COMPUTER SCIENCE

Basic Programming Concepts
CONSTRUCTS

Lesson Objectives

At the end of the entire lesson, you:

- Must be able to identify the three main programming constructs
- Should be able to understand the application of the three programming constructs
- Could be able to implement programs in Python programming language

Starter:
Knowing What You Know

1.

Go to: https://joinmyquiz.com

JOIN CODE:_____

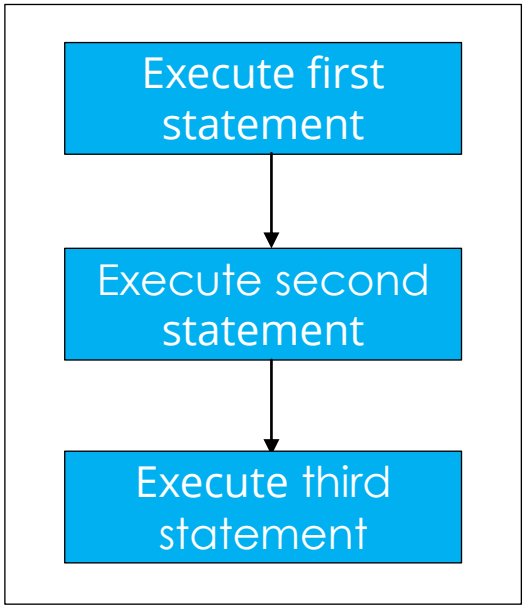
Join with your real name
Example: **Thanh 10Gx**

Introduction

- There are three types of basic programming constructs:
  a) Sequence
  b) Selection
  c) Iteration
- We will implement these programming constructs in Python.
- Python is an open-source, high-level programming language. Please log-in to your account in OnlineGB. You may go to FEBSTAR to follow the link to OnlineGB

Sequence

- Sequence as the name implies, is the execution of statements or functions one after the other.



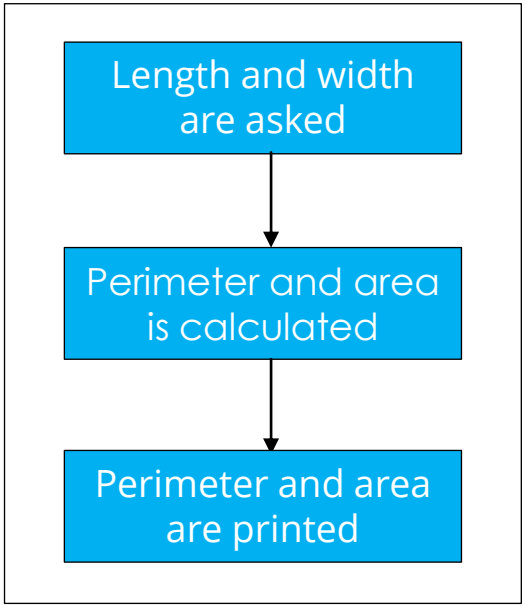
Sequence: Calculating area and perimeter of rectangle

Can you think of a solution to come up with this output?

```
Enter the length of a Rectangle: 4.38
Enter the width of a Rectangle: 5.13
Perimeter of Rectangle is: 19.02
Area of a Rectangle is: 22.47
```

Sequence: Calculating area and perimeter of rectangle


```
length = float(input('Enter the length of a Rectangle: '))
width = float(input('Enter the width of a Rectangle: '))
Perimeter = 2 * (width + length)
Area = width * height
print("Perimeter of Rectangle is: %.2f" %Perimeter)
print("Area of a Rectangle is: %.2f" %Area)
```



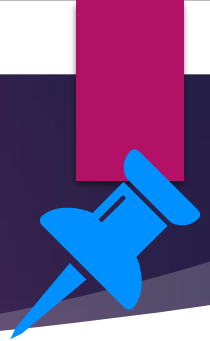
Sequence: Calculating area and perimeter of rectangle

```
length = float(input('Enter the length of a Rectangle: '))
width = float(input('Enter the width of a Rectangle: '))
Perimeter = 2 * (width + length)
Area = width * height
print("Perimeter of Rectangle is: %.2f" %Perimeter)
print("Area of a Rectangle is: %.2f" %Area)
```

Output:

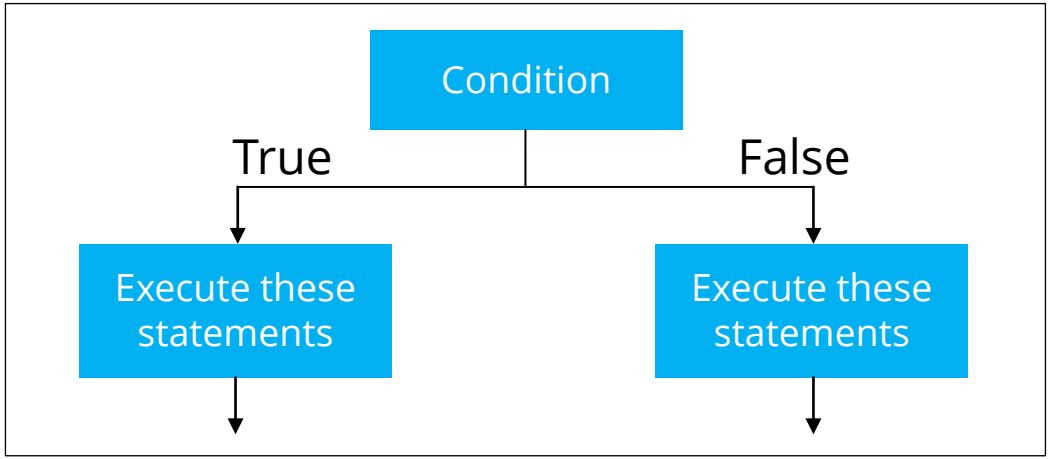
```
Enter the length of a Rectangle: 4.38
Enter the width of a Rectangle: 5.13
Perimeter of Rectangle is: 19.02
Area of a Rectangle is: 22.47
```

Selection

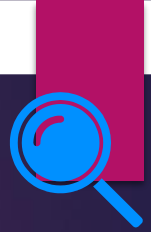


- Selection is used to execute only a particular set of statements if a condition is satisfied.
- The program may have many paths from the starting point to the end point.
- The path is chosen according to the condition satisfied.

Selection


Condition
True
False
Execute these statements
Execute these statements

Selection: nested if-else structure



- **Can you think of a solution to come up with either of this outputs?**

Output 1

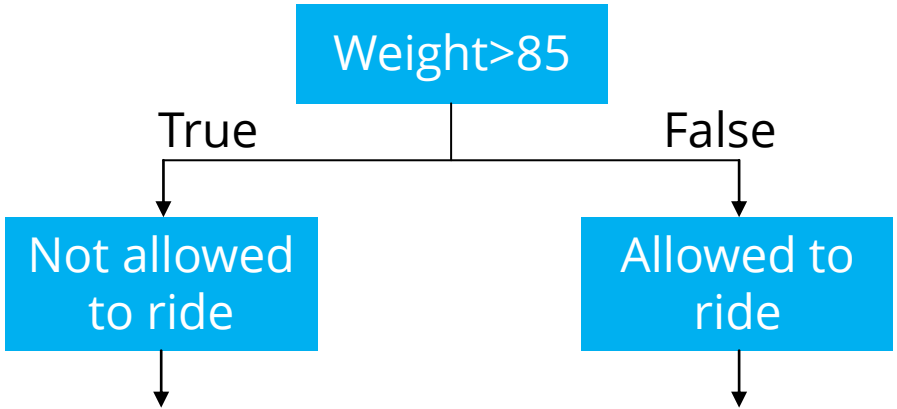
```
Enter age: 12
Enter height in cm: 132
Allow on to ride
```

Output 2

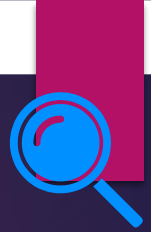
```
Enter age: 11
Enter height in cm: 129
Do not allow on to ride
```

Selection: if-else structure

```
weight=int(input("Enter weight in kg: "))
if weight>85:
    print("Do not allow on to ride")
else:
    print("Allow on to ride")
```



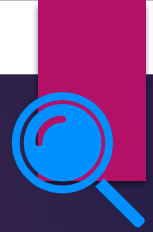
Selection: nested if-else structure



- In this program, two conditions are checked: age and height.
- A person must more than 10 years old and his/her height must be greater than 130 cm to be allowed on the ride.

```
age=int(input("Enter age: "))
if age<10:
    print("Do not allow on to ride")
else:
    height=int(input("Enter height in cm: "))
    if (height<130):
        print("Do not allow on to ride")
    else:
        print("Allow on to ride")
```

Selection: nested if-else structure



Output 1

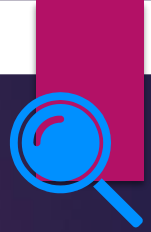
```
Enter age: 12
Enter height in cm: 132
Allow on to ride
```

Output 2

```
Enter age: 11
Enter height in cm: 129
Do not allow on to ride
```

```
age=int(input("Enter age: "))
if age<10:
    print("Do not allow on to ride")
else:
    height=int(input("Enter height in cm: "))
    if (height<130):
        print("Do not allow on to ride")
    else:
        print("Allow on to ride")
```

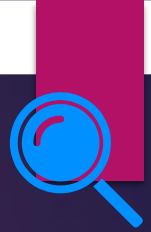
Selection: nested if-else structure



- An alternate approach to simplify this program is to use Boolean operators.
- AND can be used to check both height and age in a single statement.

```
age=int(input("Enter age: "))
if age<10:
    print("Do not allow on to ride")
else:
    height=int(input("Enter height in cm: "))
    if (height<130):
        print("Do not allow on to ride")
    else:
        print("Allow on to ride")
```

Selection: nested if-else structure



- An alternate approach to simplify this program is to use Boolean operators.
- AND can be used to check both height and age in a single statement.

```
age=int(input("Enter age: "))
height=int(input("Enter height in cm: "))
if age<10 & height<130:
    print("Do not allow on to ride")
else:
    print("Allow on to ride")
```

Selection: if… elif… else… statement

- Multiple conditions can be checked using this structure.

```
if expression1:
statement(s)
elif expression2:
statements(s)
elif expression3:
statement(s)
…
else:
statement(s)
```

Arithmetic operators in Python

Operator	Function
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Remainder of division (MOD)
//	Quotient obtained from division is rounded to the next whole number (DIV)
**	Exponentiation

Iteration

- Iteration is used to execute a particular set of statements repeatedly until a condition is satisfied. There are three ways to create iteration:

a) By counting how many times the statements are to be executed
b) By repeatedly executing the statements until a condition is true
c) By repeatedly executing the statements while a condition is true

Iteration: For structure

Can you think of a solution to come up with this Output?

```
1 times 4 is  4
2 times 4 is  8
3 times 4 is  12
4 times 4 is  16
5 times 4 is  20
6 times 4 is  24
7 times 4 is  28
8 times 4 is  32
9 times 4 is  36
10 times 4 is  40
```

Iteration: For structure

- Counts how many times the statements are to be executed.
- Program to print multiplication table of 4 from 1 to 10 is given.
- The value of k is incremented from 1 to 10 each time the print statement is executed.

```
for k in range(1,11):
    print (k, "times 4 is ", k*4)
```

Iteration: For structure

Output:

```
1 times 4 is  4
2 times 4 is  8
3 times 4 is  12
4 times 4 is  16
5 times 4 is  20
6 times 4 is  24
7 times 4 is  28
8 times 4 is  32
9 times 4 is  36
10 times 4 is  40
```

```
for k in range(1,11):
    print (k, "times 4 is ", k*4)
```

Iteration

- In some cases, it is not easy to determine the number of times the statements are to be executed.
- In such cases, we use the second and third approach.
- By executing statements until a condition is met or while a condition is met.
- Executing statements until a condition is met is achieved by using “Repeat until” or “Do while” loop.
- This type of loop is not available in Python but is available in other programming languages like Javascript.

Iteration: “While” loop

- **Can you think of a solution to come up with this output?**

```
Enter a number: 34
Enter a number: 56
Enter a number: 77
Enter a number: 90
Enter a number: 81
The total is:  338
The average is: 67.60
```

Iteration: “While” loop

- Executing statements while a condition is met is achieved by using “While” loop.
- Python code to calculate total and average of 5 numbers entered by user is given.

```
count=0
total=0
while (count<5):
    value=int(input("Enter a number: "))
    count=count+1
    total=total+value
print("The total is: ", total)
average=total/count
print("The average is: %.2f" %average)
```

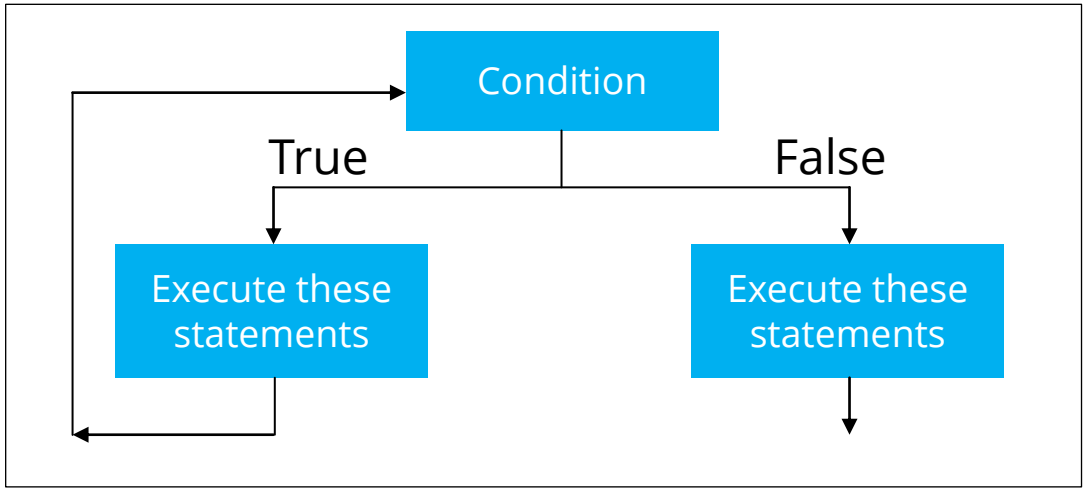
Iteration: “While” loop

- Output:

```
Enter a number: 34
Enter a number: 56
Enter a number: 77
Enter a number: 90
Enter a number: 81
The total is:  338
The average is: 67.60
```

```
count=0
total=0
while (count<5):
    value=int(input("Enter a number: "))
    count=count+1
    total=total+value
print("The total is: ", total)
average=total/count
print("The average is: %.2f" %average)
```

Iteration: “While” loop



Iteration: break statement

- Break statement in Python terminates the current loop and resumes operation from the next statement.

Let's review some concepts

Pair the descriptions to the words. Answer once for all

_____ is the execution of statements or functions one after the other.

_____ is used to execute only a particular set of statements if a condition is satisfied.

_____is used to execute a particular set of statements repeatedly until a condition is satisfied.

Iteration

Selection

Sequence

Let's review some concepts

Sequence

Sequence is the execution of statements or functions one after the other.

Selection

Selection is used to execute only a particular set of statements if a condition is satisfied.

Iteration

Iteration is used to execute a particular set of statements repeatedly until a condition is satisfied.

Plenary:
Knowing What you learned

Go to:
https://joinmyquiz.com

Join with your real name
Example: Thanh 10Gx

CLASS ACTIVITY

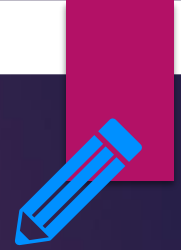
2.

Unleash your Power!

PYTHON PROGRAMMING

Activity-1

Duration: 20 minutes

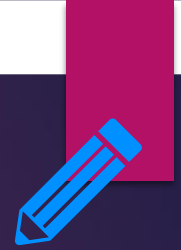


1. Write a Python program to convert temperature from Celsius to Fahrenheit and vice versa. Your program asks the user to input temperature either in Celsius, converts to Fahrenheit and prints the result.

Temperature (ºF)= Temperature (ºC) × 1.8 + 32

Activity-1

Duration: 20 minutes

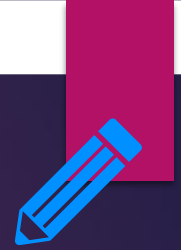


2. Write a Python program to find out the divisors of a number.
3. Write a Python program to find out whether a number is prime or not.

End of topic questions

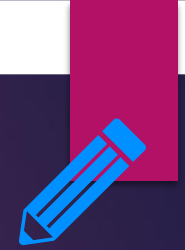
3.

Review: Think, Pair & Share



1. What are the three basic programming constructs?
2. What is sequence programming construct used for?
3. What is the syntax of if… else… statement in Python programming language?
4. How can multiple conditions be checked in if… else… statement?
5. What are the different iteration statements available in Python?
6. What is the syntax of “for loop” statement?

Review: Think, Pair, and Share



7. How is for loop different from while loop?
8. What is the syntax of while loop used in Python?
9. What is the purpose of break statement in loops?
10. Write a program to find LCM (Lowest common multiple) of two numbers.

CREDITS

Credit to text, images, graphics, videos, etc. owners for educational purposes only.

Thank you