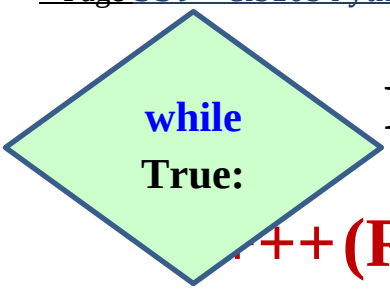
10) **factorial (x) function** returns factorial of x. where x ≥ 0
factorial (5) will return **120 which is (5 * 4 * 3 * 2 * 1 = 120)**

math.factorial (n)

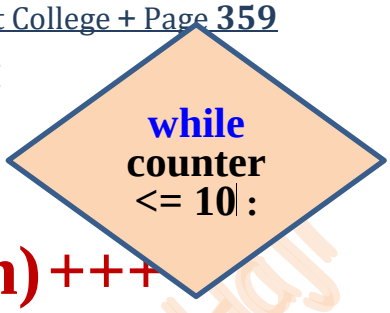
Please, Read, Study and Practice the Lessons in the Python Handout

factorial (x) is a **function while**

math.remainder (x, y) is a **method** which has the **dot .** after **math** as in **math.remainder()**



Using 'while' statement to: Print out Numbers, Power 2 and Power 3 Backwards



+++ (Read and Study This Lesson) +++

Lesson 171 + How to use While statement to Find Power2 and Power3 of Numbers Backwards From 10 down to 0?

There are 2 Types of Loops:

1) **Count-Controlled Loops:** Use '**for**' statement for **Count-Controlled Loops** when you **know** in **advance** the **number** of **times a loop will occur** or **happen**. For example:

```
for counter in range (10):
    power2 = pow (counter, 2)
```

2) **Condition-Controlled Loops:** Use '**while**' statement for **Condition-Controlled Loops** when you **test a condition**, and the **condition** is **True**.

Problem or Project: Design and Code in Python Language the project to Find Power 2, Power 3 of a Number from 10 till 0 using while statement

You can use pow (n,2) to find the Power 2 of a number n.

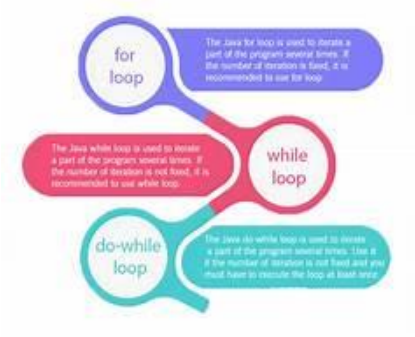
pow (n, 2)

```
counter = 0  # Initialize counter to 0
# 1) Use 'while' statement to loop 10 Times from 0 till 10
print ('Number\t Power 2 \t Power 3')
while counter <= 10 :
    power2 = pow (counter, 2)
    power3 = pow (counter, 3)
    print ('{0} \t {1} \t\t {2}'.format (counter, power2, power3) )
    counter +=1     # Increment the counter by 1
```

Indent

Exponent Code	Multiplication	
2^3	2·2·2	(= 8)
3^4	3·3·3·3	(= 81)
5^3	5·5·5	(= 125)
10^3	10·10·10	(= 1,000)

```
*** Printing the Number, Power 2 and Power 3 ***

Number    Power 2           Power 3
0         0                 0
1         1                 1
2         4                 8
3         9                 27
4         16                64
5         25                125
6         36                216
7         49                343
8         64                512
9         81                729
10        100               1000

+++++++++ End of the Project ++++++++
+++++++ Programmer: Ogar Haji +++++++
```



1)Code Python Code,
2) Save File as PowerOfNumbers.py.

```
# You can use pow (n,2) to find the Power 2 of a number n.
# 1) Initialize the counter to 0
counter = 0     # Initialize counter to 0
# 2) Print the headings of the columns
print ('\n*** Printing the Number, Power 2 a
print ('Number\t Power 2 \t Power 3')
# 3) Use 'while' statement to loop 10 Times from 0 till 10
while counter <= 10:
    # 4) Find out the power 2 of the number using pow() function
    power2 =  pow (counter, 2)
    # 5) Find out the power 3 of the number using pow() function
    power3 =  pow (counter, 3)
    # 6) Print out the Number, Power2 and power3
    print ('{0} \t {1} \t\t {2} '.format (counter, power2, power3) )
```

*** Printing the Number, Power 2 and Power 3 ***

Number Power 2 Power 3
0 0 0
1 1 1
2 4 8
3 9 27
4 16 64
5 25 125
6 36 216
7 49 343
8 64 512
9 81 729
10 100 1000

+++++++++ End of the Project ++++++++
+++++++ Programmer: Ogar Haji +++++++

```
# or use f string format with variable names in place holders {counter}
    print (f'{counter} \t {power2} \t\t {power3} ' )

        # 7) Increment the counter by 1
        counter += 1    # Increment the counter by 1

# 8) Print out the End of the project
print ('\n+++++++++ End of the Project ++++++++')
print ('+++++++ Programmer: Ogar Haji +++++++')
```

```
*** Printing the Number, Power 2 and Power 3 ***

Number    Power 2          Power 3
0         0                0
1         1                1
2         4                8
3         9                27
4         16               64
5         25               125
6         36               216
7         49               343
8         64               512
9         81               729
10        100              1000

+++++++++ End of the Project ++++++++
+++++++ Programmer: Ogar Haji +++++++
```

```
print ( '{0} \t {1} \t\t {2} '.format
    (counter, power2, power3) )
# or
print (f'{counter} \t {power2} \t\t {power3} ')
```

Random Numbers 7 3 1

Types of Random Number Generators: random_random() method

(Read and Study this Lesson)

Random Numbers 0 4 9

Lesson 172 Ex: Types of Random Numbers Generators in Python?

There are many Methods used in Python language to Generate Numbers. First, you should import the random module in order to Generate Random Numbers.

To Import random Module:

import random

The following are some of random methods used in Python language:

1) **random** method of **random.random ()** method: **Returns a Floating Number** which is **Greater than 0 and Less than 1** like **0.945623148**.

```
File  Edit  Format  Run  Options  Window  Help
1 import random
2 random_number = random.random()
3 print (random_number)
4
```

Run the project and the output will look like the following:

```
= RESTART: C:/Users/ogar
n311/Homework 4 Demo.py
0.40130603404468 11
```

2) **randint** method of **random.randint (0,9)** method: **Returns an Integer Number** which is **from 0 up to 9** like **8**.

```
File  Edit  Format  Run  Options  Window  Help
1 import random
2 for i in range (20):
3     random_number = random.randint(0,9)
4     print (random_number)
5
```

Run the project and the output will look like the following:

```
= RESTART: C:/Users/ogar
n311/Homework 4 Demo.py
8
```

3) **randrange** method of **random.randrange (25)** method: **Returns an Integer Number** which is **from 0 up to 24** like 17.

```
File  Edit  Format  Run  Options  Window  Help
1 import random
2 for i in range (51):
3     random_number = random.randrange(25)
4     print (random_number)
5
```

Run the project and the output will look like the following:

```
= RESTART: C:/Users/ogarh/
n311/Homework 4 Demo.py
1
21
23
8
12
```

4) **choice** method of **random.choice (list)** method: **Returns a Random item from a list populated with items** like color 'Red'.

```
File  Edit  Format  Run  Options  Window  Help
1 import random
2 colors_list = ['Red', 'Green', 'Blue', 'Gold']
3
4 for i in range (10):
5     random_color = random.choice(colors_list)
6     print (random_color)
7
```

Run the project and the output will look like the following:

```
= RESTART: C:/Users/ogarh
n311/Homework 4 Demo.py
Green
Blue
Blue
Red
```

5) **shuffle** method of **random.shuffle (list)** method: **Returns a Random List with the items rearranged in different order in a list..**

```
File  Edit  Format  Run  Options  Window  Help
1 import random
2 colors_list = ['Red', 'Green', 'Blue', 'Gold']
3
4 for i in range (5):
5     random.shuffle(colors_list)
6     print (colors_list)
7
```

Run the project and the output will look like the following:

```
= RESTART: C:/Users/ogarh/AppDat
n311/Homework 4 Demo.py
['Red', 'Green', 'Gold', 'Blue']
['Green', 'Gold', 'Red', 'Blue']
['Green', 'Red', 'Gold', 'Blue']
```

Random Numbers
7
3
1

Random Number Generator: Pick 3 Lotto
random_number = random.randint (0,9)
+++ (Do Lab Exercise 5) +++

Random Numbers
0
4
9

Do Lab Exercise 5

Lesson 172 Ex: How to Generate Pick 3 Lotto Random Numbers in Python ?

Problem or Project: Design and Code in Python Language the project to use random.randint (0,9) module to Generate Random Numbers from 0 to 9 for Pick 3 Lotto.

To Generate Random Numbers from 0 thru 9 use **random.randint (0,9) as shown below:**

 # 1) Import random module

```
import random
```

 # 2) Pick 3 Lotto Today Winner Number 157

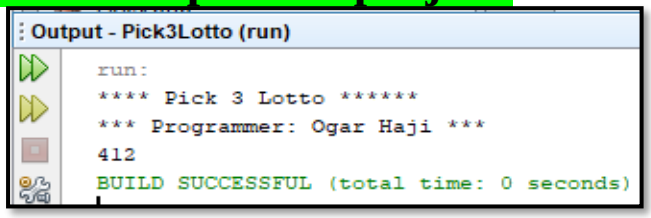
```
for i in range (3):
    random_number = random.randint (0,9)   # random numbers from 0-9
    print (random_number, end = '')
    # End of for statement
```

n = random.randint (0, 9)

This is an Explanation of What we will Do in this project:

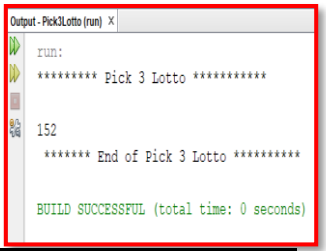
 # 1) Add Comments about the project on the top of the project.

```
'''     Write your Comments here
        Date: Sunday, February 05, 2023
        Programmer: Ogar Haji
'''
```



 # 2) Import the Classes needed in the project in the package section

```
# Import the Classes needed in this project
import random
```



3) Use random.randint() method to generate random numbers
4) Use for statement to print to output console Pick 3 Lotto

```
for i in range (3):
    random_number = random.randint(0,9)  #random numbers from 0-9
    print (random_number, end = '')
## End of for statement
# 5) Print out a Blank line
print ()
```

n = random.randint (0, 9)

+ Do Lab Exercise 5 + Do Lab Exercise 5 +

1) Ex. Type the following Python Code and

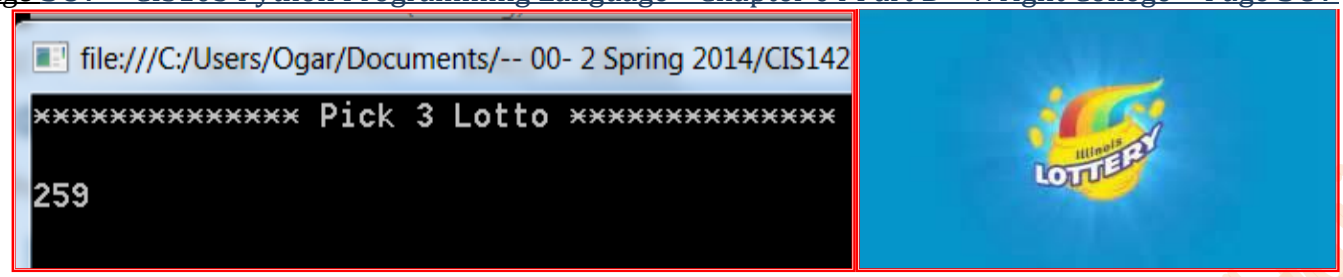
2) Save Python project as Pick3Lotto.py

Do Lab Exercise 5

```
# 1) Add Comments about the project on the top of the project.
'''
Write your Comments here
Date: Saturday, September 11, 2020
Programmer: Ogar Haji
'''
import random

# 2) Print the Headings of the Project
print ('********** Pick 3 Lotto **********')

# 3) Use random.randint(0,9) method to generate random numbers from 0-9
# 4) Use for statement to print out to output console Pick 3 Lotto
for i in range (3):
    random_number = random.randint (0,9)   #random numbers from 0-9
    print (random_number, end = '')
# End of for statement

# 5) Print out a Blank line
print ()
```

Do Lab Exercise 5

```
********** Pick 3 Lotto ******
457
```



Modify This Project to Do the Following Modifications: 30%

Modify the Python Project to do the following:

1) **Modify the Program to Pick 4 Lotto. (like 9751).**
2) **Insert Computer Date in the Header at the top of the project.**
3) **Print a nice header consisting of many lines.**
4) **Print a nice footer consisting of many lines and include your full name.**
5) Use **while** statement to **prompt** the **User** if he/she **wants** to **continue Running** the **program again.** .

Note: Always **Upload** to **Brightspace the Modified Python Project:**
1) The **Microsoft Word Document** of the **Modified Python Project Code along with the Python Output Screen shots.**

Please, Read, Study and Practice the Lessons in the Python Handout

Chapter 4 + Quiz 2 + Test Your Python Language Knowledge:

---

1. What is the output after executing the following Python code?

```
for i in range (5):          # Outer nested for 'i' loop from 0 to 4
    for j in range (5):      # Inner nested for 'j' loop from 0 to 4
        if j == 3:
            break            # break out of the Inner for loop when j == 3
print (i * j)                # after exiting for loop, i = 4, j = 3
```

A. 25 B. 20 C. 16 D. 12 E. error

---

2. What is the output after executing the following Python code?

```
for i in range (5):          # Outer nested for loop from 0 to 4
    for j in range (5):      # Inner nested for loop from 0 to 4
        if j == 3:
            continue         # continue will continue the Inner for loop
print (i * j)                # after exiting for loop, i = 4, j = 4
```

A. 25 B. 20 C. 16 D. 12 E. error

---

3. What is the output after executing the following Python code?

```
greeting = 'Good Morning'
for ch in greeting:
      if ch == 'o':          # ch is equal to G, so else will execute
            break
            print (ch)
       else:
            print ('Good Night')
```

A. G B. Go C. Good Morning D. Good Night E. error

---

4. What is the output after executing the following Python code?

```
sum = 0
for i in range (1, 5):
    sum  +=  i       # 0+1+2+3+4 = 10 sum = 10
print (sum)
```

A. 0 B. 1 C. 10 D. 15 E. error

---

5. What is the output after executing the following Python code?

```
for i in range (1):          # i is = 0
    for j in range (1):      # j is = 0
        print (i, j)         # prints 0  0
```

A. 0 1 B. 0 0 C. 1 0 D. 1 1 E. error

---

6. What is the output after executing the following Python code?

```
val = 8
while val > 0:       # result is True because 8 is greater than 0
    val = val - 2    # val = 8 - 2 = 6
    if val <= 5:     # False for val = 6, then another while and val = 4 True
                     # then it prints 4 and exit the while loop and hi will print
        print (val, end="")
        break
print ('hi')
```

A. 8hi B. 4hi C. 2hi D. hi E. error

---

7. What is the output after executing the following Python code?

```
x = 'python' ;  i = 0
while i < len (x):    # length of x = 6, so this is True
   i += 1             # i = 0 + 1 = 1,
   pass
print ('Value of i =', i)
```

A. Value of i = 0 B. Value of i =1 C. Value of i = 5 D. Value of i = 6 E. error

---

8. What is the output after executing the following Python code?

```
str = 'Python language is easy to learn'
print ('easy' in str)       # The 'in' operator is used to find if 'easy' is in str
```

A. easy B. Python C. True D. False E. error

---

9. What is the output after executing the following Python code?

```
print (7 % 2 ** 3)          # Evaluate the ** first, 2 ** 3 = 8, then 7 % 8 = 7
```

A. 2 B. 3 C. 7 D. 8 E. error

---

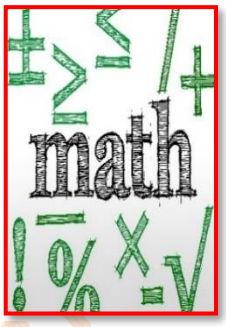
10. To Change the shape of the turtle to 'turtle' Shape, type the following:

```
turtle.shape ('turtle')
```

(True/False)

Answers Are Found at The End of This Chapter 04.

Using random.randint() method in Python: To Generate Random Integer Numbers From a Range of Numbers



+++ (Read and Study This Lesson) +++

Lesson 173 Ex: How to use random.randint() method to Generate Random Numbers in Python?

Problem or Project: Design and Code in Python Language the project to use random.randint() method to Generate Random Numbers from a Range of Numbers.

Do the following 12 Must Steps to Design, Code and Solve a problem using Python Language.

Do Steps 1 thru 7 in your Note Book or on Paper.

Step 1) Purpose of the Program: State what Program will do: (5 Points)

a) This Program will Generate a Random Number from a Range of Numbers supplied by user. (5 Points)

Step 2) Input: State what the Input will look like:

a) The user will be prompted to enter the Starting and Ending Numbers in this project. (5 Points)

Step 3) Processing and Calculation: What Processing and Calculation to be done in this Project:

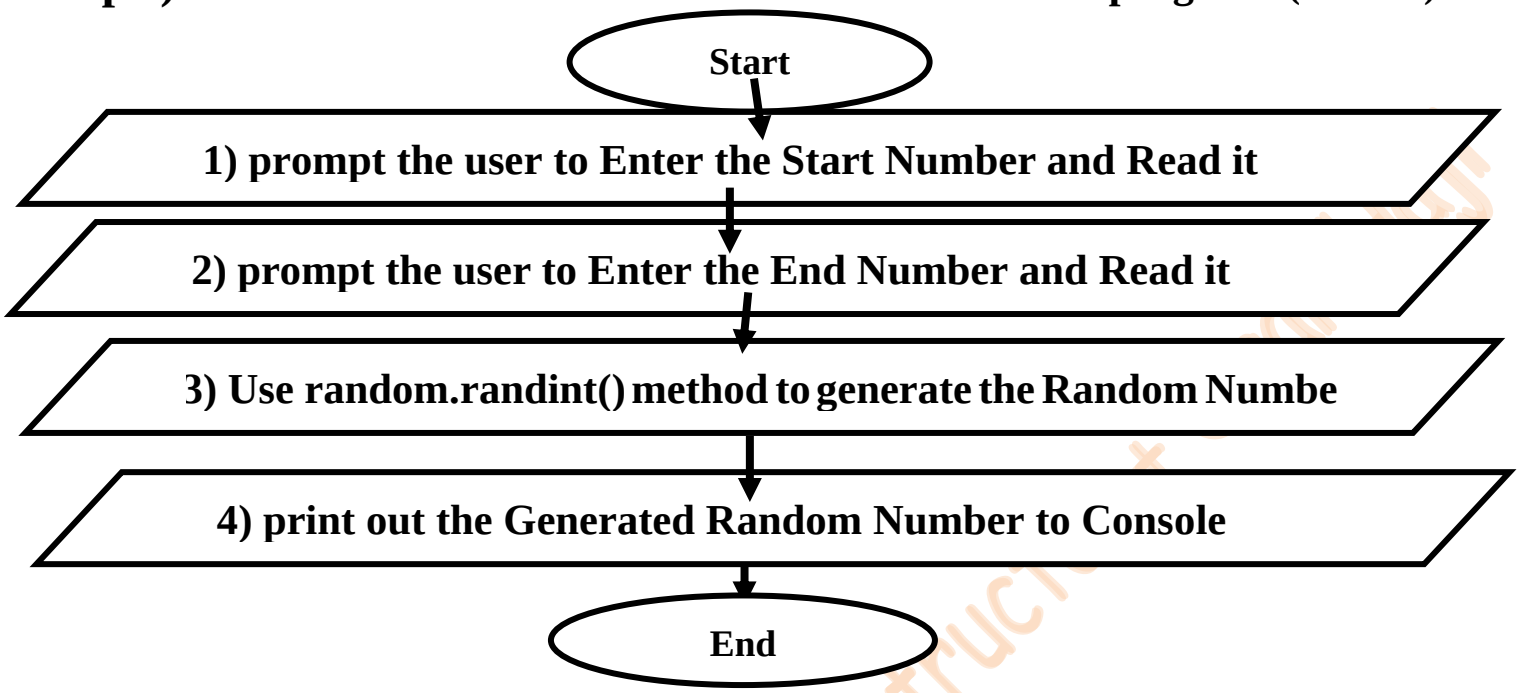
a) Use random.randint() to generate the Random Number: (5 Points)

Step 4) Output: You should know how output should look like: (5 Points)

a) This **Project** will **print out the Generated Number** to **output Console**.

```
Output - GenerateRandomNumberFromARange (run)
run:
Enter the Starting Number of the range: 1
Enter the Ending Number of the range: 10
The Random Number Generated is: 8
BUILD SUCCESSFUL (total time: 13 seconds)
```

Step 5) Flowchart: Draw a Flowchart for HelloWorld program. (5 Points)



Step 6) PseudoCode: write a PseudoCode for the Program. (5 Points)

a) Prompt user to Enter the Start Number and Read and store it
b) Prompt user to Enter the End Number and Read and store it
c) Use random,randint() method to Generate the Number and print it

5) Use the **random.randint()** method to generate the Random Number

6) Use the random.randint() method to generate the Random Number
generated_number = random.randint (start_number, end_number)

Step 7) Code the Program in Python using NIDLE IDE by referencing the Flowchart or Pseudocode you designed above.

This is an Explanation of What we will Do in this project:

This project will generate a Random Number between a Range of numbers.

1) Import the Class Random
import random

2) Prompt the user to enter the Starting Number and read it
start_number = input ('Enter the Starting Number of the range: ')

3) Convert the string number into int using int() function
start_number = int (start_number)

4) Prompt the user to enter the Ending Number and read it
end_number = input ('Enter the Ending Number of the range: ')

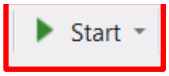
5) Convert the string number into int using int() function
end_number = int (end_number)

for i in range (10):

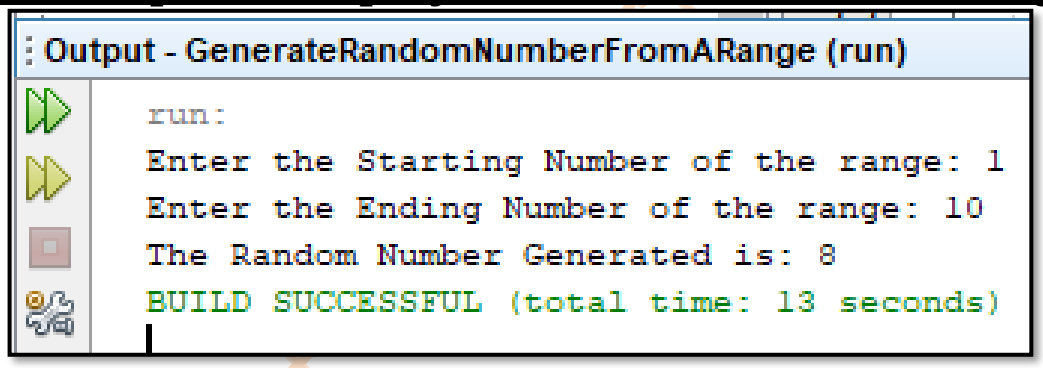
    # 6) Use the random.randint() method to generate the Random Number
    generated_number = random.randint (start_number, end_number)

    # 7) Print out the Random Number Generated
    print ('The Random Number Generated is: ', generated_number)

a) Click **Save All**  button to Save All Files

Step 8) Click **Start**  button to **Start Running** the program 

The Output of the project will look like the following:



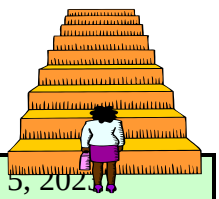
If any Syntax Errors Found Do Next Step 9:

Step 9) Debug the Program: Debug or Correct any Syntax Errors until you have a clean Compiled program. (5 Points) (Clean compiled program means No Errors in the program).

Step 10) Test the Program: Test the Program with Test Data. (5 Points)
Repeat Step 10) Test the program many Times and Test the Program again and again until All conditions are tested:

Step 11) Documentation (5 Points): You have to add more comments to the Project (like Comments about the Purpose of the Project, Your Name and the Date the Project was written).

‘’’



****

Purpose of the Project:

a) This Project Generates a Random Number using random() method

b) Project Name: Generate a Random Number

c) Date: Sunday, February 05, 2023

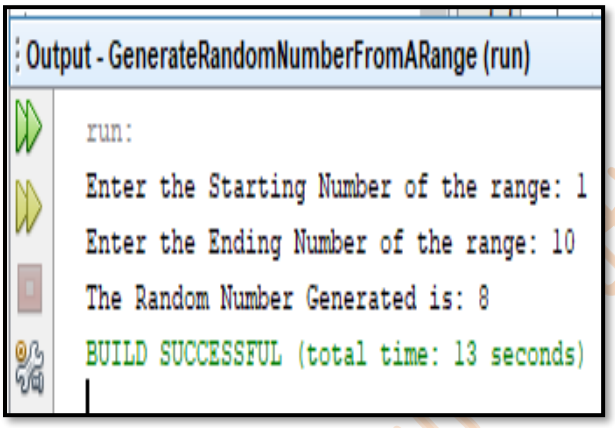
b) Programmer: Instructor – Ogar Haji

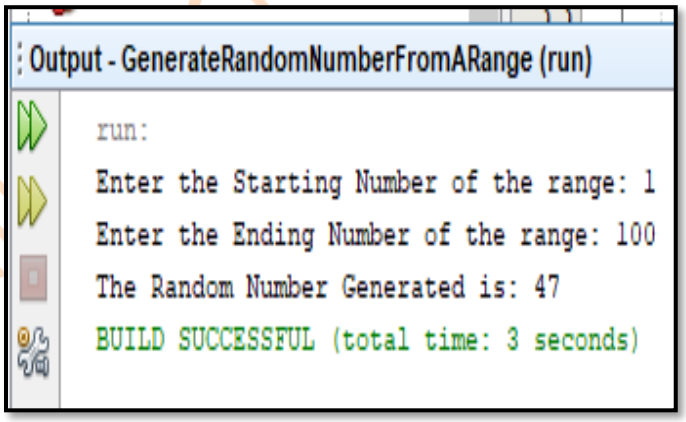
****

‘’’

Step 12) Print a Copy of Python Code along with screen printout of the Running program. Submit to your Instructor the Print Copy and the screen Printout (Snaps) along with the following: (Which you did on Paper)

The Input and Output of this Python project looks like the following:





This is an Explanation of What we will Do in this project:

 # 1) Add Comments about the project on the top of the project.

''' 1) Add Comments about the project on the top of the project.

This project will generate a Random Number between a Range of numbers. The program will prompt the user to enter the Starting Number and then enter the Ending Number. The Random Generated Number will be between the Starting Number and the Ending Number.

Program Name: GenerateRandomNumberFromRange

Programmer: Ogar Haji (Python Instructor)

```
Enter the Starting Number of the range: 10
Enter the Ending Number of the range: 30
The Random Number Generated is:  16
The Random Number Generated is:  20
The Random Number Generated is:  14
The Random Number Generated is:  20
The Random Number Generated is:  25
```

Date Written: 05/11/2020

'''

1) Import the Class random module
import random

2) Prompt the user to enter the Starting Number and read it
start_number = input ('Enter the Starting Number of the range: ')

3) Convert the string number into int using int() function
start_number = int (start_number)

4) Prompt the user to enter the Ending Number and read it
end_number = input ('Enter the Ending Number of the range: ')

5) Convert the string number into int using int() function
end_number = int (end_number)

```
Enter the Starting Number of the range: 10
Enter the Ending Number of the range: 30
The Random Number Generated is:  16
The Random Number Generated is:  20
```

for i in range (10):

    # 6) Use the random.randint() method to generate the Random Number
    generated_number = random.randint (start_number, end_number)

    # 7) Print out the Random Number Generated
    print ('The Random Number Generated is: ', generated_number)

1) Ex. Type the following **Python Code in IDLE**

2) Save Python project as **GenerateRandomNumberFromRange.py**

'''

1) Add Comments about the project on the top of the project.
This project will generate a Random Number between a Range of numbers. The program will prompt the user to enter the Starting Number and then enter the Ending Number. The Random Generated Number will be between the Starting Number and the Ending Number.
Program Name: GenerateRandomNumberFromRange

Programmer: Ogar Haji (Python Instructor)
Date Written: 05/11/2020
'''

```
Enter the Starting Number of the range: 10
Enter the Ending Number of the range: 30
The Random Number Generated is:  16
The Random Number Generated is:  20
The Random Number Generated is:  14
The Random Number Generated is:  20
The Random Number Generated is:  25
```

1) Import the random module
import random

2) Prompt the user to enter the Starting Number and read it
start_number = input ('Enter the Starting Number of the range: ')

3) Convert the string number into int using int() function
start_number = int (start_number)

4) Prompt the user to enter the Ending Number and read it
end_number = input ('Enter the Ending Number of the range: ')

Enter the Starting Number of the range: 10
Enter the Ending Number of the range: 30
The Random Number Generated is: 16
The Random Number Generated is: 20
The Random Number Generated is: 14
The Random Number Generated is: 20
The Random Number Generated is: 25

5) Convert the string number into int using **int()** function
end_number = int (end_number)

for i in range (10):

    # 6) Use the **random.randint()** method to generate the Random Number
    generated_number = random.randint (start_number, end_number)

    # 7) Print out the Random Number Generated
    print ('The Random Number Generated is: ', generated_number)

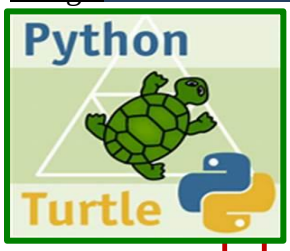
The Input and Output of this Python project looks like the following:

```
Enter the Starting Number of the range: 10
Enter the Ending Number of the range: 30
The Random Number Generated is:  16
The Random Number Generated is:  20
The Random Number Generated is:  14
The Random Number Generated is:  20
The Random Number Generated is:  25
The Random Number Generated is:  13
The Random Number Generated is:  22
The Random Number Generated is:  10
The Random Number Generated is:  10
The Random Number Generated is:  15
```

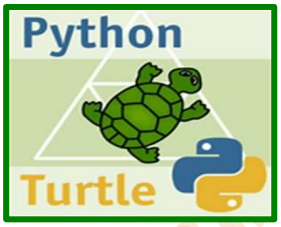
Modify This Project to Do the Following Modifications: 30%

1) **Insert Computer Date at the top**.
2) **Use 'for' statement to Generate 7 Random Numbers**.
3) **Use While statement to prompt the User 'Do You Want to Continue Enter (Y/N) ?'.**

Note: Always **Upload** to **Brightspace the Modified Python Project:**
1) The **Microsoft Word Document** of the **Python Project Code along with the Python Output Screen shots.**



Python Language
Introduction to Python Turtle
To Draw Shapes



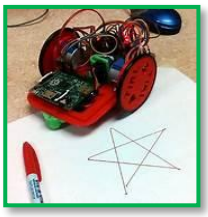
+++(Read and Study This Lesson)+++

Lesson 175 : How to Draw Shapes using Python Turtle program?

Turtle is a **Python Module** or **Library** which is used to **Create** and **Design Shapes, graphics, pictures, games** and **Animations**.

Python Turtle was developed by **Wally Feurzeig, Seymour Parpet** and **Cynthina Slolomon** in **1967**.

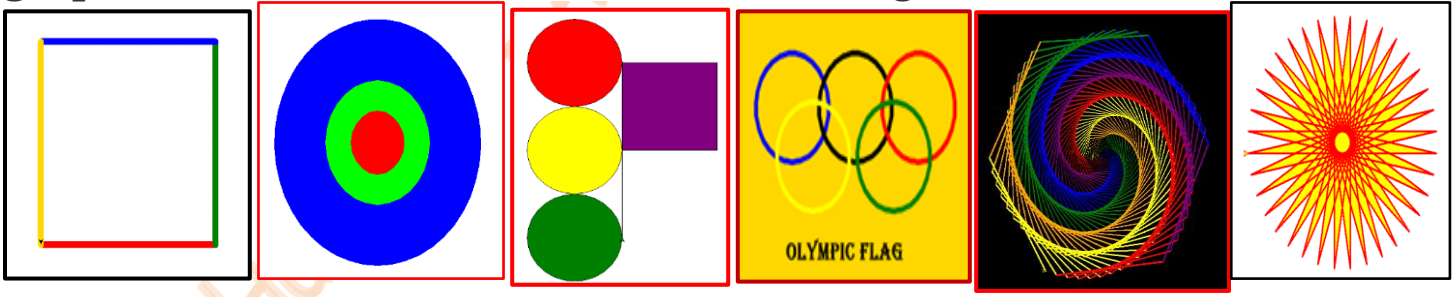
It was a part of the **original Logo programming language**.



In the **late 1960s**, **MIT professor Seymour Papert** used a **Robotic 'turtle'** to **teach programming to students**.

'turtle' is a **Python module** or **feature** like a **drawing board**, which lets us **command** a **turtle** to **draw all over the paper**.

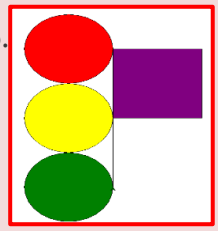
By using the **commands** in **Python turtle**, you can **draw nice shapes**, **graphics** and **do animations** like the following:



- **Python turtle** is a **Module** or **Library** which is used to **create shapes, patterns,** and **Games** on the Screen.
- **Python turtle** is initially **positioned** in the **Center** of the **Graphic Window** and the **turtle** looks like an **Arrow Head ▶**.
- **Python turtle** default heading is **0 degrees** and facing **East**.

```
import turtle
turtle.forward(100)
```

Some of Python turtle() functions:

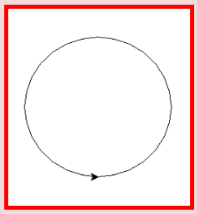


- **Turtle** is a **python feature** like a **drawing board**, which **allows** you **to command** a **turtle** to **draw Lines** and **make Shapes** and **Games**.
- **1)** To **use turtle**, we must **import turtle** module first.

```
import turtle
```

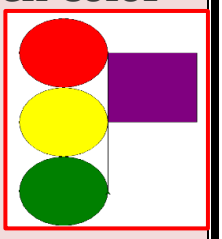
- **2)** Use **turtle.forward (100)** function to **move forward** and **draw** a **line** of **length 100 pixels**. **(96 pixels are equal to 1 inch)**
- **3)** Use **turtle.left (90)** function to **turn the turtle left** by **90 degrees (left is Up the window screen)**.
- **4)** Use **turtle.right (90)** function to **turn the turtle Right** by **90 degrees (right is Down the window screen)**.
- **5)** Use **turtle.setheading (90)** function to **set the turtle heading** to **90 degrees**.

- **6)** Use **turtle.heading ()** function to **get and display the turtle current heading**.
- **7)** Use **turtle.penup ()** function to **raise** the **turtle pen up** so **No Drawing will take effect**.
- **8)** Use **turtle.pendown ()** function to **lower the turtle pen down and will start Drawing**.



- **9)** Use **turtle.circle (25)** function to **Draw a Circle** of **radius 25 pixels**.
- **10)** Use **turtle.pensize (5)** function to **set** the **turtle pensize** to **size 5 pixels**. **(a little bit thicker than 1 pixel the default pensize)**
- **11)** Use **turtle.dot ()** function to **draw a Dot (.)** on window screen.

12) Use **turtle.pencolor ('red')** function to change the **pen color** to **red color**. (**Black** color is the **default pencolor**)



13) Use **turtle.bgcolor ('yellow')** function to change the **background color** of the window screen to **yellow color**.

14) Use **turtle.setup (640,480)** function to **set the size** of the **window screen** to **Width 640 pixels** and **Height** to **480 pixels**.

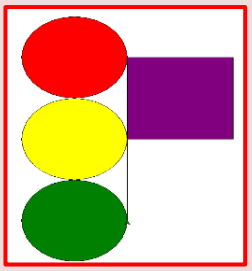
15) Use **turtle.goto (0, 100)** function to **move** the **turtle pen** to **location 0, 100** which **is x and y coordination's**.

16) Use **turtle.xcor ()** function to **return** the **x coordination** of the **current position** of **turtle pen**.

17) Use **turtle.ycor ()** function to **return** the **y coordination** of the **current position** of **turtle pen**.

18) Use **turtle.speed (0)** function to change turtle **Animation speed** of the **turtle pen drawing** to **0**. (**0** is the **Fastest speed with No Animation**). Turtle **Speed settings** can be set between **1 (Slowest)** to **10 (Fastest)**.

turtle.speed (0) is the Super Fastest and No Animation.
turtle.speed (10) is the Fast speed.
turtle.speed (6) is the Normal speed.
turtle.speed (3) is Slow speed.
turtle.speed (1) is the Slowest speed.



19) Use **turtle.hide ()** function to **hide** the **turtle arrow head**.

20) Use function **turtle.showturtle ()** to **show turtle arrow head**.

21) Use **turtle.write ('Ogar Haji')** function to **write the literal string 'Ogar Haji'** on the Window screen.

22) Use **turtle.begin_fill ()** function to **begin** to **fill** a **shape** with **color**

23) Use **turtle.fillcolor ('green')** function to **fill the shape** with **color green**.

24) Use **turtle.end_fill ()** function to **end** to **fill** a **shape** with **color**

25) Use **turtle.shape ('turtle')** to **display** the **turtle shape instead** of **Arrow shape.**

26) Use **turtle.home()** function to **move** the **turtle** to the **origin coordinates (0,0)**.

27) Use **turtle.done ()** function to Inform turtle that **we are done** and **leave the window open on screen**.

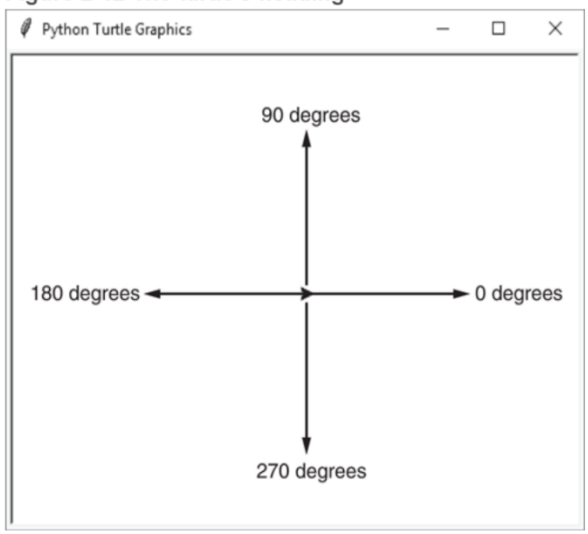
turtle.done()

The **python turtle Module or Library contains** all the **methods** and **functions** that we need to **create Shapes and Animation on screen**.

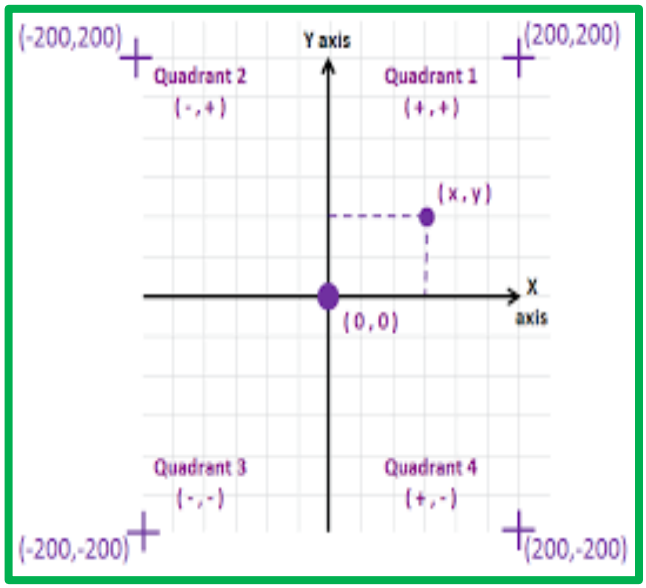
This shape is **Turtle Heading** ▶ which is **Facing toward East**.

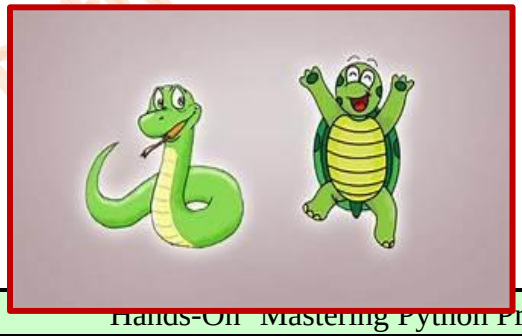
Turtle Left and Right Degrees

Figure 2-12 The turtle's heading

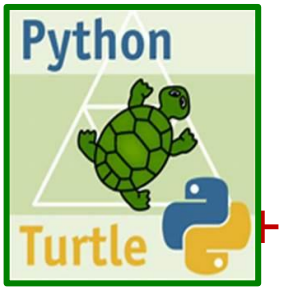


Turtle X and Y Coordinates

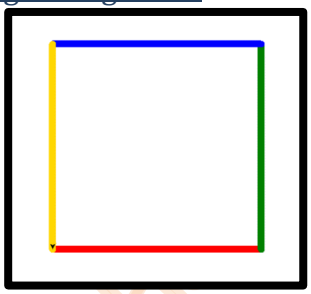




```
import turtle
turtle.forward(100)
```



Python Language to Draw a Square with Colored Lines (red, green, blue, gold)



++ (Read and Study This Lesson)

Lesson 176 Ex + How to Draw a Square with Thicker & Colored Lines in Python?

To Draw a Square with Colored Lines using Python turtle methods.

1) Use **turtle.pensize (10)** method to change the **pensize** to **10 thicker size**.

2) Use **turtle.pencolor ('red')** method to change **pen color** to **red** color.

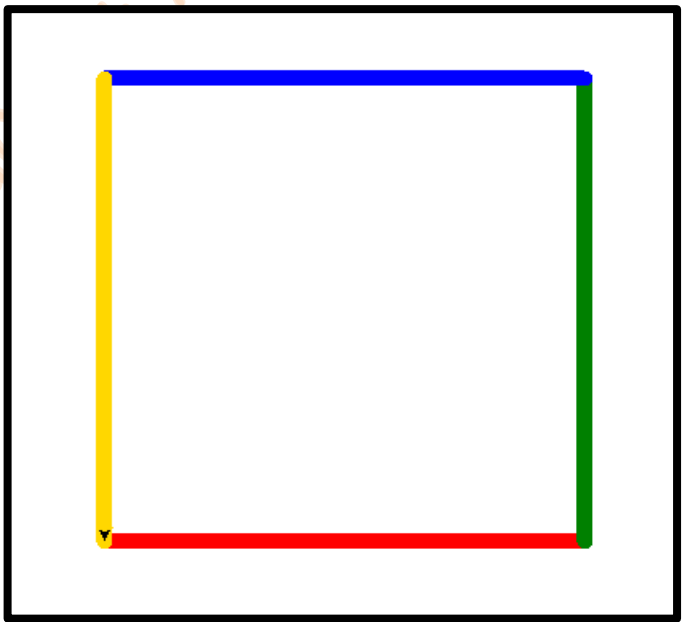
3) Use **turtle.forward (300)** method to go **forward** and draw a **red color line** 300 pixels.

```
import turtle
turtle.pensize (10)
turtle.pencolor ('red')
turtle.forward (300)
turtle.left (90)
```

turtle.pensize (10)

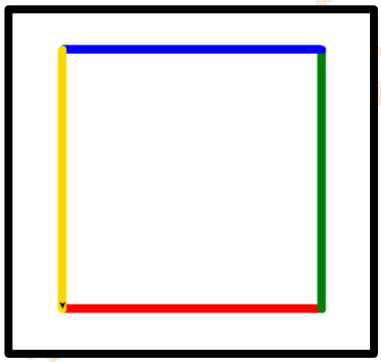
turtle.pencolor ('red')

The Output to Draw a Square with Different Lines Colors:



This is an Explanation of What we will Do in this project:

Draw a Square shape with 4 different colors lines
and make pen size thicker size like turtle.pensize(10)



1) import the turtle class Library
import turtle

2) Change the **turtle.pensize (10)** to 10 thicker size
turtle.pensize (10)

3) Change the **turtle.pencolor('red')** to red color
turtle.pencolor ('red')

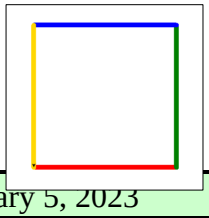
4) Move the **turtle forward** by **300** pixels
turtle.forward (300)

5) Turn the **turtle left by 90 degrees,** change pencolor and move forward
turtle.left (90)
turtle.pencolor ('green')
turtle.forward (300)

6) Turn the turtle **left by 90 degrees**, change pencolor and move forward
turtle.left (90)
turtle.pencolor ('blue')
turtle.forward (300)

7) Turn the turtle **left by 90 degrees**, change pencolor and move forward
turtle.left (90)
turtle.pencolor ('gold')
turtle.forward (300)

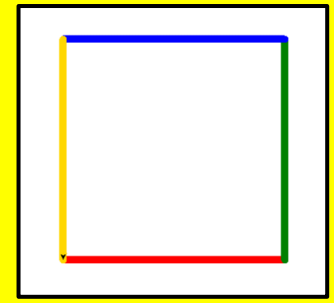
8) Signal to turtle you are done and leave the window open
turtle.done ()

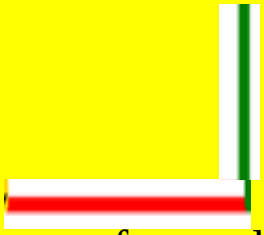


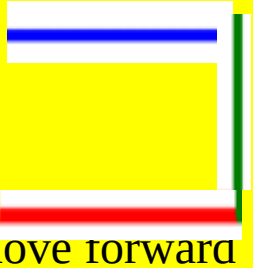
1) Get into **Python IDLE** and

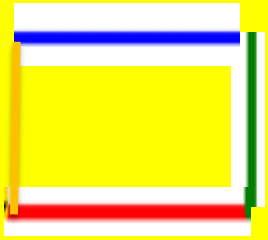
2) Save Python turtle project as **TurtleDrawSquareWithColoredLines**

```
# Draw a Square shape with 4 different colors lines
# and make pen size thicker size like turtle.pensize(10)

# 1) import the turtle class Library
import turtle

# 2) Change the turtle.pensize (10) to 10 thicker size
turtle.pensize (10)

# 3) Change the turtle.pencolor('red') to red color
turtle.pencolor ('red')

# 4) Move the turtle forward by 300 pixels
turtle.forward (300)

# 5) Turn the turtle left by 90 degrees, change pencolor and move forward
turtle.left (90)
turtle.pencolor ('green')
turtle.forward (300)

# 6) Turn the turtle left by 90 degrees, change pencolor and move forward
turtle.left (90)
turtle.pencolor ('blue')
turtle.forward (300)

# 7) Turn the turtle left by 90 degrees, change pencolor and move forward
turtle.left (90)
turtle.pencolor ('gold')
turtle.forward (300)

# 8) Signal to turtle you are done and leave the window open
turtle.done ()
```

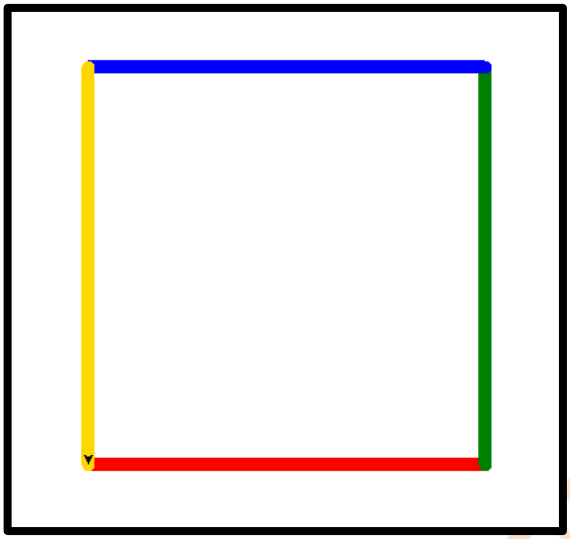








The output to Draw a Square with Different Lines Colors:



Modify This Project to Do the Following Modifications: 30%

A **Named Constant** is a **name that represents a Constant Value that does not change during the program's execution**.

The following is an example of how we will **declare named constants and initialize Named Constant** in our code:

```
PIXELS = 300
```

Note: It is **customary** to **Declare** a **Named Constant** in **ALL UPPER CASE** in **Python language**.

```
# 1.1 ) Declare a Named Constant 'PIXELS' and initialize it to 100
PIXELS = 300
```

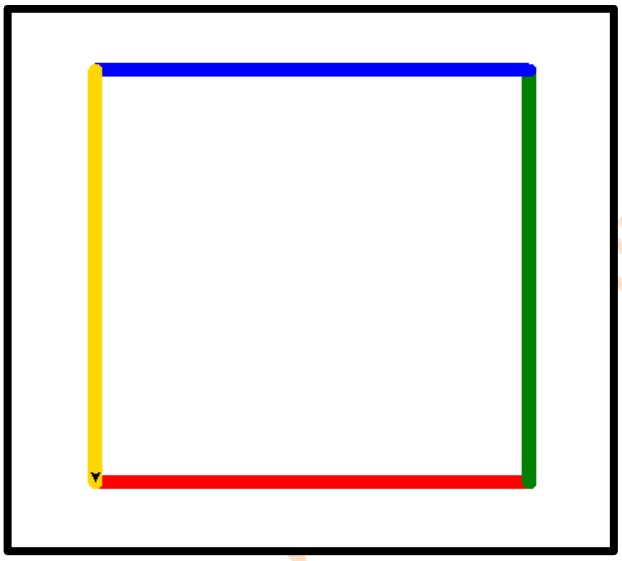
 # 2) Then add PIXELS to all turtle.forward (PIXELS) functions as shown below:

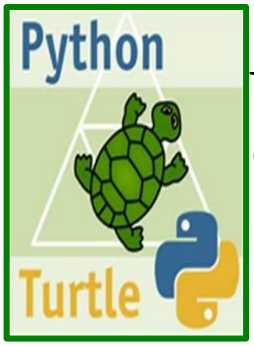
```
turtle.forward (PIXELS)
```

To do the following modifications, just assign these length 200 to the Named Constant 'PIXELS':

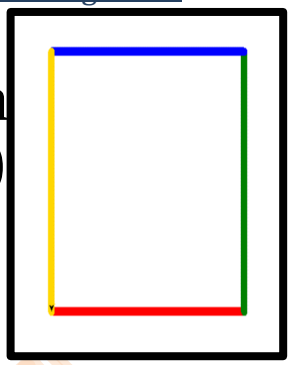
```
PIXELS = 200
```

1) **Use Turtle to Draw a Square of Length 200 pixels**.
2) **Use Turtle to Draw a Square of Length 350 pixels**.
3) **Use Turtle to Draw a Square of Length 400 pixels**.
4) **Upload the Last Modified project to Brightspace**.





Python Turtle Language
Use For and List to Draw Square with Colored Lines (red, green, blue, gold)
+++(Do Lab Exercise 6)+++



Do Lab Exercise 6

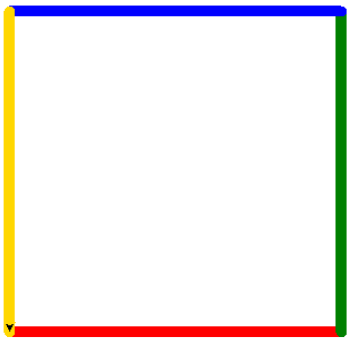
Lesson 177 Ex + How to use For and List to Draw a Square with Thicker & Colored Lines in Python?

We will declare **List** called **'colors_list'** and **assign** some **colors** to it:

4) Declare a List called 'colors_list' and assign different colors to it

colors_list = ['red', 'green', 'blue', 'gold']

red	green	blue	gold
0	**1**	**2**	**3**



in the above colors list:

red has an **index i value** of **0,**
green has **index i value** of **1**
blue has **index i value** of **2**
gold has **index i value** of **3**

To Draw a Square with Colored Lines using Python turtle methods.

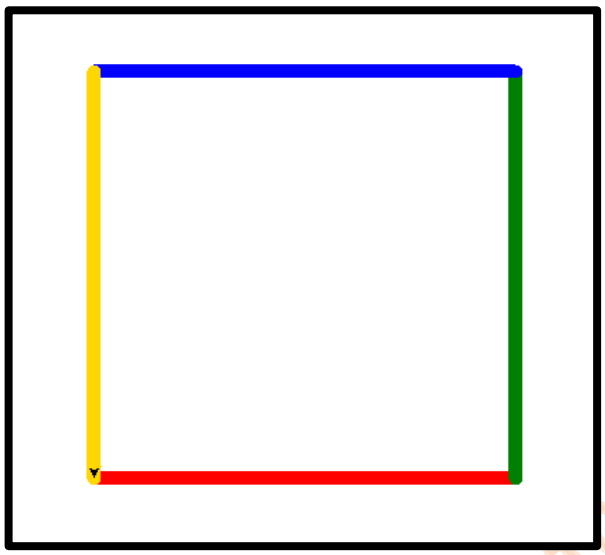
1) Use **turtle.pensize (10)** method to change the **pensize** to **10 thicker size**.
2) Use **turtle.pencolor ('red')** method to change **pen color** to **red** color.
3) Use **turtle.forward (300)** method to go **forward** and draw a **red color line** 300 pixels.

```
import turtle
turtle.pensize (10)
turtle.pencolor ('red')
turtle.forward (300)
turtle.left (90)
```

turtle.pensize (10)

turtle.pencolor ('red')

The Output to Draw a Square with Different Lines Colors:



+ Do Lab Exercise 6 + Do Lab Exercise 6 +

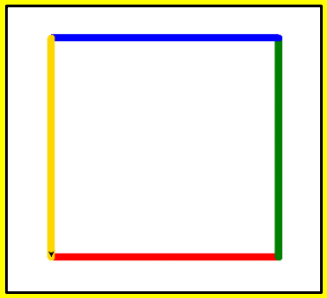
Do Lab Exercise 6

1) Type the following Python Project

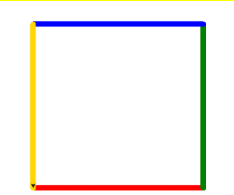
2) Save turtle project as **TurtleForDrawSquareColoredLines.py**

Do Lab Exercise 6

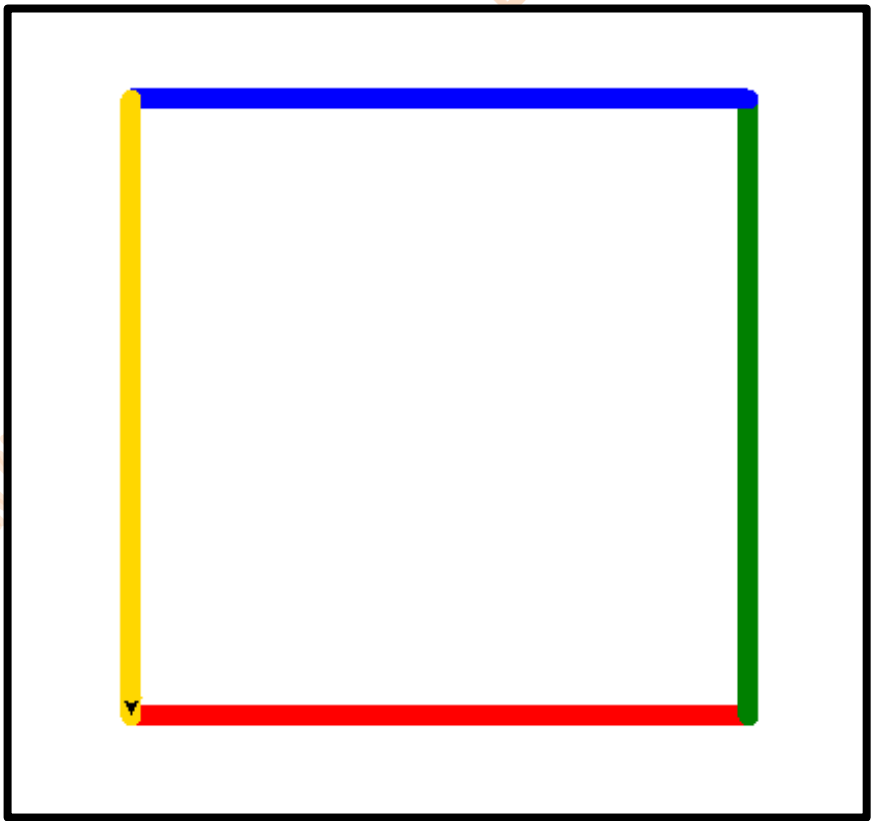
```
# Draw a Square shape with 4 different colors lines
# using for statement and a List of colors
# and make pen size thicker size like turtle.pensize(10)

# 1) import the turtle class Library.
import turtle

# 2) Set the turtle speed to 1 slow.
turtle.speed (1)

# 3) Change the turtle.pensize (10) to 10 thicker size
turtle.pensize (10)

# 4) Declare a list called colors_list and populate it with some colors
colors_list = ['red','green','blue','gold']
```



```
# 5) Use for statement to draw the colored square
# and assign the line colors from the colors_list
# Below color will start with index = 0 and so will select the red color from
# colors_list which is index 0
for color in (colors_list):
    turtle.pencolor (color)      # To access colors in the colors_list
    turtle.forward (300)
    turtle.left (90)

# 9) Signal to turtle you are done and leave the window open
turtle.done ()
```



The Output to Draw a Colored Lines Square

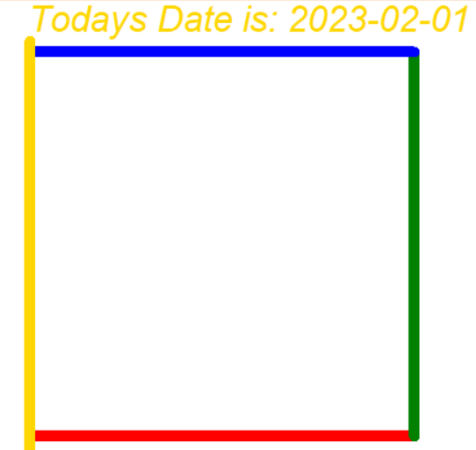


Modify This Project to Do the Following Modifications: 30%

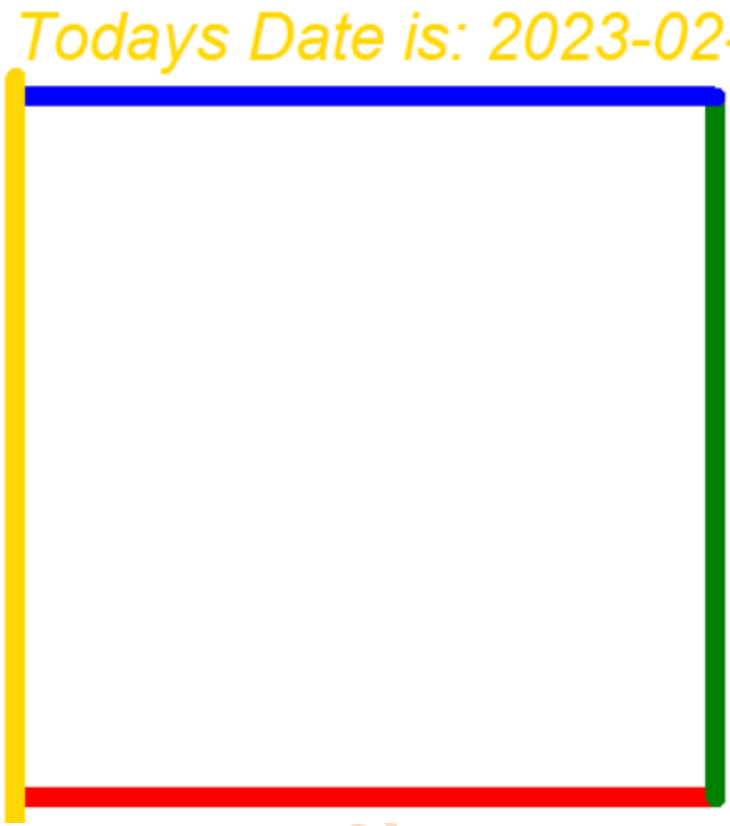
Modify the turtle project to do the following:

1) **Add** the **Computer Date** as a **Header** at the **Top** of the **image** by **Adding** the following **code above turtle.done()** method.

```
# 8) Import the following date classes:
from datetime import date

# 8.1) Create an instance object called 'today' from the module
#  date.today()

today = date.today ()

# 8.2) Declare the following x,y locations
# and set their values as shown below
x = 0
y = 360

# 8.3) Tell turtle to go to x,y position to write the Today's Date
turtle.goto (x,y)

# 8.4) Write the Today Date to output console
turtle.write ('Todays Date is: '+ str(today),
              font = ('arial', 25, 'italic' )  )
```



The following is the Output after adding the Computer Date:

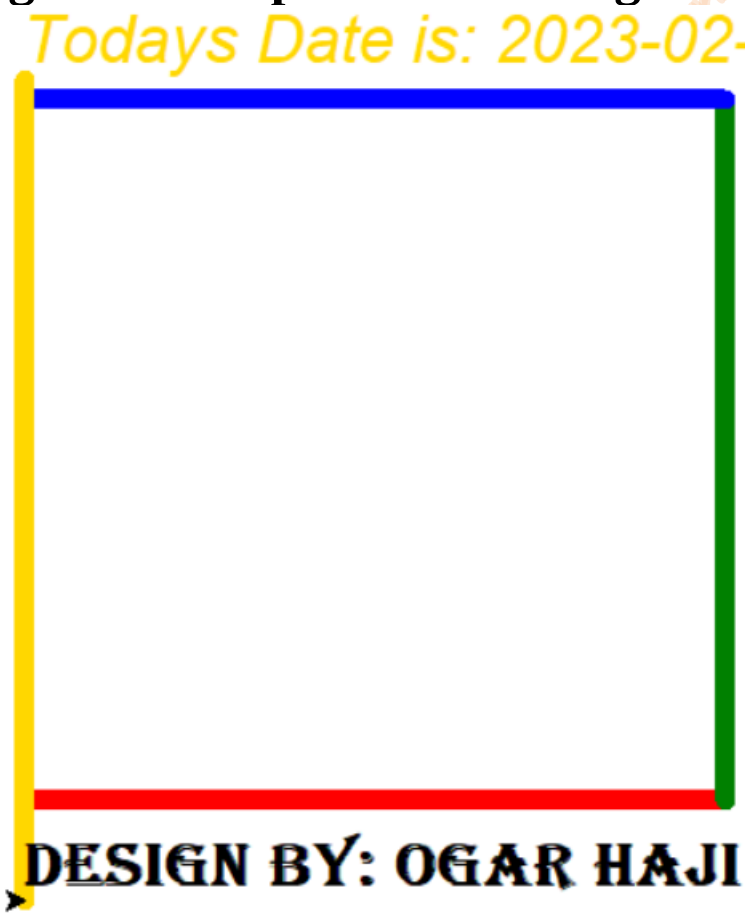


2) Add Your Name as a **Footer** at the **Bottom** of the **image** by **Adding** the following **code above turtle.done()** method.

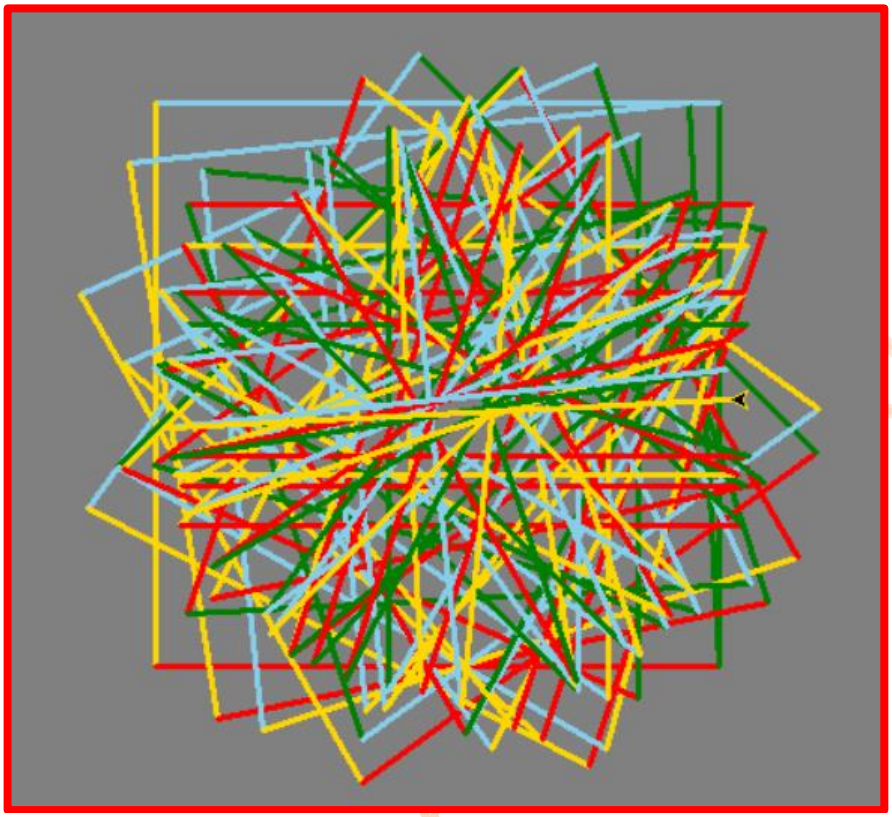
```
# 9) Use the turtle.write() method to write your name
# and also change the font type, size and make it bold
# 9.1) Declare the following x,y locations and set their values
x = 0
y = -50
# 9.2) Tell turtle to go to x,y position
# to write Your Name there
turtle.goto (x,y)
```

```
# 9.3) Change turtle pen color to black color
turtle.pencolor('black')

# 9.4) Write the Footer Your Name to output console
turtle.write ('Design by: Ogar Haji',
              font = ('algerian', 24, 'bold')  )
```

The following is the Output after adding Your Full Name:



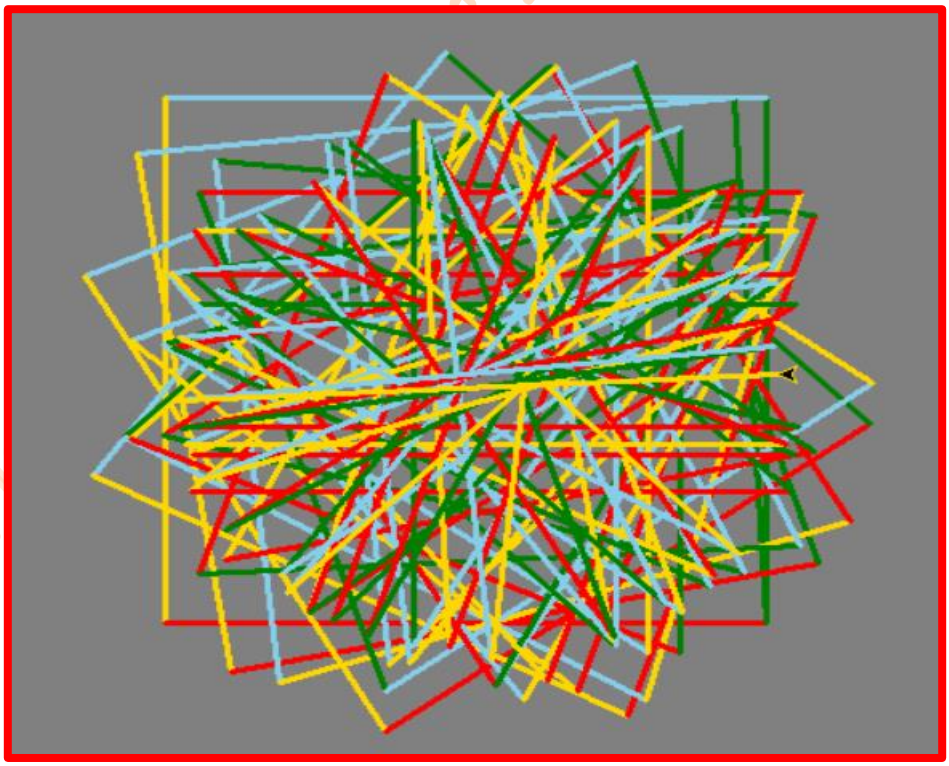
3) Upload the Last Modified project to Brightspace.

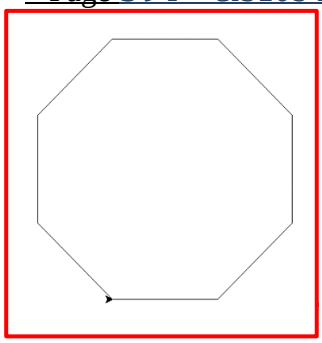


Modify the previous turtle project.

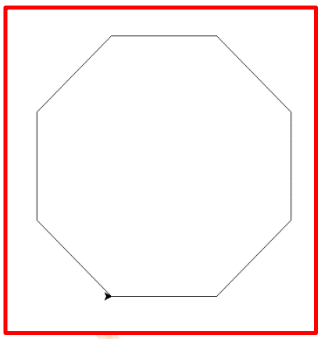
```
# Draw a Square shape with 4 different colors lines
# using for statement and a List of colors
# and make pen size thicker size like turtle.pensize(10)

# 1) import the turtle class Library
import turtle

# 2) Set the turtle speed to 0 (Fast) with no Animation
# Then change to 1 slow
turtle.speed(10)

# 3) Change the turtle.pensize (10) to 10 thicker size
turtle.pensize (3)
turtle.bgcolor('gray')
```

```
# 4) Declare a list called colors and populate with colors
colors_list = ['red','green','skyblue','gold']

# 5) Use Outer for loop to loop 10 times
for i in range (10):  # start with 10 then 20, 30, 40

    # 6) Use Inner for loop statement to draw the square
    # and assign the side colors with the list colors
    for color in  (colors_list):
        turtle.pencolor (color)
        turtle.forward (300)
        turtle.left (90 + i * 3)     # add 1 * 3 = 3 + 90 = 93

# 6) Signal to turtle you are done and leave the window open
turtle.done ()
```





Python Language
Introduction to Python Turtle
To Draw a Simple Octagon
(Read and Study This Lesson)



Lesson 178 Ex : How to Draw a Simple Octagon using Python Turtle program?

To Draw a Simple Octagon which has 8 sides using Python turtle methods as shown below:
Use 'for' statement with range (8) to draw All 8 Sides of the Octagon.

This is an Explanation of What we will Do in this project:

1) Import the turtle module
import turtle

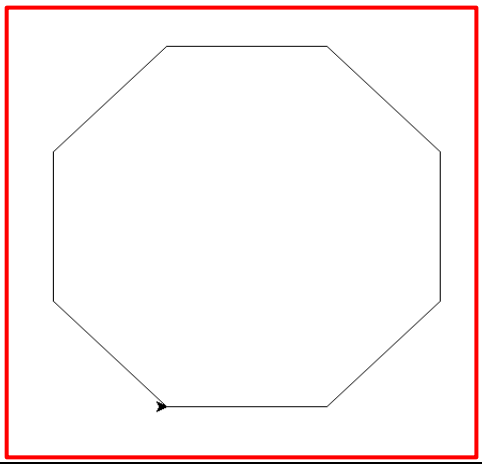
2) Use for statement to loop 8 times to draw all 8 sides of Octagon
for side in range (8):

```
for side in range (8):
    turtle.forward (100)
    turtle.left (45)
```

    # 3) Move the turtle forward 100 pixels
    turtle.forward (100)

    # 4) Turn the turtle left 45 degrees
    turtle.left (45)

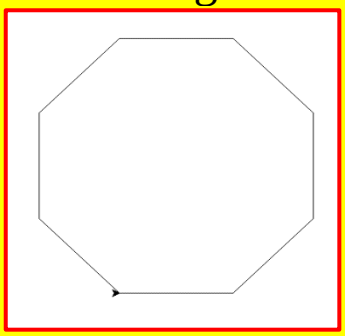
5) Inform turtle that you are done and leave the window open
turtle.done ()



1) Launch **Python IDLE IDE** program

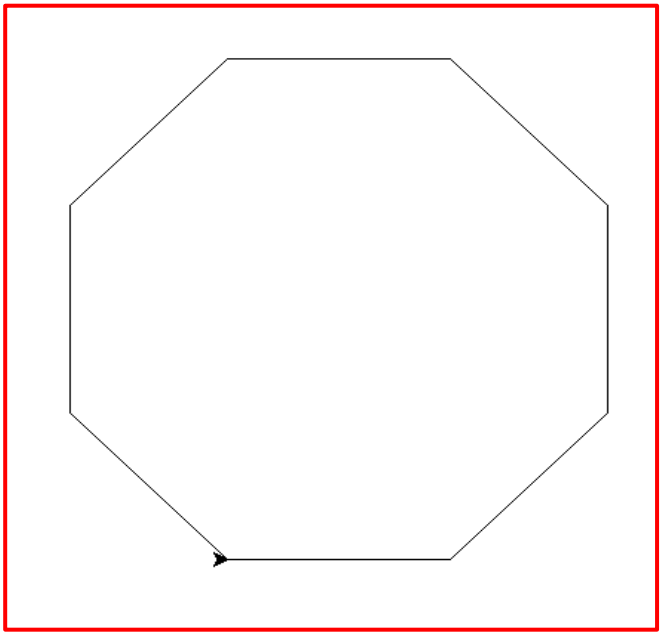
2) Type following **Python Lab exercise, Save** as **TurtleSimpleOctagon.py**.

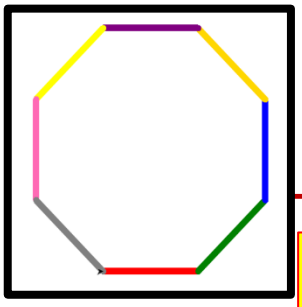
```
# 1) Import the turtle module
import turtle

# 2) Use for statement to loop 8 times to draw all 8 sides of Octagon
for side in range (8):
    # 3) Move the turtle forward 100 pixels
    turtle.forward (100)
    # 4) Turn the turtle left 45 degrees
    turtle.left (45)
# 5) Inform turtle that you are done and leave the window open
turtle.done ()
```



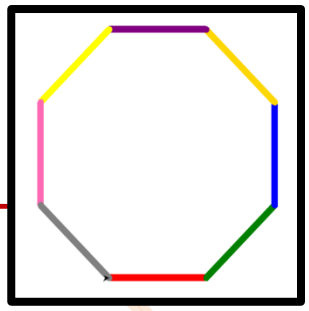
Octagon Shape With 8 Sides

```
for side in range (8):
    turtle.forward (100)
    turtle.left (45)
```





Python Language
Introduction to Python Turtle
To Draw a Colored Sides Octagon
++++++(Do Lab Exercise 7)++++



Do Lab Exercise 7

Lesson 179 Ex : How to Draw a Simple Octagon using Python Turtle program?

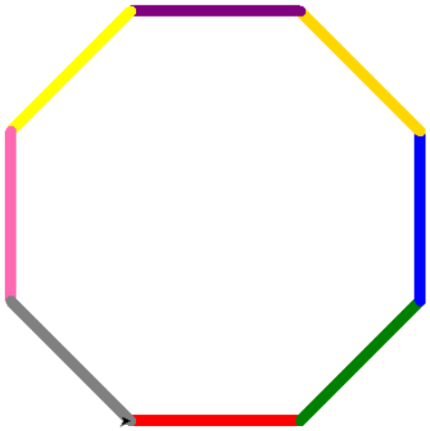
To Draw a Simple Octagon which has 8 sides using Python turtle methods as shown below:
Use 'for' statement with range (colors_list) to draw All 8 Colored Sides of the Octagon.

4) Declare a List called 'colors_list' and assign 8 different colors to it

colors_list = ['red','green','blue','gold','purple','yellow','hotpink','gray']

red	green	blue	gold	purple	yellow	hotpink	gray
0	**1**	**2**	**3**	**4**	**5**	**6**	**7**

in the above colors list:
red has an **index i value** of **0**
green has **index i value** of **1**
blue has **index i value** of **2**
gold has **index i value** of **3**
purple has an **index i value** of **4**
yellow has **index i value** of **5**
hotpink has **index i value** of **6**
gray has **index i value** of **7**



5) Use for statement to loop 8 times to draw all 8 sides of Octagon

for color in (colors_list):

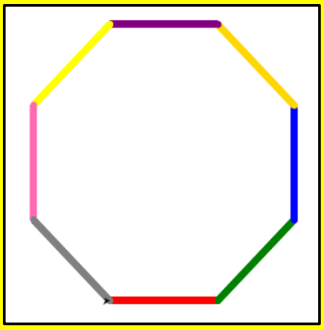
    # 6) Pick up or set the pencolor from the colors_list (red to gray)
    turtle.pencolor(color) # first time index side is 0 red

+ Do Lab Exercise 7 + Do Lab Exercise 7 +

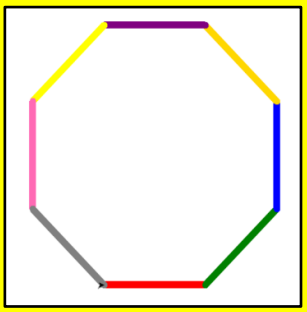
1) Launch **Python IDLE IDE** program.

Do Lab Exercise 7

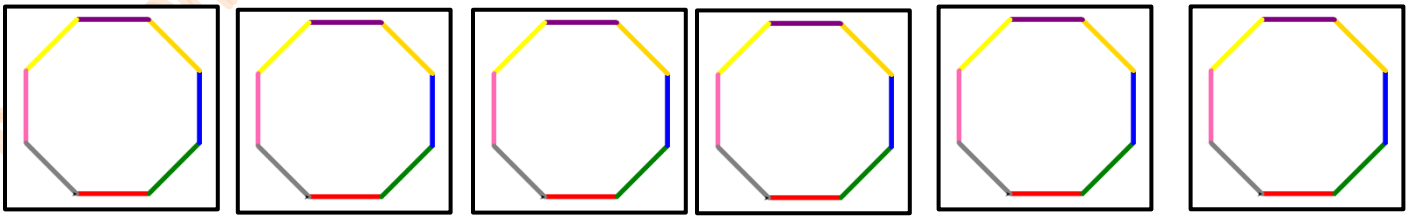
2) Type **Python Lab exercise, Save** as **TurtleColoredSidesOctagon.py.**



```
# 1) Import the turtle module
import turtle
# 2) Change the turtle pen size to size 10 (thicker)
turtle.pensize(10)
# 3) Change the turtle speed to 1 (slow)
turtle.speed(1)       # a little slow
# 4) Declare a List called 'colors_list' and populate it with colors
colors_list=['red','green','blue','gold','purple','yellow','hotpink','gray']
# 5) Use for statement to loop 8 times to draw all 8 sides of Octagon
for color in (colors_list):
    # 6) Pick up or set the pencolor from the colors_list (red to gray)
    turtle.pencolor (color)  # first time index side is 0 red
    # 7) Move the turtle forward 100 pixels
    turtle.forward (150)
    # 8) Turn the turtle left 45 degrees.
    turtle.left (45)
# 9) Inform turtle that you are done and leave the window open
turtle.done ()
```



Octagon Shape with 8 Colored Sides

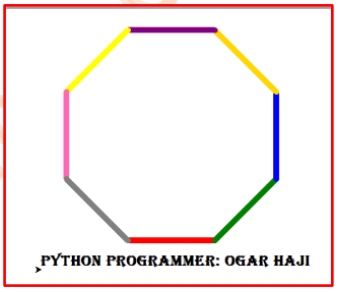


Modify This Project to Do the Following Modifications: 30%

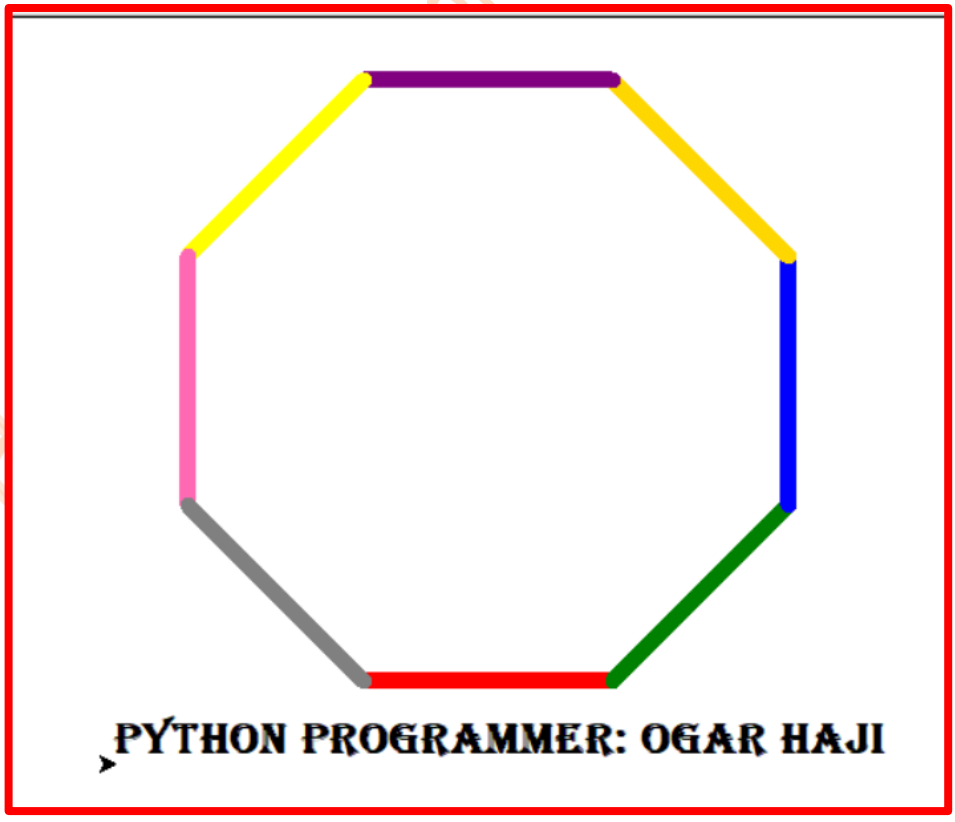
1) **Add Computer Date as a Header at the Top of the Image.**
2) **Add Your Full Name as the Footer at the Bottom of the Image.**

3) Use **turtle.write()** to write text in turtle module

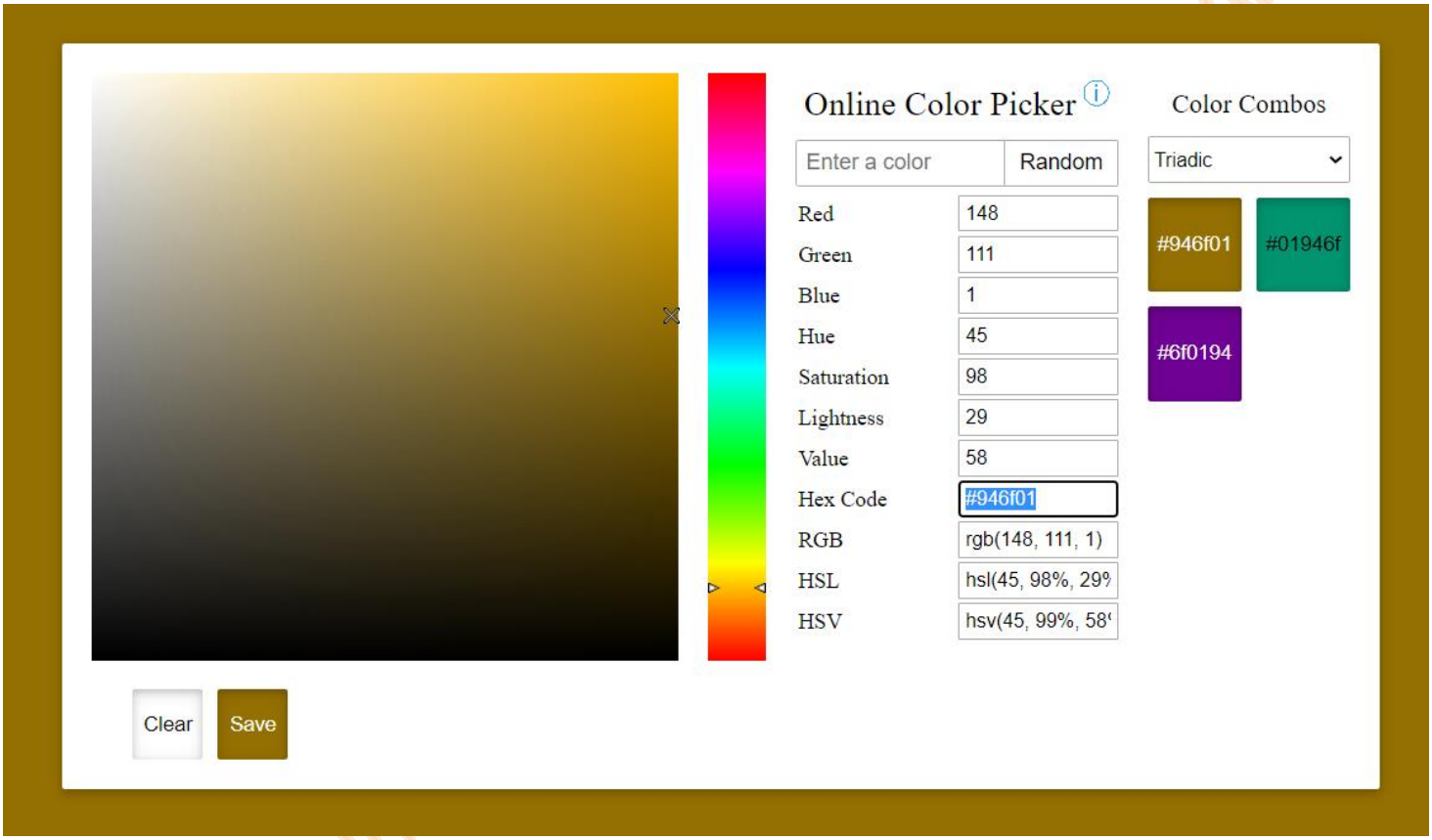
Do the following turtle code to write the text below the Octagon shape: **turtle.write('Python Programmer: Ogar Haji',**
font = ('algerian',20,'bold'))

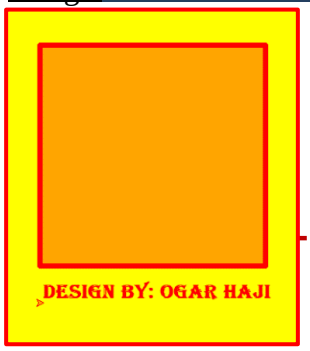



```
turtle.penup()
turtle.goto(-150,-50)
turtle.pendown()
turtle.pencolor('black')
turtle.write('Python Programmer: Ogar Haji',
        font = ('algerian',20,'bold'))
```



www.ColorPicker.me or www.w3schools.com/colors and click Color Picker on left side of screen





Python Language
Introduction to Python Turtle To Draw Shapes
+(Do Lab Exercise 4) 100 Points+

Do Lab Exercise 4

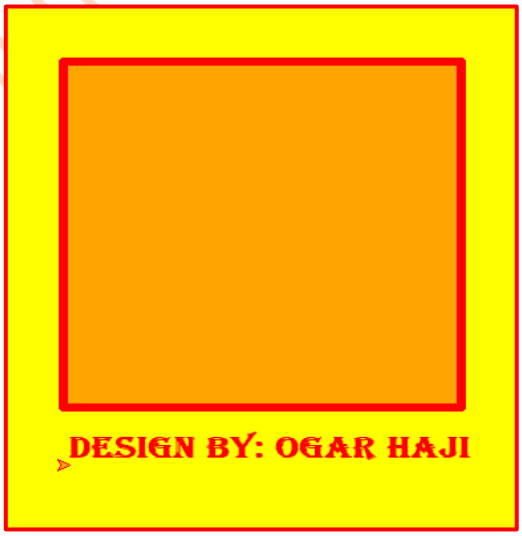
Lesson 179 Ex : How to Draw a Moon or Cirlce, Fill it with Color and Write a Message using Python Turtle program?

To Draw and Square, Fill it with Color and Write a Message using Python turtle methods.

Use **turtle.shape ('turtle')**  to **display** the **turtle shape instead** of **Arrow**. 

Output of the Python Turtle project to Draw a Square and Color it

```
for line in range (4):
    turtle.forward (300)
    turtle.right (90)
```



This is an Explanation of What we will Do in this project:


```
# This Turtle project will draw a Square and change
# the pencolor to red and the fillcolor with orange
# Programmer: Ogar Haji
# 1) Import the turtle module library
import turtle
```

1.2) Change turtle shape to 'turtle'
turtle.shape ('turtle')

2) Change the turtle speed to 1 which is very slow
turtle.speed (1) # speed 1 is very slow

3) Change the turtle pensize to size 7 which is thicker
turtle.pensize (7)

4) Change the background color of the screen to yellow
turtle.bgcolor ('yellow')

5) Begin the turtle begin_fill to start the fill color
turtle.begin_fill()

6) Change the turtle pencolor to hexidecimal red #ff0000
turtle.pencolor ('#ff0000')

7) Change the turtle fillcor to Orange which is hex #ffa500
turtle.fillcolor ('#ffa500')

8) Use for statement with range (4) to loop 4 times and draw the Square
for line in range (4):

    # 1) Draw First line of the length 0f 300 pixels
    turtle.forward (300)

    # 2) Turn the turtle right by 90 degrees
    turtle.right (90)

9) End the turtle fill to end_fill the colors
turtle.end_fill()

10) Raise the turtle pen by applying turtle.penup() function
turtle.penup ()

11) Move the turtle to location (5, -350) by
calling the function turtle.goto ()
turtle.goto (5,-350)

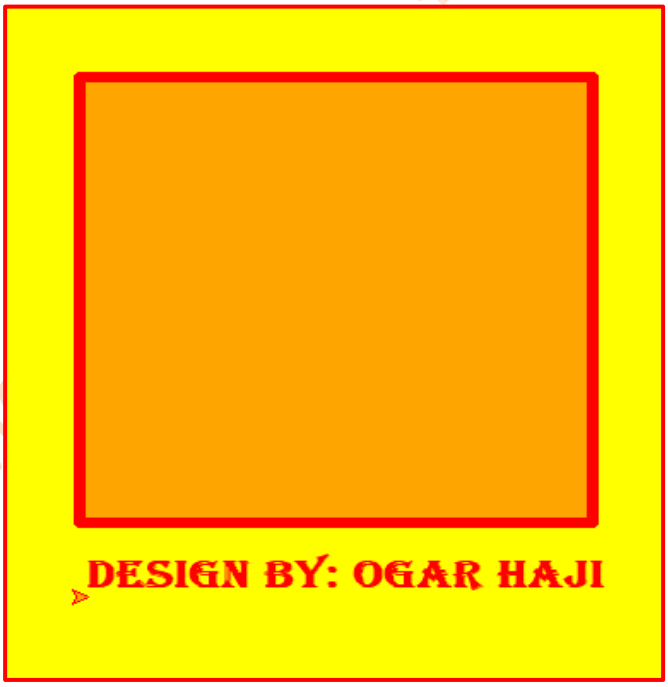
12) Put the pendown by calling the function turtle.pendown()
turtle.pendown ()

13) Use the turtle.write() function to write your name
and also change the font type, size and make it bold
turtle.write ('Design by: Ogar Haji', font = ('algerian', 20, 'bold'))

14) End the execution of the turtle program
turtle.done()

Output of the Python Turtle project to Draw a Square and Color it

```
for line in range (4):
    turtle.forward (300)
    turtle.right (90)
```



1) Launch **Python IDLE IDE** program and type following
2) **Save** file as **TurtleDrawSquareFillWithColor.**

Do Lab Exercise 4

```
# This Turtle project will draw a Square and change
# the pencolor to red and the fillcolor with orange
# Programmer: Ogar Haji
# 1) Import the turtle module library
import turtle
# 1.2) Change turtle shape to 'turtle'
turtle.shape ('turtle')
```

Do Lab Exercise 4

2) Change the turtle speed to 1 which is very slow
turtle.speed (1) # speed 1 is very slow

3) Change the turtle pensize to size 7 which is thicker
turtle.pensize (7)

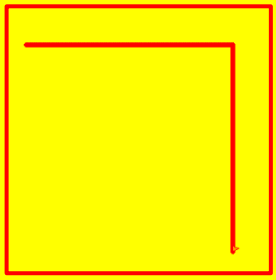
4) Change the background color of the screen to yellow
turtle.bgcolor ('yellow')

5) Begin the turtle begin_fill to start the fill color
turtle.begin_fill ()

6) Change the turtle pencolor to hexadecimal red #ff0000
turtle.pencolor ('#ff0000')

7) Change the turtle fillcor to Orange which is hex #ffa500
turtle.fillcolor ('#ffa500')

8) Use for statement with range (4) to loop 4 times and draw the Square
for line in range(4):



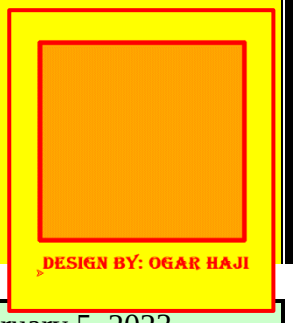
    # 1) Draw First line of the length 0f 300 pixels
    turtle.forward (300)

    # 2) Turn the turtle right by 90 degrees
    turtle.right (90)

9) End the turtle fill to end_fill the colors
turtle.end_fill()

10) Raise the turtle pen by applying turtle.penup() function
turtle.penup ()

11) Move the turtle to location (5, -350) by calling the function goto()
turtle.goto (5,-350)



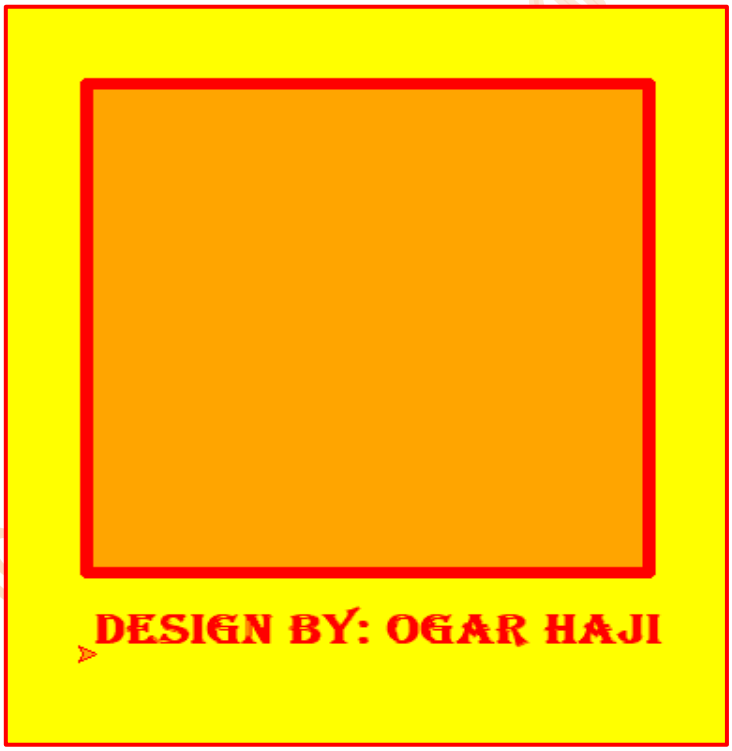
#12) Put the turtle pendown by calling function turtle.pendown()
turtle.pendown ()

13) Use the turtle.write() function to write your name

```
# and also change the font type, size and make it bold
turtle.write ('Design by: Ogar Haji', font = ('algerian', 20, 'bold'))

# 14) End the execution of the turtle program
turtle.done ()
```

Output of the Python Turtle project to Draw a Square and Color it

```
for line in range (4):
    turtle.forward (300)
    turtle.right (90)
```



Chapter 4 + Quiz 3 + Test Your Python Language Knowledge:

---

1. What is the output after executing the following Python code?

programming_language = 'Python'

print (programming_language [-1]) # index **[-1]** returns last index char '**n**'

A. Nothing is printed B. n C. -1 D. P E. error

---

2. What is the output after executing the following Python code?

print (type (7 / 2)) # / is **float** division, so the **type** is **float**

A. 2.5 B. 2 C. <class 'int'> D. <class 'float'> E. error

---

3. What is the output after executing the following Python code?

import math

print (math.ceil (7.23)) # **math.ceil() raises up** value to **next integer 8**

A. 7 B. 7.23 C. 7.2 D. 8 E. error

---

4. What is the output after executing the following Python code?

print (math.floor (7.23)) # **math.floor() lower down** value to integer **7**

A. 7 B. 7.23 C. 7.2 D. 8 E. error

---

5. What is the output after executing the following Python code?

print (round (7.23)) # because **.23** so it **rounds down** to integer **7**

A. 7 B. 7.23 C. 7.2 D. 8 E. error

---

6. What is the output after executing the following Python code?

print (round (7.53)) # because **.53** so it **rounds up** to integer **8**

A. 7 B. 7.23 C. 7.2 D. 8 E. error

---

7. What is the output after executing the following Python code?

```
import random
random_number = random.randint (1, 3)
print (random_number)
```

A. 0 thru 3 B. 1 thru 3 C. 0 thru 4 D. 1 thru 4 E. error

---

8. What is the output after executing the following Python code?

```
if 5 > 10:                  # This is False because 5 is Not > than 10
   print ('Python')
elif 8 != 9:                # This is True because 8 is Not equal to 9
   print ('language')      # this statement is executed, so language is printed
else:
   print ('is easy')
```

A. Python B. language C. is easy D. Python is easy E. error

---

9. What is the output after executing the following Python code?

```
if False:                   # False will skip to next elif
   print ('Chevy')
elif True:                  # True will execute this and print 'Ford'
   print ('Ford')
elif True:
   print ('Buick')
else:
   print ("Jeep")
```

A. Chevy B. Ford C. Buick D. Jeep E. error

---

10. What is the output after executing the following Python code?

```
number = int (input ('Enter an integer like 21: ') )   # user enters 21
print ( len (number) )        # len () is used to find the length of strings, lists
```

A. 21 B. 2 C. TypeError: object of type 'int' has no len() D. 3

Answers Are Found at The End of This Chapter 04.

Chapter 4 + Python Homework #4B (Due This Sunday) 100 Points

Name:_____ **CIS103 Python** Programming Language + Wright College

Do the Following Chapter 4 Part B Python Homework: *Instructor: Ogar Haji*

1) What is the **math module**, explain briefly?

2) Write the Python code to **import** the **math** module.

3) Write the Python code to get **help** on **math** module.

4) Write the Python code to get the **directory** of **math** module.

5) What is the math **.factorial()** method, give an example?

6) What is **the result** of the following Python example?

print (math.factorial (7))

7) What is the math **.pi()** method, give an example?

8) What is **the result** of the following Python example?

print ('The Value of math.pi = ' , math.pi)

9) What is a **for** statement **used for** in Python Language?

10) What is the output after executing the following Python code?

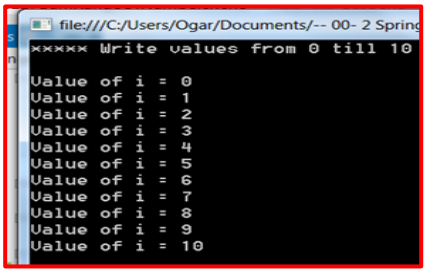
```
greeting = 'Good Morning'
for ch in greeting:
if ch == 'o':          # ch is equal to G, so else will execute
      break
      print (ch)
 else:
      print ('Good Night')
```

A. G B. Go C. Good Morning D. Good Night E. error

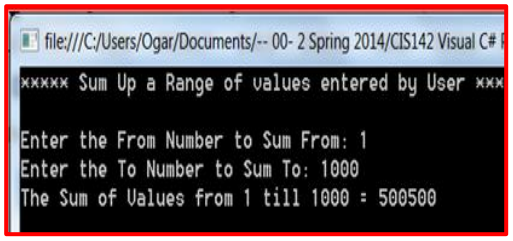
11)

12) What are the **3 important features** that are **included** in a **for statement**?

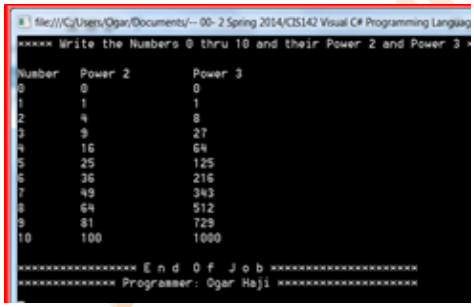
13) Write the Python code to **Print** to the **Console the Numbers from 0 till 10.**



14) Write the Python code to **Prompt** the **user** to **Enter** the **From Number** and also **prompt user** for **To Number** and then **Sum Up** the **Numbers** in that **range.**



15) Write the Python code to **use While** statement and **Print to Console the Power 2** and **Power 3** of **Numbers from 0 till 10.**



16) What does the **max()** function do, Give a Python coded example?

17) What does the **min()** function do, Give a Python coded example?

18) What does the **math.fabs()** method do, Give a Python coded example?

19) What does the **random.random()** method do, Give a Python coded example?

20) What does the **math.ceil()** method do, Give a Python coded example?

21) What does the **math.floor()** method do, Give a Python coded example?

22) What does the **round()** function do, Give a Python coded example?

23) What does the **math.sqrt()** method do, Give a Python coded example?

24) What does the **math.pow(3, 2)** method do, Give a Python coded example?

25) What is the value of **math.pi** Constant Value, Give a Python coded example?

26) What does the **math.remainder()** method do, Give a Python coded example?

27) What is **the result** of the following Python **max()** function? _____

max_number = max (22, 33, 11, 77)

a. 22 b. 33 c. 11 d.77 e. 0

28) What is **the result** of the following Python **min()** function? _____

min_number = min (22, 33, 11, 77)

a. 22 b. 33 c. 11 d.77 e. 0

29) What is **the result** of the following Python math.**ceil()** method? _____

ceil_number = math.ceil (8.56789)

a. 7.0 b. 8.0 c. 9.0 d. 0

30) What is **the result** of the following Python math.**floor()** method? _____

floor_number = math.floor (8.56789)

a. 7.0 b. 8.0 c. 9.0 d. 0

31) What is **the result** of the following Python **round()** function? _____

```
print (round (8.56789) )
```

a. 7.0 b. 8.0 c. 9.0 d. 0

32) What is **the result** of the following Python **random.random** method? _____

```
random_number = random.random ()
```

a. Less than zero b. greater than 1 c. between 0 and less than 1 d. greater than 10

33) What is **the result** of the following Python random **randint** method? _____

```
random_number = random.randint (0,9)
```

a. Less than zero b. greater than 1 c. between 0 and 9 d. greater than 10

34) What is **the result** of the following Python **math.sqrt()** method? _____

```
square_root_number = math.sqrt (49)
```

a. 5.0 b. 6.0 c. 7.0 d. 0

35) What is the result of the following Python **math.pi** method? _____

```
print (f'Value of math.pi is = {math.pi }')
```

36) What is the result of the following Python **math.pi** method? _____

```
print (f'Value of math.pi is = {math.pi  :.4f }')
```

37) What is the result of the following Python **math.pi** method? _____

```
print (f'Value of math.pi is = {math.pi  :.2f }')
```

38) What is **the result** of the following Python **math.pow()** method? _____

```
power_number = math.pow (5, 2)
```

a. 5.0 b. 7.0 c. 25.0 d. 0

39) Write the Python code to **Sum** the **Odd Numbers from 0 till 100** and **print to output** the **Sum** of these **numbers**.

40) Write the Python code to **Sum** the **Even Numbers from 0 till 100** and **print to output** the **Sum of these numbers**.

41) What is a **'break'** keyword used for in Python Language, explain briefly?

42) What is the output after executing the following Python code?

```
for i in range (5):          # Outer nested for 'i' loop from 0 to 4
    for j in range (5):      # Inner nested for 'j' loop from 0 to 4
        if j == 3:
            break            # break out of the Inner for loop when j == 3
print (i * j)                # after exiting for loop, i = 4, j = 3
```

A. 25 B. 20 C. 16 D. 12 E. error

43) Write the Python code to try to **print the numbers from 1 to 25** and **break out** when the **number** is **equal to 13**.

41) What is the output after executing the following Python code?

```
sum = 0
for i in range (1, 5):
    sum  +=  i
print (sum)
```

A. 0 B. 1 C. 10 D. 15 E. error

44) What is a **'continue'** keyword used for in Python Language, explain briefly?

45) What is the output after executing the following Python code?

```
for i in range (5):          # Outer nested for loop from 0 to 4
    for j in range (5):      # Inner nested for loop from 0 to 4
        if j == 3:
            continue         # continue will continue the Inner for loop
print (i * j)                # after exiting for loop, i = 4, j = 4
```

A. 25 B. 20 C. 16 D. 12 E. error

46) What is the output after executing the following Python code?

```
for i in range (5):          # Outer nested for loop from 0 to 4
    for j in range (5):      # Inner nested for loop from 0 to 4
        if j == 3:
            continue         # continue will continue the Inner for loop
print (i * j)                # after exiting for loop, i = 4, j = 4
```

A. 25 B. 20 C. 16 D. 12 E. error

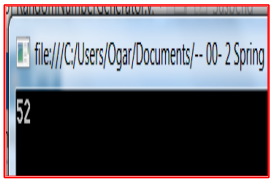
47) What is the output after executing the following Python code?

```
str = 'Python language is easy to learn'
print ('easy' in str)        # The 'in' operator is used to find if 'easy' is in str
```

A. easy B. Python C. True D. False E. error

48) Write the Python code to try to **print the numbers from 1 to 100** and **continue printing but without** the **number 21**.

49) Write the Python code to **Generate a Number between 0 till 100.**



50) Write the Python code to **Declare** a **List** called **'colors list'** and **populate** it with **4 color names**.

51) Write the Python code to **loop inside** the **'colors list'** and to **print each color** in the **colors list**.

52) What is the output after executing the following Python code?

print (7 % 2 ** 3) # Evaluate the ** first, **2 ** 3 = 8**, then **7 % 8 = 7**

A. 2 B. 3 C. 7 D. 8 E. error

53) To Change the shape of the turtle to '**turtle**' Shape, type the following:

turtle.shape ('turtle') (True/False)

CIS 103 Python Instructor:
Ogar Haji

Chapter 4 + Python Lab Assignment #4A (Due This Sunday) 100 Points

Random Number Generator: Pick 3 Lotto

Check if Pick3 Lotto is the Winner Ticket

+++(Lab Assignment + Lab 4A)+++

Lab Assignment Lab 4 A

Lesson 174 Ex: How to Generate Pick 3 Lotto Random Numbers in Python ?

Problem or Project: Design and Code in Python Language the project to use random.randint(0,9) module to Generate Random Numbers from 0 to 9 for Pick 3 Lotto.

The Input/Output of the Project 'Pick 3 Lotto' Project

```
= RESTART: C:\Users\ogarh\AppData\Local\Programs\P
********** Pick 3 Lotto **********
Pick 3 Lotto number is: 259.

Enter the Pick3 Lotto Today Winning Number: 259
Today Pick3 Lotto Winning Number is: 259

Congratulations. Your Number '259' is the Winner.

******** End of the Project **********
***** Python Programmer: Ogar Haji *******

= RESTART: C:\Users\ogarh\AppData\Local\Programs\P
********** Pick 3 Lotto **********
Pick 3 Lotto number is: 901.

Enter the Pick3 Lotto Today Winning Number: 777
Today Pick3 Lotto Winning Number is: 777

Sorry. Your Number '901' is NOT the winner Today.

******** End of the Project **********
***** Python Programmer: Ogar Haji *******
```

This is an Explanation of What we will Do in this project:

```
# 1) Add Comments about the project on the top of the project.
'''
    This project will randomly create a Pick3 Lotto.
    Then it will prompt the user to enter the pick3 Winning number
    And then it will check if the numbers are the same (Winner)
    Date: Saturday, September 11, 2020
    Programmer: Ogar Haji
'''
```


```
# 2) Import random module
import random

# 3) Declare a string variable and initialize to empty string
pick3 = ''
```


```
# 4) Print the Headings of the Project
print('********** Pick 3 Lotto **********')

# 5) Use random.randint() module to Generate Random Numbers
# 6) Use for statement to print out to output console Pick 3 Lotto
for i in range (3):
    # 1) Generate random numbers from 0 till 9
    # and then convert to strings
    random_number = str (random.randint (0,9) )
    pick3 +=  random_number
    # End of for statement
# 7) Print out the pick3 lotto number
print (f'Pick 3 Lotto number is: {pick3}.')

# 8) Print out a Blank line
print ()
```

```
# 9) Prompt the user to Enter the Pick3 Lotto Winning Number
winning_number = input ('Enter the Pick3 Lotto Today Winning Number: ')
# 10) print the winning number and a blank line
print (f'Today Pick3 Lotto Winning Number is: {winning_number}')
print ()
# 11) Check if you are the winner
if pick3 == winning_number:
    print (f"Congratulations. Your Number '{pick3}' is the Winner. You won $500.")
else:
    print (f"Sorry. Your Number '{pick3}' is NOT the winner Today.")
# 12) Print the Footers of the Project
print ('\n********** End of the Project **********')
print ('***** Python Programmer: Ogar Haji *******')
```

+ Do Lab Exercise 4A + Do Lab Exercise 4A +

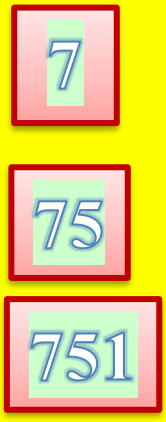
Do Lab Exercise 4A

1) **Ex. Type** the following **Python Code in IDLE**
2) **Save** Python project as **Pick3LottoCheckWinners.py**

```
# 1) Add Comments about the project on the top of the project.
'''
        This project will randomly create a Pick3 Lotto.
        Then it will prompt the user to enter the pick3 Winning number
        And then it will check if the numbers are the same (Winner)
        Date: Saturday, September 11, 2020
        Programmer: Ogar Haji
'''
# 2) Import random module or library
import random
```

Do Lab Exercise 4A

```
# 3) Declare a string variable and initialize to empty string
pick3 = ''

# 4) Print the Headings of the Project
print('********** Pick 3 Lotto **********')

# 5) Use random.randint() module to Generate Random Numbers
# 6) Use for statement to print out to output console Pick 3 Lotto
for i in range (3):
    # 1) Generate random numbers from 0 till 9
    # and then convert to strings
    random_number = str (random.randint (0,9) )
    pick3 +=  random_number
    # End of for statement

# 7) Print out the pick3 lotto number
print (f'Pick 3 Lotto number is: {pick3}.')

# 8) Print out a Blank line
print ()

# 9) Prompt the user to Enter the Pick3 Lotto Winning Number
winning_number = input ('Enter the Pick3 Lotto Today Winning Number: ')

# 10) print the winning number and a blank line
print (f'Today Pick3 Lotto Winning Number is: {winning_number}')
print ()

# 11) Check if you are the winner
if pick3 == winning_number:
   print (f"Congratulations. Your Number '{pick3}' is the Winner. You won $500.")
else:
   print (f"Sorry. Your Number '{pick3}' is NOT the winner Today.")
```



```
# 12) Print the Footers of the Project
print ('\n********** End of the Project **********')
print ('***** Python Programmer: Ogar Haji *******')
```

The Input/Output of the Project 'Pick 3 Lotto' Project

```
= RESTART: C:\Users\ogarh\AppData\Local\Programs\P
********** Pick 3 Lotto **********
Pick 3 Lotto number is: 259.

Enter the Pick3 Lotto Today Winning Number: 259
Today Pick3 Lotto Winning Number is: 259

Congratulations. Your Number '259' is the Winner.

******** End of the Project **********
***** Python Programmer: Ogar Haji *******

= RESTART: C:\Users\ogarh\AppData\Local\Programs\P
********** Pick 3 Lotto **********
Pick 3 Lotto number is: 901.

Enter the Pick3 Lotto Today Winning Number: 777
Today Pick3 Lotto Winning Number is: 777

Sorry. Your Number '901' is NOT the winner Today.

******** End of the Project **********
***** Python Programmer: Ogar Haji *******
```

259 259 259 901 777 901

Modify This Project to Do the Following Modifications: 30%

Modify this Python Project to do the following:

1) **Check if the Lotto Player has 1 Matching Number in the Winning Lotto, then the Lotto Player will win $10.**
2) **Check if the Lotto Player has 2 Matching Number in the Winning Lotto, then the Lotto Player will win $20.**
3) **Check if the Lotto Player has 3 Matching Number in the Winning Lotto, then the Lotto Player will win $30.**

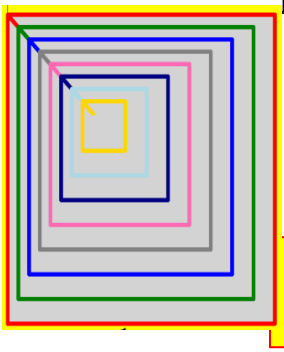
The following is the Python Modification Code

```
# 11) Check if you are the winner
if pick3 == winning_number:
    print (f"Congratulations. Your Number '{pick3}' is the Winner. You won $500.")
    winner = True
else:
    print (f"Sorry. Your Number '{pick3}' is NOT the winner Today.")
    winner = False
# 12) Convert the pick3 and winning_number to string using str() function
pick3 = str(pick3)
winning_number = str(winning_number)
# 13) Check if the winner flag is False (means the pick3 is Not a Winner)
if winner == False:
        digits_count = 0            # declare and initialize count to 0
        # 14) Use Outer for in range statement to loop the pick3 Lotto numbers
        for x in range (0, len(pick3)):
            print ('inside first for x statement', x)
            # 15) Use Inner for in statement to loop winning numbers
```

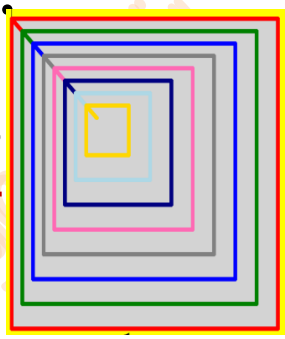
```
        for y in range (0, len(winning_number) ):
                if (pick3[x]) == (winning_number[y]):
                    digits_count += 1
                else:
                    pass

    # End of Outer for x statement

    # 15) Check the numbers of pick3 been matched with winning number
    if digits_count == 1:
        print ('You have 1 Digit in the Winning ticket')
        print ('You have Won $10.')

    elif digits_count == 2:
        print ('You have 2 Digits in the Winning ticket')
        print ('You have Won $20.')

    elif digits_count == 3:
        print ('You have 3 Digits in the Winning ticket')
        print ('You have Won $30.')

    else:
        print ('You Do Not have Any Digit in the Winning ticket')

# 12) Print the Footers of the Project
print ('\n********** End of the Project **********')
print ('***** Python Programmer: Ogar Haji *******')
```

Chapter 4 + Python Lab Assignment #4B (Due This Sunday) 100 Points

Name:_____ **CIS103 Python language + Wright College**



Python Programming Assignment #4B: Python Turtle Project to Draw Colored Lines Squares within Squares +++(Lab Assignment + Lab 4B)+++



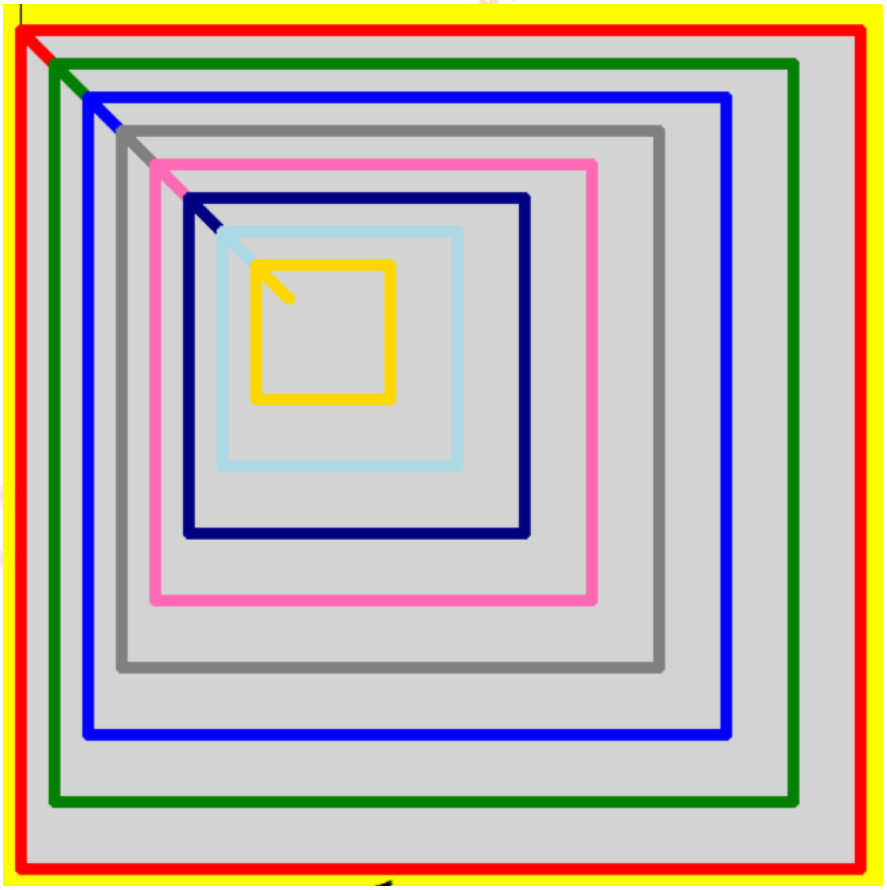
Lab Assignment Lab 4 B

Do the 12 Must Steps to Design, Code and Solve a Project in Python Language.

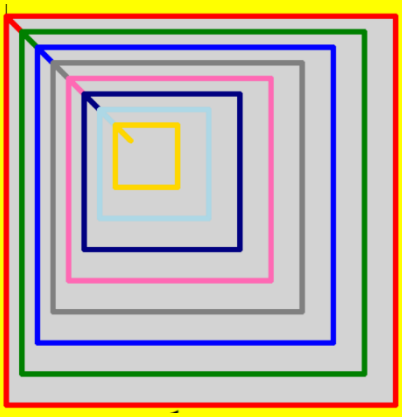
Do Lab Assignment 4 B

Do Steps 1 thru 7 in your Note Book or on Paper.

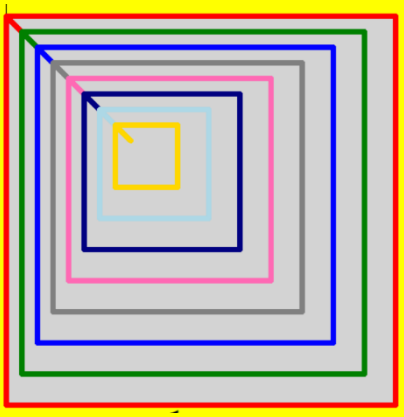
Note: When Modifying a Program, Do One Modification at a time.



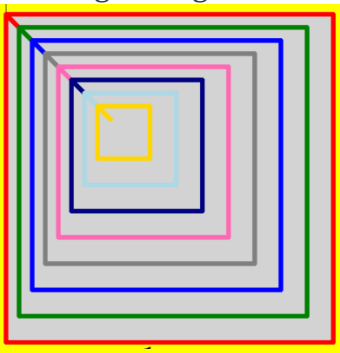
1)Design and Code a Python Turtle App and Save as TurtleDrawColoredLinesSquaresWithinSquaresLab4B.



```
# This Turtle project will draw a Square and change
# the pencolor to red and the fillcolor with orange
# Programmer: Ogar Haji
##############################################
# 1) Import the turtle module library.
import turtle

# 3) Change turtle shape to ‘turtle’
turtle.shape ('turtle')

# 4) Set new values to x, y and tell turtle to go to this location
x = 0
y = 200
turtle.goto (x,y)

# 5) Change the turtle speed to 1 which is very slow
turtle.speed (10)          # speed 10 is faster

# 6) Change the turtle pensize to size 7 which is thicker
turtle.pensize (7)

# 7) Change background color of the screen to yellow
turtle.bgcolor ('yellow')

# 8) Begin the turtle begin_fill to start the fill color
turtle.begin_fill ()

# 9) Change the turtle pencolor to hexadecimal red #ff0000
turtle.pencolor ('#ff0000')

# 10) Change the turtle fillcolor to gray
turtle.fillcolor ('lightgray')
```



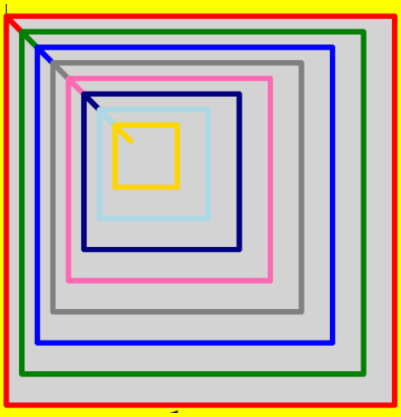
length = 500



11) Declare a List called 'colors_list' and
populate it with colors
colors_list = ('red','green','blue','gray','hotpink',
'navy', 'lightblue', 'gold')

12) Use **First Outer for** statement to loop thru the colors_list
for color in (colors_list):
turtle.pencolor (color)

13) Use **Second Inner for** statement with range (4) to loop 4 times
and draw the Square
for line in range (4):



1) Draw First line of the length 0f 300 pixels
turtle.forward (length)

2) Turn the turtle right by 90 degrees
turtle.right (90)

 # End of Second Inner for statement

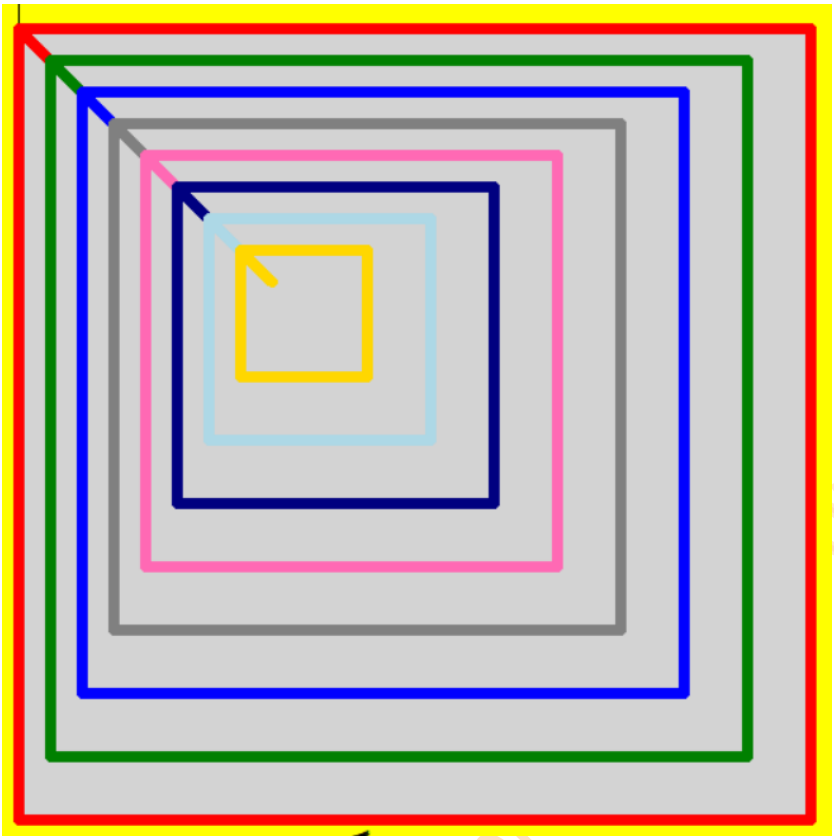
14) Subtract 60 pixels from the length
length = length - 60
x = x + 20 # Add 20 pixels to x axis (move turtle to Right)
y = y – 20 # Subtract 20 pixels from y axis (move turtle to up)
turtle.goto (x,y)

 # End of First Outer for statement

15) End the turtle fill to end_fill the colors
turtle.end_fill()

20) Inform turtle you are done and leave the window open.
turtle.done ()

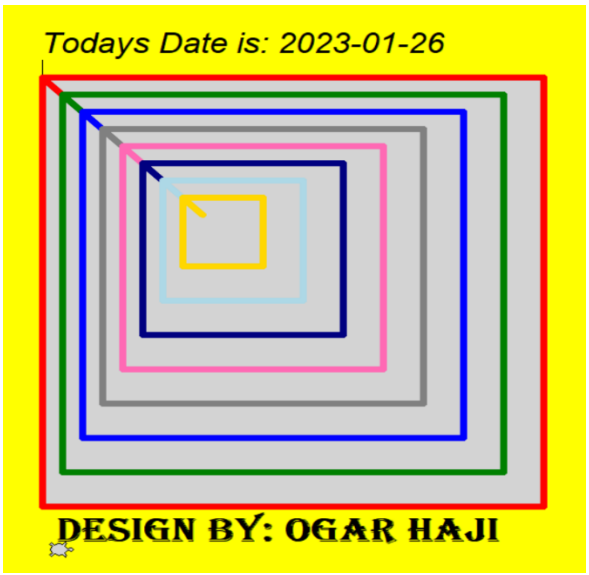
The final Project will look like the following:



Modify This Project to Do the Following Modifications: 30%

Modify this Python Project to do the following:

1) Add Computer Date at the Top of the image you draw as shown below:



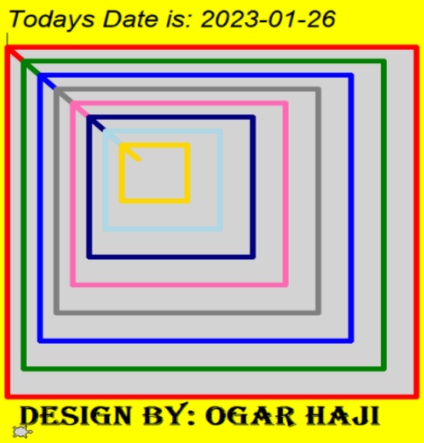
2) Import the following date class:
from datetime import date

2.1) Create an instance object called 'today' from module **date.today()**
today = date.today ()

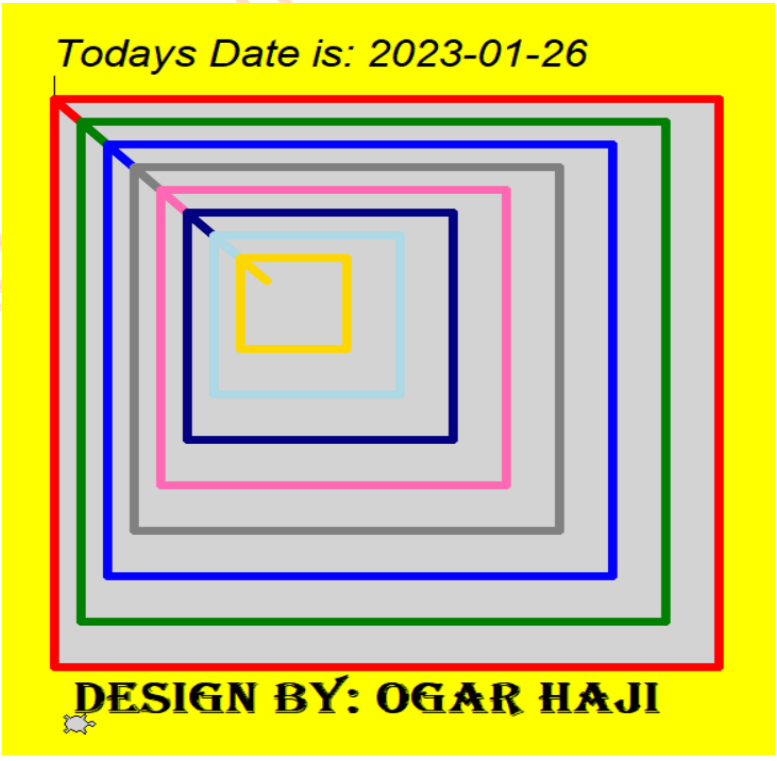
2.2) Define and initialize the x and y coordinates.
x = 0
y = 220

2.3) Tell turtle to go to location x, y
turtle.goto (x,y)

2.4) Write out today to output console
turtle.write ('Todays Date is: '+ str(today),
font = ('arial', 25, 'italic'))



2) Add and Write the Footer (Design by: Your Name' at the Bottom of the image you draw as shown below:



```
# 16) Raise the turtle pen turtle.penup() function
turtle.penup ()

# 17) Move the turtle to location (5, -350) by calling the function goto()
turtle.goto (15,-350)

# 18) Put the turtle pendown by calling function turtle.pendown()
turtle.pendown ()

# 19) Use the turtle.write() function to write your name
# and also change the font type, size and make it bold
turtle.pencolor('black')
turtle.write ('Design by: Ogar Haji', font = ('algerian', 30, 'bold'))
```

3) Upload the Final Modified Python Turtle project to Brightspace.

Note: Always **Upload** to **Brightspace the Modified Python Project:**
1) The **Microsoft Word Document** of the **Modified Python Project Code along with the Python Output Screen shots.**

Chapter 4 + Quiz 1 + Answers to Python Language Knowledge:

---

1. What is the output after executing the following Python code?

```
for i in range (5):          # this for in range loop will loop from 0 till 4
    print (i, end = '  ')
```

A. 12345 B. 0 1 2 3 4 5 **C. 0 1 2 3 4** D. 1 2 3 4 5 E. none

---

2. What is the output after executing the following Python code?

```
for num in range(1, 10, 2):    # remember 3 S with range (start:stop:step)
    print(num, end = ",")      # Starts with 1, increments by 2, till 10
```

A. 2,4,6,8,10 B. 2,4,6,8 **C. 1,3,5,7,9,** D. 1,3,7,9 E. none

---

3. What is the output after executing the following Python code?

```
number = 13
if (number % 2 == 0):   # % returns remainder if 0, means number is Even
    print ('Even')
else:
    print ("Odd")
```

A. even B. odd C. Even **D. Odd** E. error

---

4. What is the output after executing the following Python code?

```
a = 'Apple';  b = 'Orange'
x, y = b, a       # swap and assign b 'Orange' to x, assign 'Apple' to y
print (x, y, sep= "::")
```

A. Apple:Orange B. Apple::Orange **C. Orange:Apple** D. Orange::Apple

---

5. What is the output after executing the following Python code?

```
course = 'CIS 103 Python'
print(course [::] )        # here [::] means from start to end and step is 1
```

A. CIS 103 Python B. 'CIS 103 Python' C. CIS D. Python E. nothing

---

---

6. What is the output after executing the following Python code?

```
for i in range (10, 12, 1):      # start = 10, end = 12 (not inclusive), step = 1
    if i % 2 == 0:               # 10 % 2 is = 0 which means True
        print (f'{i} is Even')   # prints 10 is Even, and 11 is Odd
    else:
        print (f'{i} is Odd')
```

A. 10 is Even
11 is Odd
B. 10 is Odd
11 is Even
C. 10 is Even
D. 11 is Odd
E. error

---

7. What is the output after executing the following Python code?

```
a = 1 ;  b = 1
while a < 2:          # here, a = 1 which is less than 2 (True)
    while b < 2:      # here, b = 1 which is less than 2 (True)
        print (a, ":", b)       # prints 1: 1
        b += 1                  # b = 2
        a += 1                  # a = 2
```

A. 1 : 2
B. 2 : 2
C. 2 : 1
D. 1 : 1
E. error

---

8. What is the output after executing the following Python code?

```
language = 'Python'
i = 0
while i < len (language):      # length of 'Python' is 6
    print (i, end = '*')       # i is = 0, so it will print 0*
    i += 1                     # increment i by 1 which is = 1
```

A. 0*1*2*3*4*5*
B. 1*2*3*4*5*6*
C. 6
D. 5
E. error

---

9. What is the output after executing the following Python code?

```
print (2 * 4 % 7)   # from left to right, 2 * 4 = 8, then 8 % 7 = 1 remainder
```

A. 0
B. 1
C. 4
D. 8
E. error

---

10. What is the output after executing the following Python code?

```
for i in range (11):        # you can use any variable you want for 'i'
   if i == 0 :
       print ('Breaking out of for, i =', i )     # prints this and break out
       break
   else:
       print ('Still working in for.')
```

A. Breaking out of for, i = 0
B. Still working in for.
C. True
D. False

Answers:

Chapter 4 + Quiz 2 + Answers to Python Language Knowledge:

1. What is the output after executing the following Python code?

```
for i in range (5):          # Outer nested for 'i' loop from 0 to 4
    for j in range (5):      # Inner nested for 'j' loop from 0 to 4
        if j == 3:
            break            # break out of the Inner for loop when j == 3
print (i * j)                # after exiting for loop, i = 4, j = 3
```

A. 25 B. 20 C. 16 **D. 12** E. error

---

2. What is the output after executing the following Python code?

```
for i in range (5):          # Outer nested for loop from 0 to 4
    for j in range (5):      # Inner nested for loop from 0 to 4
        if j == 3:
            continue         # continue will continue the Inner for loop
print (i * j)                # after exiting for loop, i = 4, j = 4
```

A. 25 B. 20 **C. 16** D. 12 E. error

---

3. What is the output after executing the following Python code?

```
greeting = 'Good Morning'
for ch in greeting:
    if ch == 'o':            # ch is equal to G, so else will execute
        break
        print (ch)
    else:
        print ('Good Night')
```

A. G B. Go C. Good Morning **D. Good Night** E. error

---

4. What is the output after executing the following Python code?

```
sum = 0
for i in range (1, 5):
    sum  +=  i               # 0+1+2+3+4 = 10 sum = 10
print (sum)
```

A. 0 B. 1 **C. 10** D. 15 E. error

---

5. What is the output after executing the following Python code?

```
for i in range (1):          # i is = 0
    for j in range (1):      # j is = 0
        print (i, j)         # prints 0  0
```

A. 0 1 **B. 0 0** C. 1 0 D. 1 1 E. error

---

6. What is the output after executing the following Python code?

```
val = 8
while val > 0:       # result is True because 8 is greater than 0
    val = val - 2    # val = 8 - 2 = 6
    if val <= 5:     # False for val = 6, then another while and val = 4 True
                     # then it prints 4 and exit the while loop and hi will print
        print (val, end="")
        break
print ('hi')
```

A. 8hi **B. 4hi** C. 2hi D. hi E. error

---

7. What is the output after executing the following Python code?

```
x = 'python' ;  i = 0
while i < len (x):    # length of x = 6, so this is True
   i += 1             # i = 0 + 1 = 1,
   pass
print ('Value of i =', i)
```

A. Value of i = 0 B. Value of i =1 C. Value of i = 5 D. **Value of i = 6** E. error

---

8. What is the output after executing the following Python code?

```
str = 'Python language is easy to learn'
print ('easy' in str)       # The 'in' operator is used to find if 'easy' is in str
```

A. easy B. Python **C. True** D. False E. error

---

9. What is the output after executing the following Python code?

```
print (7 % 2 ** 3)          # Evaluate the ** first, 2 ** 3 = 8, then 7 % 8 = 7
```

A. 2 B. 3 **C. 7** D. 8 E. error

---

10. To Change the shape of the turtle to 'turtle' Shape, type the following:

turtle.shape ('turtle') (**True**/False)

Answers:

Chapter 4 + Quiz 3 + Answers to Python Language Knowledge:

---

1. What is the output after executing the following Python code?

programming_language = 'Python'

print (programming_language [-1]) # index **[-1]** returns last index char '**n**'

A. Nothing is printed **B. n** C. -1 D. P E. error

---

2. What is the output after executing the following Python code?

print (type (7 / 2)) # / is **float** division, so the **type** is **float**

A. 2.5 B. 2 C. <class 'int'> **D. <class 'float'>** E. error

---

3. What is the output after executing the following Python code?

import math

print (math.ceil (7.23)) # **math.ceil() raises up** value to **next integer 8**

A. 7 B. 7.23 C. 7.2 **D. 8** E. error

---

4. What is the output after executing the following Python code?

print (math.floor (7.23)) # **math.floor() lower down** value to integer **7**

A. 7 B. 7.23 C. 7.2 D. 8 E. error

---

5. What is the output after executing the following Python code?

print (round (7.23)) # because **.23** so it **rounds down** to integer **7**

A. 7 B. 7.23 C. 7.2 D. 8 E. error

---

6. What is the output after executing the following Python code?

print (round (7.53)) # because **.53** so it **rounds up** to integer **8**

A. 7 B. 7.23 C. 7.2 **D. 8** E. error

---

7. What is the output after executing the following Python code?

```
import random
random_number = random.randint (1, 3)
print (random_number)
```

A. 0 thru 3 **B. 1 thru 3** C. 0 thru 4 D. 1 thru 4 E. error

---

8. What is the output after executing the following Python code?

```
if 5 > 10:                  # This is False because 5 is Not > than 10
   print ('Python')
elif 8 != 9:                # This is True because 8 is Not equal to 9
   print ('language')      # this statement is executed, so language is printed
else:
   print ('is easy')
```

A. Python **B. language** C. is easy D. Python is easy E. error

---

9. What is the output after executing the following Python code?

```
if False:                   # False will skip to next elif
   print ('Chevy')
elif True:                  # True will execute this and print 'Ford'
   print ('Ford')
elif True:
   print ('Buick')
else:
   print ("Jeep")
```

A. Chevy **B. Ford** C. Buick D. Jeep E. error

---

10. What is the output after executing the following Python code?

```
number = int (input ('Enter an integer like 21: ') )   # user enters 21
print ( len (number) )    # len () is used to find the length of strings, lists
```

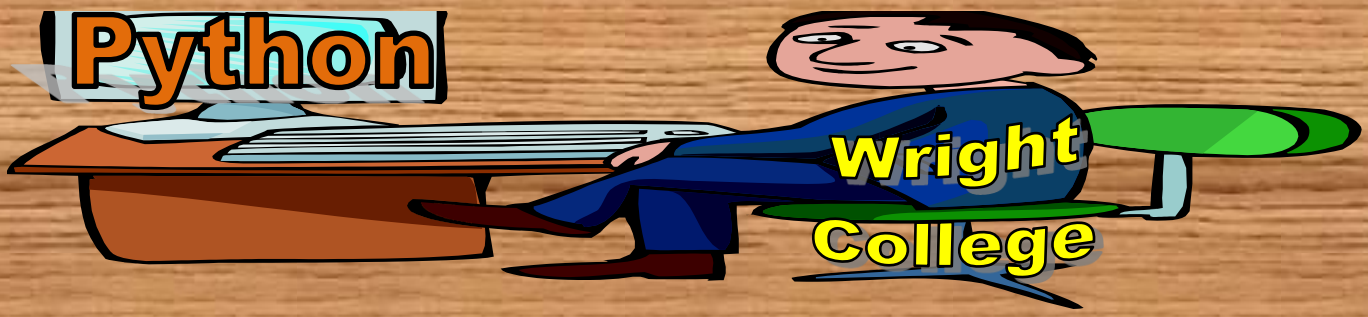
A. 21 B. 2 **C. TypeError: object of type 'int' has no len()** D. 3

Answers.

Wright College + Chapter 4

Loops and Iteration:
Using 'for' and 'while' Statements, Math functions and Random Numbers

CIS 103 Python Programming Language + Introduction to Computer Programming



'Hands-On' Mastering Computer Logic, Design and Programming Using Python Language

Written By:

OGAR HAJI

Master's Degree in Computer Science

DePaul University ✚ Chicago, Illinois + USA

Date Published: February 5, 2023