

Control flow statements

It is important to make decisions in programming about how your code will be looked at. You may need to be selective, iterative or repetitive with the code statements. Python provides us some control flow statements as mentioned below to do this task. a) *if-elif-else* ladder to make decisions b) *for* loop for iteration and repetition c) *while* loop for selective repetition

In addition to this it also offers some keywords like *break*, *continue*, *pass* for further control.

if-else condition

For decision making *if-else* condition is the option python provides. To understand it's importance let us look at the example below:

```
# Example: Voter application validation
age = int(input("Dear applicant, please enter your age: "))
if age>18:
    print("Your application is accepted.")
else:
    print("You are underage now. Please come back later.")
```

In this example the program control does not flow over both the *print* statements. The selection here is based on a *Boolean* condition following *if* keyword. If the condition outcome is *True* it selects the statement under '*if block*' otherwise under '*else block*'. The same would have been very difficult without it.

Furthermore, if more such conditions participate in the selection process the keyword use is *elif*, which is like 'else if'.

Syntax:

```
if Boolean_condition:
    True statement
elif Boolean_condition:
    True statement
elif Boolean_condition:
    True statement
else:
    False statement
```

Note: *else* never takes a condition and is not compulsory in the above decision making ladder.

Ternary operator: A one-liner shortcut to if-else condition

Syntax:

if-else: **True if condition else False**

if-elif-else: `True_for_if if cond1 else True_for_elif if cond2 else False`

```
# Example: Equilateral triangle
sideA, sideB, sideC = -3,-3,-3
if sideA == sideB == sideC and sideA>0:
    print("It is equilateral")
else:
    print("Not an equilateral")

# Same code with ternary operator
print("It is equilateral") if sideA == sideB == sideC and sideA>0 else print("Not an equilateral")
```

for loop

for loop serves two important purposes:

- a) repeated execution of a code block
- b) iteration over sequence

Example

```
for i in range(3):
    print("Line 1")
    print("Line 2")
```

```
Line 1
Line 2
Line 1
Line 2
Line 1
Line 2
```

```
for i in "hello":
    print(i)
```

```
h
e
l
l
o
```

Deep-dive into for loop

1. 'for' is a keyword to define 'for loop'.
2. 'i' is just a variable name which can be replaced with any other variable not defined in the same scope.
3. 'in' is another keyword and membership operator which returns *True* if the operand is present in the sequence, otherwise it returns *False*.
4. Last part in the 'for loop' definition is a sequence object like string, list or tuple. `range()` function also defines a sequence object.

Syntax: `range(start=0,end=len(sequence),step=1)`

```
list(range(1,5))
```

```
[0,1,2,3,4]
```

```
list(range(1,10,2))  
[1,3,5,7,9]
```

while loop

It repeats over the code block as long as entry condition is *True* **Example**

```
n = 5  
i = 0  
while i<n:  
    print(i,"Hello world")  
    i += 1
```

```
0 Hello world  
1 Hello world  
2 Hello world  
3 Hello world  
4 Hello world
```

```
# Sometimes entry condition can be replaced by True to make an infinite loop  
while True:  
    print("Infinite loop")
```

Loop keywords: *break* and *continue*

break : It use used to break the loop at any given condition. As a result, the program control comes out of the loop. In case of nested loop *break* breaks only the innermost loop.

continue : It skips the current iteration of loop and rejoins from the next iteration. The program control does not leave the loop here, it merely skips the current iteration.

```
for i in "Python":  
    if i=="t":  
        break  
    print(i)
```

```
P  
Y
```

```
for i in "Python":  
    if i in "aeiou":  
        continue  
    print(i)
```

```
P  
Y  
t  
h  
n
```

pass is another keyword that makes no direct change to program control but can be used in place of

`pass` is another keyword that makes no direct change to program control but can be used in place of no definition.

```
for i in "Python":  
    pass
```

`pass` can also be used outside loop but not *break* and *continue*

```
try:  
    print(1/0)  
except:  
    pass
```

Practice Questions

1. Write a program to find if a given number is Even or Odd.

```
def isEvenOdd(number):  
    if number%2==0:  
        return "Even"  
    else:  
        return "Odd"
```

2. Write a program to check if a given number is divisible by 7 or 11.

```
def isDivisible(number):  
    if number%7==0 or number%11==0:  
        return True  
    else:  
        return False
```

3. Write a program to find the largest of three numbers.

```
def largestOfThree(a,b,c):  
    if a>=b and a>=c:  
        return a  
    elif b>=a and b>=c:  
        return b  
    else:  
        return c
```

4. Write a program to find the grade of a student.

```
def studentGrade(marks):  
    if marks>=70:  
        return "First class distinction"  
    elif marks>=60:  
        return "First class"  
    elif marks>=50:  
        return "Second class"  
    elif marks>=35:
```

```

    return "Third class"
else:
    return "Fail"

```

5. Write a program to print numbers from 1 to 100, but replace the number with "fizz" if it is divisible by 3 and by "buzz" if it is divisible by 4 and "fizzbuzz" if it is divisible by both.

```

def fizzbuzz():
    for i in range(1,101):
        if i%3==0 and i%5==0:
            print("fizzbuzz")
        elif i%3==0:
            print("fizz")
        elif i%5==0:
            print("buzz")
        else:
            print(i)

```

```

# one liner answer for fizz buzz
for i in range(1,101): print("fizzbuzz") if i%3==0 and i%5==0 else
print("fizz") if i%3==0 else print("buzz") if i%5==0 else print(i)

```

6. Write a program to find the count of vowels in a given country name.

```

def countVowels(name):
    count = 0
    for i in name:
        if i in "aeiouAEIOU":
            count += 1
    return count

```

7. Write a program to delete vowels from a given string.

```

def deleteVowels(str1):
    str2 = ""
    for i in str1:
        if i in "aeiouAEIOU":
            continue
        str2 += i
    return str2

```

8. Write a program to print the squares of two upto a given number.

```

def twoSquares(number):
    i = 1
    while i<=number:
        print(2**i)
        i += 1

```

9. Write a program to break the string on first vowel.

```
def firstVowel(str1):
    str2 = ""
    for i in str1:
        if i in "aeiouAEIOU":
            break
    str2 += i
    return str2
```

10. Brain teaser: Car Puzzle

Consider all four digit car numbers of which one number is True on all the below conditions:

- a) Last digit is double the first digit, like 1--2
- b) Last two digits are double the first two, like 1122
- c) Middle digits are same, like -11-

Find the number.

```
for i in range(1000,10000):
    if i%10 == 2*(i//1000) and i%100 == 2*(i//100) and i//100%10 == i%100//
10:
    print("The car number is", i)
```