Lists, for loops

Python basics

Kunal Khurana

2023-10-06

Table of contents

learning outcomes

1. How to work through a list using a for loop, how Python uses indentation to structure a program, and how to avoid some common indentation errors.
2. Numerical lists, as well as a few operations you can perform on numerical lists.
3. Slice a list to work with a subset of items and how to copy lists properly using a slice
4. Tuples (which provide a degree of protection to a set of values that shouldn't change) and how to style your increasingly complex code to make it easy to read.

```
# looping through list

courses = ['Agriculture', 'Soil Science', 'Francisation', 'génie civil']
for course in courses:
    print(course)
```

```
Agriculture
Soil Science
Francisation
génie civil
```

Remarks-

1) don't forget indentation in second line
2) don't forget to include two dots
3) with for loop, we associated course with courses, The output gets printed in seperate lines of code
4) generic- for item in list_of_items

```
for course in courses:
    print(f"{course.title()}, was awesome!")
```

```
Agriculture, was awesome!
Soil Science, was awesome!
Francisation, was awesome!
Génie Civil, was awesome!
```

```
# adding a line in front
for course in courses:
    print (f"I look forward to becoming more smart in {course.title()}.")
```

```
I look forward to becoming more smart in Agriculture.
I look forward to becoming more smart in Soil Science.
I look forward to becoming more smart in Francisation.
I look forward to becoming more smart in Génie Civil.
```

```
# adding an empty line after
for course in courses:
    print (f"I will add empty line after {course.title()}.\n")
```

```
I will add empty line after Agriculture.

I will add empty line after Soil Science.

I will add empty line after Francisation.

I will add empty line after Génie Civil.
```

```
# way to write multiple lines of code using for loop
for course in courses:
    print(f"{course.title()} was a great subject")
    print(f"I got to learn a lot about {course.title()}.")
    print("Thanks awesome people who came to my life as mentors and changed it.\n")
```

```
Agriculture was a great subject
I got to learn a lot about Agriculture.
Thanks awesome people who came to my life as mentors and changed it.

Soil Science was a great subject
I got to learn a lot about Soil Science.
Thanks awesome people who came to my life as mentors and changed it.

Francisation was a great subject
I got to learn a lot about Francisation.
Thanks awesome people who came to my life as mentors and changed it.
```

```
Génie Civil was a great subject
I got to learn a lot about Génie Civil.
Thanks awesome people who came to my life as mentors and changed it.
```

Numerical lists

1. we use range function for that

```
for value in range(1,6):
    print(value)
```

```
1
2
3
4
5
```

```
# printing a list of numbers
for value in range(1,6):
    print(list[value])
```

```
list[1]
list[2]
list[3]
list[4]
list[5]
```

```
# printing list of numbers using list function
numbers = list(range(1,6))
print(numbers)
```

```
[1, 2, 3, 4, 5]
```

```
# printing even numbers
even_numbers = list(range(0,11,2))
print(even_numbers)
```

```
[0, 2, 4, 6, 8, 10]
```

```
squares = []
for value in range(1,11):
    square = value ** 2
    squares.append(square)
    print(squares)
print (squares)                          #difference between this and previous, wow!

# indentation is important
```

```
[1]
[1, 4]
[1, 4, 9]
[1, 4, 9, 16]
[1, 4, 9, 16, 25]
[1, 4, 9, 16, 25, 36]
[1, 4, 9, 16, 25, 36, 49]
[1, 4, 9, 16, 25, 36, 49, 64]
[1, 4, 9, 16, 25, 36, 49, 64, 81]
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

```
print(squares)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

```
# precise way

sq_numbers = []
for a in range(1,11):
    sq_numbers.append(a**2)
print(sq_numbers)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

List comprehensions

1. extended form of a for loop (result we need in front, followed by for loop)
2. combines for loop and creates new elements in one line

```
sq = [value**2 for value in range(1, 11)]
print(sq)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

```
cubes = [value**3 for value in range(1,11)]
print(cubes)
```

```
[1, 8, 27, 64, 125, 216, 343, 512, 729, 1000]
```

Looping through a slice

```
universities = ['punjabi', 'pau', 'laval', 'Mcgill']
print("here are the list of universities, I studied:")
for university in universities:
    print(university.title())
```

```
here are the list of universities, I studied:
Punjabi
Pau
Laval
Mcgill
```

Tuples

1. they are immutable lists
2. uses parenthesis instead of square brackets
3. To define a tupel with one element, you need a comma behind

```
Buffet= ('Aloo gobhi', 'matar paneer', 'parantha', 'raita', 'roti',)
print(Buffet)
```

```
('Aloo gobhi', 'matar paneer', 'parantha', 'raita', 'roti')
```

```
Buffet[1]= ('pappad')
```

```
print("Buffet:")
for item in Buffet:
    print(item)
```

```
Buffet:
Aloo gobhi
matar paneer
parantha
raita
roti
```

```
New_buffet = ('raita', 'parantha', 'butternan', 'allogobhi', 'Kasaurimethi')
print('New_buffet:')
for item in New_buffet:
    print(item)
```

```
New_buffet:
raita
parantha
butternan
allogobhi
Kasaurimethi
```

Respecting community; some guidelines 1. Indentation- PEP 8 2. line length- 79 characters long 3. Blank lines - 5 lines of code follwoed by a blank line is suggested