

1811/2807/7001ICT

Programming Principles

School of Information and Communication Technology
Griffith University

Trimester 1, 2024

13 Definite Loops With for

Python's `for` statement is its form of definite loop.

It iterates over sequences of values.

13.1 The for statement

The basic `for` loops in languages like C and Java are just slightly rearranged `while` statements.

C++, Java, and Javascript have different kinds of `for` loops, for looping through the elements of an array or some other container.

Python only offers this latter kind of `for` loop.

Template:

```
for variable-name in sequence:  
    loop body statement(s)
```

The *sequence* will be an object that contains a sequence of values that the named variable will be assigned each time the loop body executes.

13.1.1 for loops with strings

A string is a sequence, and we can process it character by character with a for loop.

```
# file: str1.py  
# Print a string vertically.
```

```
s = "abc"  
for c in s:  
    print(c)
```

```
$ python3 str1.py  
a  
b  
c  
$
```

13.1.2 for loops with ranges

To loop a fixed number of times, we need an object containing a sequence of values, for example 1 to 10.

The class `range` makes such sequences.

A range is made by calling its constructor. Examples:

<i>constructor call</i>	<i>range</i>	<i>explanation</i>
<code>range(5)</code>	0, 1, 2, 3, 4	The range starts at starts at 0, ends at 5 (non inclusive).
<code>range(1, 5)</code>	1, 2, 3, 4	The range starts at starts at 1 (inclusive), ends at 5 (non inclusive).
<code>range(0, 10, 3)</code>	0, 3, 6, 9	The range starts at starts at 0 (inclusive), ends at 10 (non inclusive), in steps of 3.

Examples in scripts:

```
# file: for1.py
# Print the numbers from 0 to 4 on one line.
```

```
for i in range(5):
    print(i, end = " ")
print()
```

```
$ python3 for1.py
0 1 2 3 4
$
```

```
# file: for2.py
# Print the numbers from 1 to 4 on one line.

for i in range(1, 5):
    print(i, end = " ")
print()
```

```
$ python3 for1.py
1 2 3 4
$
```

Examples in scripts:

```
# file: for2.py
# Print the numbers from 0 to 9 in steps of 3.

for i in range(0, 10, 3):
    print(i, end = " ")
print()
```

```
$ python3 for1.py
0 3 6 9
$
```


The steps can be negative, so the range returns values in descending order:

```
# file: for4.py
# Print the numbers from 10 down to 1 on one line.

for i in range(10, 0, -1):
    print(i, end = " ")
print()
```

```
$ python3 for1.py
10 9 8 7 6 5 4 3 2 1
$
```

13.2 In-class Examples

Solve these problems as a class.

Problem: Prompt for and read a number. Print that many hashes (#) in a line.

```
Enter a number: 5  
#####
```

Problem: Prompt for and read numbers of rows and columns. Print those many rows and columns of hashes.

```
Enter rows: 2  
Enter columns: 5  
#####  
#####
```

Problem: Prompt for and read a string. Print the number of characters in the string that are decimal digits.

<pre>Enter a string: To do: 1. Party like it's 1999. Number of digits = 5</pre>

Problem: Previously we computed the average of a class's test marks using a `while` loop and the sentinel pattern to read the marks.

Write a new program that instead of using a negative number to end input, asks how many marks there are first, then reads them using a `for` loop.

Section summary

This section covered:

- the `for` statement; and
- the `range` class for generating sequences of values.