



VIETCODE ACADEMY INTRODUCTION TO PROGRAMMING

Class 3: Loop





TEACHER INFORMATION

TEACHER



TEACHING ASSISTANT





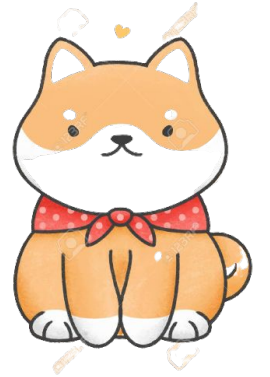
Loop and why?

The job of the computer not only handles something like printing a sentence like in the previous lessons, but it can also print out 1000 sentences.

However we are human we cannot write 1000 print

=> The loop will solve this problem

=> Loop is a way to execute one or a set of statements repeatedly





Loop Types

In programming, there are 2 main types of loops that we need to pay attention to:

1. For loop
2. While loop





For loop



Let's come to a simple question of printing the numbers from 1 to 10, how will using for loop help us solve this problem?

```
for i in range(1,11):  
    print(i)
```

In the for loop variable *i* has run from 1 to **before** 11, so when printing variable *i* it means we have printed the numbers from 1 to 10 in turn. Now the terminal will have numbers from 1 to 10.



Structure of for loop

As seen in the above example, the for loop has the following structure:

```
for variable in range (start, end, step):  
    code
```

Now the variable will run from start to **before** end

In each run the code will execute.

After the end of the code, the variable will add a step and if it is less than the end, the code will continue to execute

* **Step is defaulted to 1, so when not declaring step, it will understand each step of running variable plus 1 as in the above example.**





While loop

Let's go back to the question of printing numbers from 1 to 10, how will using while loop help us solve this problem?



```
i=1
while i<11:
    print(i)
    i=i+1
```

First the variable *i* has a value of 1, the while loop will check if the variable *i* is less than 11. If less then *i* will be printed and *i* incremented by 1. Then the while loop will be repeated until the condition is not satisfied.



Structure of while loop

As seen in the above example, the while loop has the following structure:

while conditional statement:

code

When conditional statement in while results True, the code executes

After the code finishes, the conditional statement will be checked again and if the result is True, the code will continue to execute until the result is False.

* Note that unlike for loop, variable will run from start to end, in while loop we need to change the variable so that conditional statement results in False in the future. If the conditional statement always returns True, the code will run forever.





Important syntax in loop

There are 2 important syntaxes to remember in the loop

1. break:
 - When there is break syntax in a loop, the following statements and even all subsequent loops will not run
2. continue:
 - When there is continue syntax in a loop, just skip the statements below it and jump to the next loop





Example

```
for i in range(1,11):  
    if i ==5:  
        break  
    print(i)
```

When i runs from 1 to 5,
there is a break statement so
the terminal only prints the
numbers from 1 to 4



```
for i in range(1,11):  
    if i ==5:  
        continue  
    print(i)
```

When i runs from 1 to 5, it will
skip the below statement
(print 5), so the terminal
prints the numbers from 1 to
10 (except 5)



PRACTICE WITH INCLASS3