

Ling 555 — Programming for Linguists

Python - List and Tuples

Robert Albert Felty

Speech Research Laboratory
Indiana University

Sep. 17, 2008

Outline

List basics

List methods

tuples

1 List basics

- indexing
- slicing
- Lists of lists
- mutable
- operations

2 List methods

- len, min, and max
- Append, extend, insert, remove, pop
- sort

3 tuples

Indexing

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Zero comes first

Python starts all indices with 0.

Practice

- ① How do you print just the first letter of “hello”?
- ② How do you print just the last letter of “hello”?
- ③ Create a list *foo*, with the following values: 25, 68, “bar”, 89.45, 789, “spam”, 0, “last item”

Indexing

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Zero comes first

Python starts all indices with 0.

Practice

- 1 How do you print just the first letter of “hello”?
`print('hello'[0])`
- 2 How do you print just the last letter of “hello”?
- 3 Create a list *foo*, with the following values: 25, 68, “bar”, 89.45, 789, “spam”, 0, “last item”

Indexing

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Zero comes first

Python starts all indices with 0.

Practice

- 1 How do you print just the first letter of “hello”?
`print('hello'[0])`
- 2 How do you print just the last letter of “hello”?
`print('hello'[-1])`
- 3 Create a list *foo*, with the following values: 25, 68, “bar”, 89.45, 789, “spam”, 0, “last item”

Indexing

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Zero comes first

Python starts all indices with 0.

Practice

- 1 How do you print just the first letter of “hello”?
`print('hello'[0])`
- 2 How do you print just the last letter of “hello”?
`print('hello'[-1])`
- 3 Create a list *foo*, with the following values: 25, 68, “bar”, 89.45, 789, “spam”, 0, “last item”
`foo = [25, 68, 'bar', 89.45, 789, 'spam', 0, 'last item']`

Slicing

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Definition

A list slice takes part of a list. A slice `list[i:j]` starts at the i_{th} index, and goes up to (**but does not include**) the j_{th} index.

Slicing practice

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Practice

HINT: remember that indices start at 0

- 1 Print the 1st to 3rd item in the list *foo*
- 2 Print the 3rd to last item in the list *foo*
- 3 Print the 2nd to the 2nd to last item in the list *foo*
- 4 Copy the entire *foo* list to a new list named *bar*

Slicing practice

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Practice

HINT: remember that indices start at 0

- 1 Print the 1st to 3rd item in the list *foo*
`print(foo[:3])`
- 2 Print the 3rd to last item in the list *foo*
- 3 Print the 2nd to the 2nd to last item in the list *foo*
- 4 Copy the entire *foo* list to a new list named *bar*

Slicing practice

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Practice

HINT: remember that indices start at 0

- 1 Print the 1st to 3rd item in the list *foo*
`print(foo[:3])`
- 2 Print the 3rd to last item in the list *foo*
`print(foo[2:])`
- 3 Print the 2nd to the 2nd to last item in the list *foo*
- 4 Copy the entire *foo* list to a new list named *bar*

Slicing practice

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Practice

HINT: remember that indices start at 0

- 1 Print the 1st to 3rd item in the list *foo*
`print(foo[:3])`
- 2 Print the 3rd to last item in the list *foo*
`print(foo[2:])`
- 3 Print the 2nd to the 2nd to last item in the list *foo*
`print(foo[1:-1])`
- 4 Copy the entire *foo* list to a new list named *bar*

Slicing practice

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Practice

HINT: remember that indices start at 0

- 1 Print the 1st to 3rd item in the list *foo*
`print(foo[:3])`
- 2 Print the 3rd to last item in the list *foo*
`print(foo[2:])`
- 3 Print the 2nd to the 2nd to last item in the list *foo*
`print(foo[1:-1])`
- 4 Copy the entire *foo* list to a new list named *bar*
`bar=foo[:]`

Lists can contain other lists

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

practice

- 1 Create a new list *lol* which contains the lists *foo* and *bar*
- 2 Print the first item of *lol*
- 3 Print the second item of the first list in *lol*
- 4 Print the third item to the last item of the second list in *lol*

Lists can contain other lists

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

practice

- 1 Create a new list *lol* which contains the lists *foo* and *bar*

```
lol=list()  
lol.append(foo)  
lol.append(bar)
```
- 2 Print the first item of *lol*
- 3 Print the second item of the first list in *lol*
- 4 Print the third item to the last item of the second list in *lol*

Lists can contain other lists

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

practice

- 1 Create a new list *lol* which contains the lists *foo* and *bar*

```
lol=list()  
lol.append(foo)  
lol.append(bar)
```
- 2 Print the first item of *lol*

```
print(lol[0])
```
- 3 Print the second item of the first list in *lol*
- 4 Print the third item to the last item of the second list in *lol*

Lists can contain other lists

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

practice

- 1 Create a new list *lol* which contains the lists *foo* and *bar*

```
lol=list()  
lol.append(foo)  
lol.append(bar)
```
- 2 Print the first item of *lol*

```
print(lol[0])
```
- 3 Print the second item of the first list in *lol*

```
print(lol[0][1])
```
- 4 Print the third item to the last item of the second list in *lol*

Lists can contain other lists

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

practice

- 1 Create a new list *lol* which contains the lists *foo* and *bar*

```
lol=list()  
lol.append(foo)  
lol.append(bar)
```
- 2 Print the first item of *lol*

```
print(lol[0])
```
- 3 Print the second item of the first list in *lol*

```
print(lol[0][1])
```
- 4 Print the third item to the last item of the second list in *lol*

```
print(lol[1][2:])
```

Lists are mutable

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Definition

If you perform an operation on a list, it changes the list. In contrast, tuples and strings are immutable.

Example

Try the following:

```
newfoo = foo
```

```
newfoo[0] = 'new value'
```

Operations

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Practice

- ① Change the first item in the *foo* list to 12
- ② Now multiply the first item in the *foo* list by 2
- ③ Test whether “ham” is in the list *foo*

Operations

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Practice

- 1 Change the first item in the *foo* list to 12
`foo[0]=12`
- 2 Now multiply the first item in the *foo* list by 2
- 3 Test whether “ham” is in the list *foo*

Operations

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Practice

- ① Change the first item in the *foo* list to 12
`foo[0]=12`
- ② Now multiply the first item in the *foo* list by 2
`foo[0]*2`
- ③ Test whether “ham” is in the list *foo*

Operations

List basics

indexing

slicing

lol

mutable

operations

List methods

tuples

Practice

- 1 Change the first item in the *foo* list to 12
`foo[0]=12`
- 2 Now multiply the first item in the *foo* list by 2
`foo[0]*2`
- 3 Test whether “ham” is in the list *foo*
`'ham' in foo`

Len, min, and max

List basics

List methods

len

pop

sort

tuples

practice

- 1 How many items does *foo* contain?
- 2 What does `min(foo)` return?
- 3 What does `max(foo)` return?

Len, min, and max

List basics

List methods

len

pop

sort

tuples

practice

- ① How many items does *foo* contain?
`len(foo)`
- ② What does `min(foo)` return?
- ③ What does `max(foo)` return?

Len, min, and max

List basics

List methods

len

pop

sort

tuples

practice

- ① How many items does *foo* contain?
`len(foo)`
- ② What does `min(foo)` return?
- ③ What does `max(foo)` return?
Is that what you expected?

Queues and stacks

List basics

List methods

len

pop

sort

tuples

FIFO and LIFO

LIFO Last in, first out (stack)

FIFO First in, first out (queue)

Popping, appending, etc.

List basics

List methods

len

pop

sort

tuples

practice

- 1 Append the value 24 to the list *foo*
- 2 Insert the value “twenty” to the list *foo* as the 4th item
- 3 Find the index of “spam” in the list *foo*
- 4 remove the last item from *foo*, and store it as a new variable

Popping, appending, etc.

List basics

List methods

len

pop

sort

tuples

practice

- 1 Append the value 24 to the list *foo*
`foo.append(24)`
- 2 Insert the value “twenty” to the list *foo* as the 4th item
- 3 Find the index of “spam” in the list *foo*
- 4 remove the last item from *foo*, and store it as a new variable

Popping, appending, etc.

List basics

List methods

len

pop

sort

tuples

practice

- 1 Append the value 24 to the list *foo*
`foo.append(24)`
- 2 Insert the value “twenty” to the list *foo* as the 4th item
`foo.insert(3, 'twenty')`
- 3 Find the index of “spam” in the list *foo*
- 4 remove the last item from *foo*, and store it as a new variable

Popping, appending, etc.

List basics

List methods

len

pop

sort

tuples

practice

- 1 Append the value 24 to the list *foo*
`foo.append(24)`
- 2 Insert the value “twenty” to the list *foo* as the 4th item
`foo.insert(3, 'twenty')`
- 3 Find the index of “spam” in the list *foo*
`foo.index('spam')`
- 4 remove the last item from *foo*, and store it as a new variable

Popping, appending, etc.

List basics

List methods

len

pop

sort

tuples

practice

- 1 Append the value 24 to the list *foo*
`foo.append(24)`
- 2 Insert the value “twenty” to the list *foo* as the 4th item
`foo.insert(3, 'twenty')`
- 3 Find the index of “spam” in the list *foo*
`foo.index('spam')`
- 4 remove the last item from *foo*, and store it as a new variable
`lastfoo=foo.pop()`

More list methods

List basics

List methods

len

pop

sort

tuples

Practice

- 1 Append the following values to *foo*: 89, 23.4, 1
- 2 Create a new list *fooSorted* with the same contents as *foo*, but sorted

More list methods

Practice

- 1 Append the following values to *foo*: 89, 23.4, 1

```
foo.extend([89, 23.4, 1])  
foo.extend((89, 23.4, 1))
```
- 2 Create a new list *fooSorted* with the same contents as *foo*, but sorted

More list methods

Practice

- 1 Append the following values to *foo*: 89, 23.4, 1
`foo.extend([89, 23.4, 1])`
`foo.extend((89, 23.4, 1))`
- 2 Create a new list *fooSorted* with the same contents as *foo*, but sorted
`fooSorted=foo.sort()`

Tuples

List basics

List methods

tuples

Definition

Tuples are very similar to lists but are **immutable**

- Indexing and slicing work with tuples just as with lists.
- Tuples do not support methods such as sorting.
- You can create them with parentheses:
`mytuple=(10,50,'foo')`