Topic 3 Writing the Scripting

(RE: ELEMENT 3.1, 3.2, 3.3, 3.4)

This is the last process of building a system. At this stage, the main characteristics of scripting language has been determined, the logical flow of the system has been designed, the big job will start from here. Learning a scripting language is a long journey, so there are five sub-topics under this topic 3.

Topic 3.1 Python Variables

Variables are an important programming concept. Many beginners have difficulty in understanding variables. So the trainers will provide some examples to support the PowerPoint, the learners will have class exercises to practice using variables.

The learners will be learning the following things in this topic:

- What is a variable
- How to declare a variable
- Types of variables
- Example of using variables
- Flowchart
- Pseudocode
- GrovePi kit for LED lights

WHAT ARE VARIABLES

Variables are nothing but reserved memory locations to store values. This means when you create a variable, you reserve some space in memory.

Based on the data type of a variable, the interpreter allocates memory and decides what can be stored in the reserved memory. Therefore, by assigning different data types to variables, you can store integers, decimals, or characters in these variables.

EXAMPLE OF VARIABLES

```
1  #testing variables
2
3  salary = 50000
4  tax_rate = 0.3
5  name = "John"
6
7  print (salary)
8  print (tax_rate)
9  print (name)
10
```

EXPLAIN THE SCRIPTING ABOVE:

Line 1 – comments, using # symbol for any other things such as the date and time, description of the scripting etc.

Line 2 – it is a gap to make code easier for readers to read, it is not necessary

Line 3 - create a variable called salary, set its value at 5000 and reserve its space in computer memory

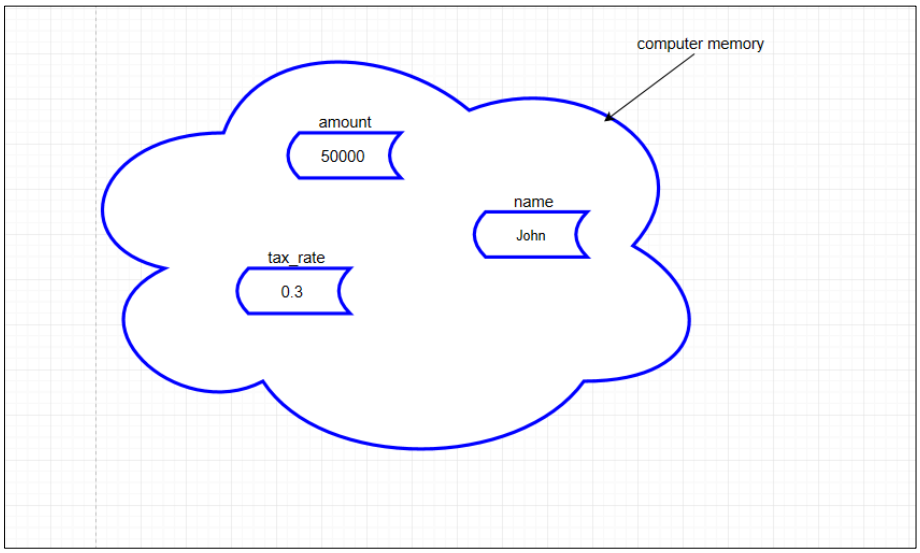
Line 4 - create a variable called tax_rate, which means to reserve a space in computer memory, notice this variable has two words that are connected by "_"

Line 5 - create a variable called name, which means to reserve a space in computer memory, notice this one is different in comparison with the two variables above, this one has double quotation. In fact, a pair of single quotation works the same.

Line 6 – space again

Line 7, 8, 9- display the value from the memory by using print key word.

THE GRAPHIC OF VARIABLES IN A COMPUTER MEMORY



Multiple assign the values

Unlike other scripting, Python has a simple way to assign the values to variables.

a=b=c = 100

The code above to assign value 100 to memory a, b and c

x, y, z = 100, 200, "John"

The code above to assign 100 to x, 200 to y name "John" to z

PYTHON OPERATOR

Math Operators (assuming a = 5, b = 2)

Operator	Description	Example
+	Adding values on either side of operators	a + b = 7
-	Subtracts right hand from left	a - b = 2
*	Multiple	a * b = 10
/	Decimal division	a / b = 2.5
//	Integer division	a // b = 2
%	Result of reminder number of the division	a % b = 1

Assignment Operators (assuming a = 5, b = 2)

Operator	Description	Example
=	Assign a value on right side to a variable on the left side	a = b, the final value of a is 2
+=	Same as a = a + b	a += b, the final value of a is 7
-=	Same as a = a – b	a -= b, the final value of a is 3
*=	Same as a = a * b	a *= b, the final value of a is 10
/=	Same as a = a / b	a /= b, the final value of a is 2.5
%=	Same as a = a % b	a %= b, the final value of a is 1
++	Increment by adding 1	a has value 6 in the memory after a++

BASIC BUILT-IN FUNCTIONS

Functions	Return(descriptions)	Examples
int()	Convert to an integer number	number = "5" number = Int(number) number is a numeric number 5

float()	Convert to a decimal number	number = “5.8” number = float(number) number is a numeric number 5.8
Input()	Keyboard input	name = input("Please enter your name ")

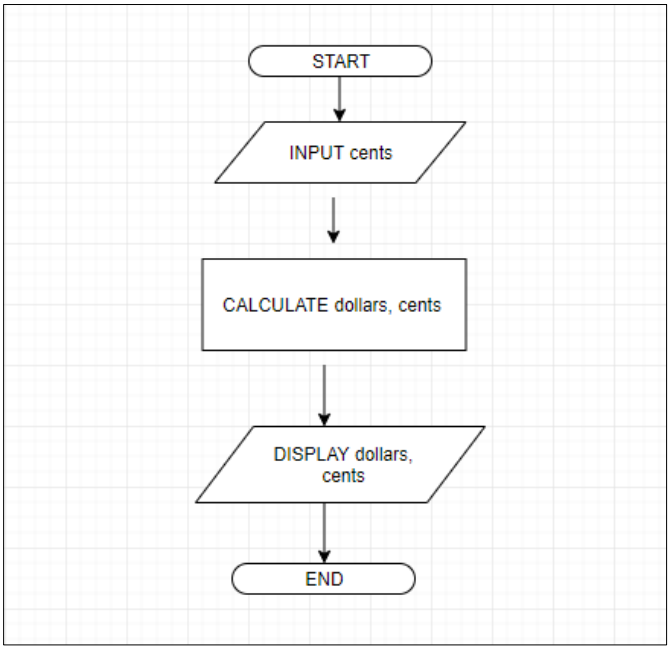
PYTHON CONNECTION SYMBOLS

Symbol	Description	Example
+	Connect string and string	firstname = "John" lastname = "Smith" fullname = firstname + lastname fullname has value “JohnSmith”
,	Connect String and Numeric	bank_balance = 3000 print ("your bank balance is ", bank_balance)

CASE EXAMPLE

You are creating a system that converts cents to dollars, the system allows user’s keyboard input of cents, and then displays a message that shows how much dollars converted and how much cents left.

FLOWCHART



PSEUDOCODE

START Terminal

INPUT cents

CALCULATE dollars = cents / 100, new_cents = cents%100

SCRIPTING

```
#cents to dollars program

cents = int(input("Please enter cents: "))

dollars = cents/100
new_cents = cents%100


print (str(cents) + " can be converted to $" + str(dollars) + " and " +
str(new_cents) + " cents")
```

GrovePI exercise 1

Important 1: you **MUST** have created a folder on the desktop(Don't delete), this folder will continue be used until the end of your Python class.

1. Follow the little green book on **page 17** to set up the hardware, and then follow the steps:
   Step 1: Menu > Programming > Python 2(or Python 3)

   Step 2: File > New

   Step 3: Type the following code:

```
# GrovePI LED Blink Exercise

import time
from grovepi  import *
from grove_rgb_lcd  import *

#connect the Grove LED to digital port D4
#connect the Grove LcD to digital port 12C
led = 4
setText("Hello  Everyone ")
setRGB(0, 255, 0)

while True:
        #Blink the light
        digitalWrite(led, 1)
        time.sleep(1)
```

   Step 4: Press F5 to debug, the LED light will be blinking when there is no error

Important 2: Please unpack your hardware, and put things together to hand it into your teacher before the end of Python class.

RESOURCE ON MOODLE:

- Python Variables (ppt)

LINKS FOR STUDENTS

- https://www.w3schools.com/python/python_variables.asp

https://www.youtube.com/watch?v=vKqVnr0BEJQ