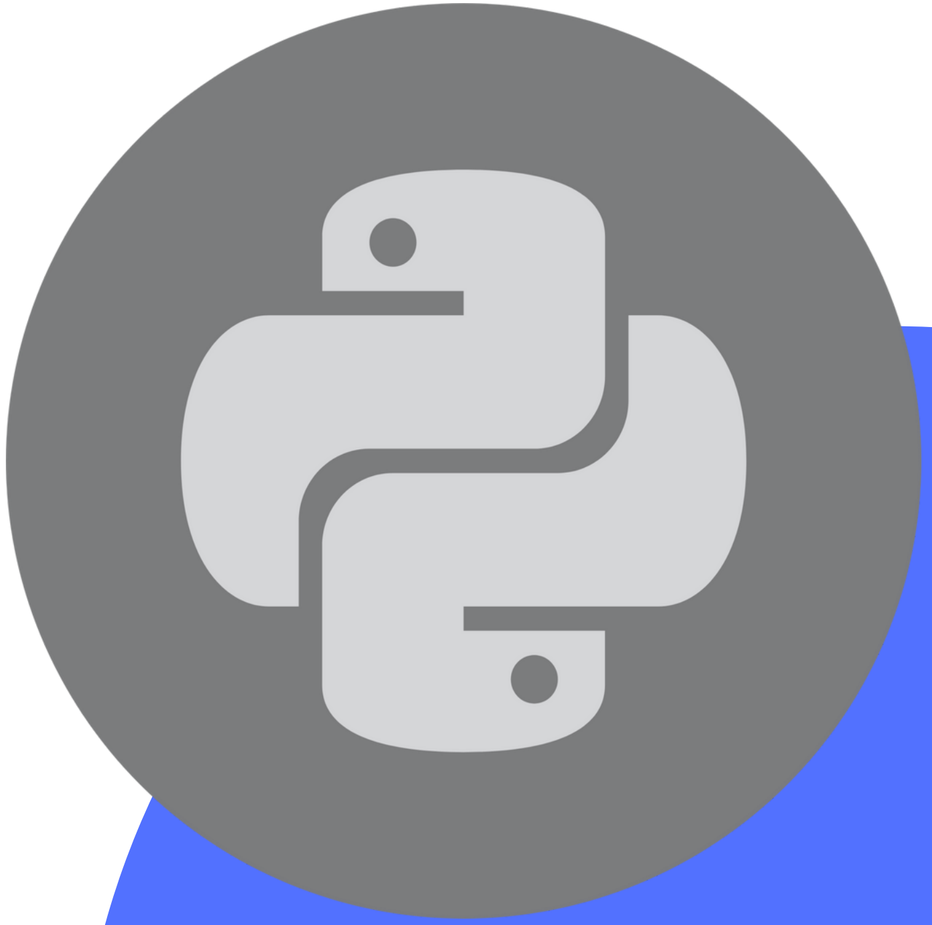
PYTHON COURSE

ENTRY LEVEL



Basics of programming in Python 3.10

This course will cover part of the arguments found in PCEP™ – Certified Entry-Level Python Programmer Certification

CONTROL FLOW: CONDITIONAL BLOCKS AND LOOPS

PYTHON

LOOPS: WHILE LOOP

Used to perform actions (execute a block of code) while a certain condition will be True:

```
counter = 0
condition = counter < 3

while condition:
    print("Hello")
    counter += 1
    condition = counter < 3
```

indentation

condition to be checked at every iteration

while keyword

PYTHON

LOOPS: WHILE LOOP

ATTENTION: create some logics to stop loop execution or it will run forever!

this condition is always True

```
while True:
    print("Hello")
```

The above example produces an infinite loop! so your pc will continue printing “Hello” while you manually stop the program

PYTHON

LOOPS: WHILE LOOP

- Using break in while loops will stop the cicle and the program will exit the loop block

```
counter = 0

while True:
    if counter < 3:
        print("Hello")
        counter += 1
    else:
        break
```

incrementing counter in every iteration of the while loop

in every iteration, if the break instruction is executed, the cicle stops and program will exit the loop

PYTHON

LOOPS: WHILE LOOP

- Using continue in while loops will skip the iteration

```
counter = 3

while counter > 0:
    if target != position:
        counter -= 1
    else:
        continue
```

context: (position is a fixed value, target is from user input at every iteration)

if this instruction is executed,
we'll skip the iteration and continue to the next one

PYTHON

PRACTICAL USE OF WHILE

```python
attempts = 0
user = "aaa"
password = "abc"


while attempts <= 3:
    select_user = input("insert username: ")
    select_pass = input("insert pass")

    if select_user == user and select_pass == password:
        print("Welcome back")
        break
    else:
        print("Wrong user or pass")
        attempts += 1
        remaining = 3 - attempts
        print(f"{remaining} attempts remaining")
```

EXERCISES