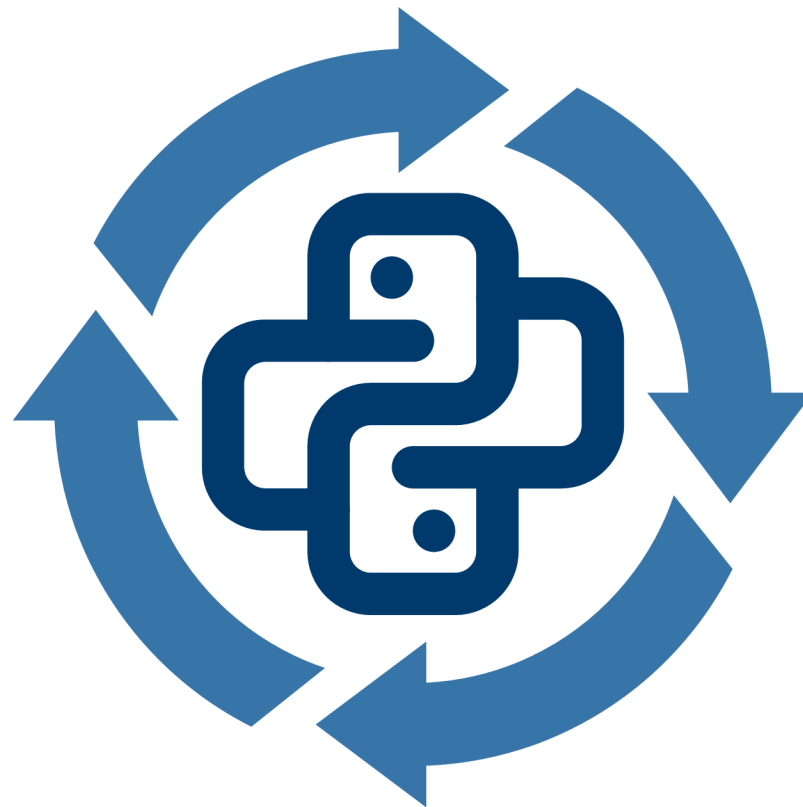


SINCE 2006

FUTURE **VISION**

MASTERING FOR LOOPS IN PYTHON



SINCE 2006

FUTURE **VISION**

WHAT IS A FOR LOOP?

A FOR LOOP ALLOWS YOU TO ITERATE OVER A SEQUENCE (LIKE A LIST, TUPLE, OR STRING) AND EXECUTE A BLOCK OF CODE MULTIPLE TIMES!



SINCE 2006

FUTURE VISION

BASIC SYNTAX

HERE'S THE BASIC SYNTAX OF A FOR LOOP:



```
for item in iterable:
```

```
    # Code block to execute
```



EXAMPLE 1 - ITERATING THROUGH A LIST

LET'S ITERATE THROUGH A LIST OF FRUITS:

```
fruits = ["apple", "banana", "cherry"]  
for fruit in fruits:  
    print(fruit)
```

Output:

Copy code

apple

banana

cherry



EXAMPLE 2 - USING RANGE 0

YOU CAN ALSO USE FOR LOOPS WITH THE RANGE() FUNCTION:

```
for i in range(5):
```

```
    print(i)
```

Output:

Copy code

0

1

2

3

4



SINCE 2006

FUTURE VISION

NESTED FOR LOOPS

YOU CAN NEST FOR LOOPS TO WORK WITH
MULTIDIMENSIONAL DATA!



```
for i in range(3):  
    for j in range(2):  
        print(f"i: {i}, j: {j}")
```

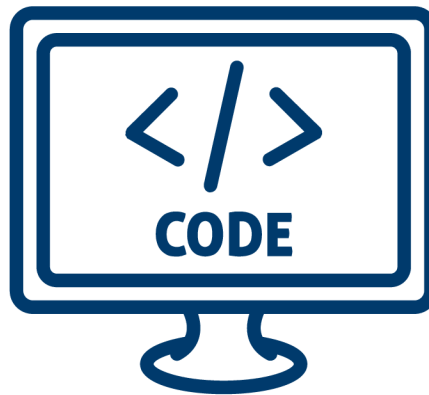


SINCE 2006

FUTURE **VISION**

PRACTICAL EXAMPLE - SUMMING NUMBERS

WANT TO SUM NUMBERS IN A LIST? A FOR LOOP CAN HELP!



```
numbers = [1, 2, 3, 4, 5]
total = 0
for num in numbers:
    total += num
print(total) # Output: 15
```

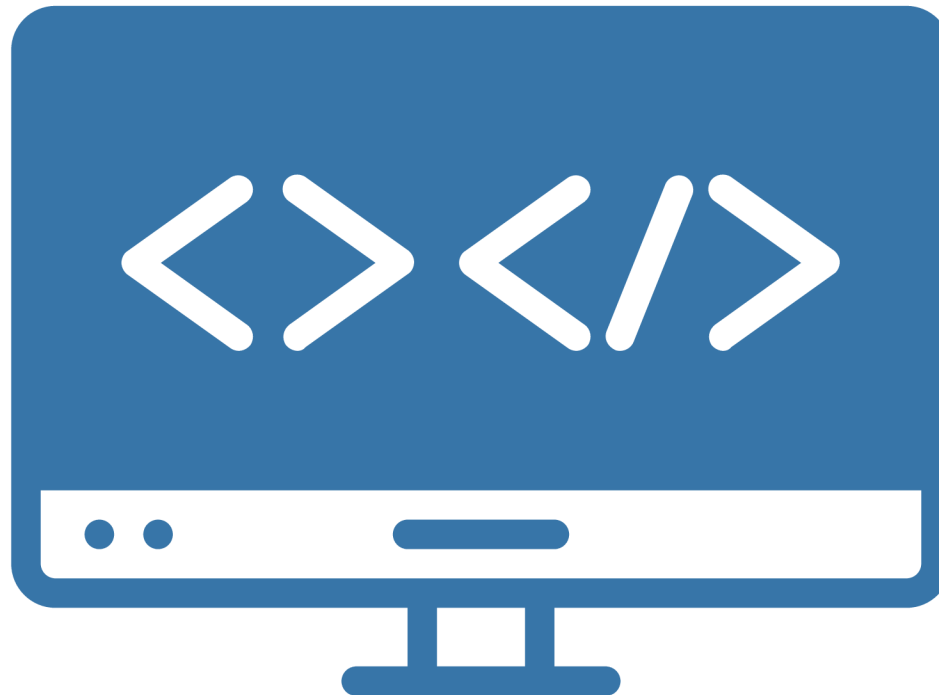


SINCE 2006

FUTURE **VISION**

CONCLUSION

FOR LOOPS ARE ESSENTIAL FOR REPETITIVE TASKS IN PYTHON. THEY HELP YOU EFFICIENTLY HANDLE DATA AND AUTOMATE PROCESSES!



Since 2006

FUTURE

VISION

Computer institute

Follow Us



+91 87991 41678



futurevisioncomputers.com



PAL, CITYLIGHT, VESU