

Python Accessing Items in a List

Textbook

Python Accessing Items in a List



As a reminder, this is how to create a list in Python:

```
1 shapes = ["square", "triangle", "circle", "rectangle", "star"]  
2  
3 print(shapes)
```

Try it!

This will print out ["square", "triangle", "circle", "rectangle", "star"]

Accessing Items in a List

Let's say that you have a list of shapes and you want to pull out the string "circle" and use that information for some new code.

```
1 shapes = ["square", "triangle", "circle", "rectangle", "star"]  
2  
3 print(shapes[2])
```

Try it!

This code will print out the string "circle".

You may be wondering why we used `shapes[2]` instead of `shapes[3]`. This is because of a concept called indexing.

Indexing in Python

Indexing refers to the position of an item in an array. **The numbers begin counting at 0 instead of at 1.** So in our previous example, the value in the list that has an index of 1 is `"triangle"`. The value of `"star"` sits at an index of 4.

```
shapes = ["square", "triangle", "circle", "rectangle", "star"]
```

What index value is `"square"` ?

`"square"` is at an index value of `0`.

Previously, you might have used indexing to figure out the position of characters in a string. For example:

What index value is `b` ?

```
"February"
```

`b` is at an index value of 2 in this string.

Indexing is both used for characters in a string, and values in a list. Both approaches begin counting at 0 instead of at 1.

Using Items Accessed from a List

Once an item is accessed from a list, it can be used for other things:

```
1 shapes = ["square", "triangle", "circle", "rectangle", "star"]
2
3 print("I just drew a purple " + shapes[2] + " on my paper.")
4
5
```

Try it!

The value of "circle" was used for string concatenation. Notice how the value "circle" was accessed right from the list and used in our code.

Index Inside an Index

Don't be alarmed if you ever see an index inside of an index. Just start with the innermost index value and move out, following the logic.

```
1 my_list = [2, 4, 3, 5, 8]
2 print(my_list[my_list[2]])
```

Try it!

This will print out 5.

First, the innermost index is `my_list[2]` which is `3`.

Now take that `3` and calculate `my_list[3]` which is `5`.

Check if an Item is in a List

If you have a really long list of items, it can be useful to use the following code to check and see if an item is in the list or not.

```
1 lucky_list = ["horseshoe", "clover", "rabbit foot", "socks"]
2
3 if "clover" in lucky_list:
4     print("clover is in lucky_list")
```

Try it!

This will print out the string `"clover is in lucky_list"`.

You can also check for the negative result.

```
1 lucky_list = ["horseshoe", "clover", "rabbit foot", "socks"]
2
3 if "hat" not in lucky_list:
4     print("hat is not in lucky_list")
```

Try it!

Locating Item in a List

You can also easily print out the list value of an item in a list using the `index()` method.

```
1 lucky_list = ["horseshoe", "clover", "rabbit foot", "socks"]
2
3 location = lucky_list.index("rabbit foot")
4
5 print(location)
```

Try it!

This will print out

`2`

Which is the index value for `"rabbit foot"`.

The Split Method

You can easily convert a string into a list in Python. This is done with the `split()` method. This will separate the string at the spaces.

```
1 safari = "We saw so many animals!"
2
3 words = safari.split()
4
5 print(words)
```

Try it!

This will print out the following code:

```
["We", "saw", "so", "many", "animals!"]
```

The split method will separate the string where the spaces are and put each word into a list as individual strings.

The List Function

You can also easily split up a single word into a list of letters by using the list() function. This will separate each individual letter.

```
1 animal = "giraffe"
2
3 mylist = list(animal)
4
5 print(mylist)
```

Try it!

This will print out the following code:

```
["g", "i", "r", "a", "f", "f", "e"]
```

Challenge Help

To create a list of strings, use the following code:

```
my_list = input().split()
```

Remember that inputs are automatically accepted as strings.

The input should look like words separated by spaces. No commas, parenthesis, or brackets are needed. For example: `apple banana strawberry` would be an acceptable input.

This would create a python list: `["apple", "banana", "strawberry"]`

These challenges may require putting a bunch of numbers as input into a list. To do this, use this code:

```
my_list = [int(n) for n in input().split()]
```

This code creates a list called "my_list" and the input().split() command breaks up the input into each individual integer. It uses a for loop to assign all the individual inputs to their index in the list.

For this code, put your input question right into the input() above. For example:

```
my_list = [int(n) for n in input("Input a list of numbers").split()]
```

The input should look like numbers separated by spaces. No commas, parenthesis, or brackets are needed. For example: `2 6 8 33 24 2` would be an acceptable input.

This would create a python list: `[2, 6, 8, 33, 24, 2]`

Checkpoint

Accessing Items in a List

1. Create a **list** that has **5 strings** in it.
2. From your list, pull out the **center** item and print it.
3. Create another print statement that **uses the center string in a concatenation**.

Requirements:

- Create a list that has 5 strings in it.
- From your list, pull out the center item and print it.
- Create another print statement that uses the center string in a concatenation.

Questions (14)

1. When indexing, at what number do we start counting?

MULTIPLE CHOICE

Choose the correct answer:

- A. 0
- B. 1
- C. 2
- D. -1

2. What kind of parentheses do we use when accessing an item in a list in Python?

MULTIPLE CHOICE

Choose the correct answer:

- A. ()
- B. { }
- C. []
- D. | |

3. What will separate a string where the spaces are and put each word into a list as individual strings?

MULTIPLE CHOICE

Choose the correct answer:

- A. split() method
- B. indexing
- C. listname[]
- D. if "_____" in listname

4. Which of the following ways is indexing specifically used for? Select all that apply.

SELECT MULTIPLE

Select all that apply:

- A. Finding the position of a character in a string.
- B. Finding the position of an item in a list.
- C. Committing a piece of code to memory.
- D. Documenting code

5. What will the following code print out?

MULTIPLE CHOICE

```
clothes = ["shirt", "pants", "hat", "shoes", "socks"] print(clothes[3])
```

Choose the correct answer:

- A. shoes
- B. socks
- C. 3
- D. ["shirt", "pants", "hat", "shoes", "socks"]
- E. shirt
- F. pants
- G. hat

6. What will the following code print out?

MULTIPLE CHOICE

```
clothes = ["shirt", "pants", "hat", "shoes", "socks"] print(clothes[0])
```

Choose the correct answer:

- A. 0
- B. ["shirt", "pants", "hat", "shoes", "socks"]
- C. shirt
- D. pants
- E. hat
- F. shoes
- G. socks

7. What is at index value of 4 in this list?

```
shapes = ["square", "triangle", "circle", "rectangle", "star"]
```

Choose the correct answer:

- A. ["square", "triangle", "circle", "rectangle", "star"]
- B. square
- C. triangle
- D. circle
- E. rectangle
- F. star

8. What will the following code print out?

```
clothes = ["shirt", "pants", "hat", "shoes", "socks"] print("Today I will wear " + clothes[3])
```

Choose the correct answer:

- A. Today I will wear shoes
- B. Today I will wear clothes
- C. Today I will wear
- D. Today I will wear hat

9. What will the following code print out?

```
clothes = ["shoes", "hat", "socks", "shirt"] if "pants" in clothes: print("Yes")
```

Choose the correct answer:

- A. Yes
- B. pants
- C. ["shoes", "hat", "socks", "shirt"]
- D. Nothing will print

10. What will the following code print out?

```
treat = "I love to eat jelly beans" myList = treat.split() print(myList)
```

Choose the correct answer:

- A. ['I', 'love', 'to', 'eat', 'jelly', 'beans']
- B. ["I love to", "eat jelly beans"]
- C. "I love to eat jelly beans"
- D. myList

11. What will the following code print out?

```
treat = "I love to eat jelly beans" myList = treat.split() print(myList[2])
```

Choose the correct answer:

- A. ['I', 'love', 'to', 'eat', 'jelly', 'beans']
- B. to
- C. love
- D. ["I love to", "eat jelly beans"]

12. Edit the text box below to debug (fix) the code:

Code to Debug:

```
1 clothes = ["shirt", "pants", "hat", "shoes", "socks"]
2
3 prin(clothes[3])
```

13. Edit the text box below to debug (fix) the code:

Code to Debug:

```
1 clothes = ["shoes", "hat", "socks", "shirt"]
2
3 if "pants" in lucky_list:
4     print("Yes")
```


14. Edit the text box below to debug (fix) the code:

DEBUG CODE

Code to Debug:

```
1 treat = "I love to eat jelly beans"
2
3 myList = treat-split()
4
5 print(myList)
```

Challenges (5)

1. Multiply Center Number

Create a program that pulls out the center number and multiplies it by 100.

Create an input that creates a **list of 9 numbers**.

Print out the center number **multiplied by 100**.

For example:

Input: 3 8 7 56 2 2 9 87 12

Output: 200

Another example:

Input: 1 2 3 4 5 6 7 8 9

Output: 500

Here is a reminder for how to create a list from an input:

```
my_list = [int(n) for n in input().split()]
```

2. Getting Around

In what ways do animals get around? Some might swim, run, slither, or crawl.

1. Create a program that pulls out the first and last item in a list of 5 strings.
2. Create an input that asks the user for **5 strings**.
3. Pull out the first and last item in the list.
4. Create a **print statement** to concatenate the first and last item in a sentence that matches the example output.

For example:

Input: swim jump walk fly crawl

Output: Some animals swim and some crawl

Another example:

Input: hop crouch slither prowl creep

Output: Some animals hop and some creep

Hint: This is how to create a list from an input `my_list = input().split()`

3. First Two, Last Two

1. Create a program that adds the first two numbers and the last two in a list.
2. Create an input that creates a list of **6 numbers**.
3. Print out the **sum of the first two and last two numbers** in the list.

Here is a reminder for how to create a list from an input:

```
my_list = [int(n) for n in input().split()]
```

For example:

Input: 2 7 2 90 10 9

Output: 28

Another example:

Input: 1 2 3 4 5 6

Output: 14

4. Missing Number

1. Write a program that generates a list from a bunch of consecutive numbers as the input.

Here's a reminder for how to do this:

```
my_list = [int(n) for n in input("Input a list of numbers").split()]
```

2. Make sure the numbers are consecutive and in order with single numbers missing.
3. Create a program that will print the missing numbers as well as a final number.

For example:

Input: 1 2 3 5 6 7 9 10

Output: 4 8 11

Another example:

Input: 22 23 24 26 27 28 30

Output: 25 29 31

Hint: Try using a for loop that goes through the list and checks for $x + 1$.

Hint: Try using the code "not in".

5. Boat Cruise

1. Consider the following string: **"On our boat cruise we saw crocodiles, flamingos, turtles, fish, and even a manatee!"**
2. Create a program that takes **a letter as an input**.
3. The program will then print out a list of items from the string separated at that letter.
4. If that letter doesn't appear in the string, print the following with the letter that's missing inserted at the blank: " **The letter _____ is not in the string** "

For example:

Input: **c**

Output: **['On our boat ', 'ruise we saw ', 'ro', 'odiles, flamingos, turtles, fish, and even a manatee!']**

Another example:

Input: **z**

Output: **The letter z is not in the string**

Hint: You can put a letter inside the **.split()**

Answer Keys & Solutions

Checkpoint Solutions

Accessing Items in a List

```
1 sports = ["basketball", "dance", "soccer", "ping pong", "rugby"]
2
3 print(sports[2])
4
5 print("I love to play " + sports[2])
```

Questions

1. When indexing, at what number do we start counting?

MULTIPLE CHOICE

Correct Answer:

- | | |
|-------|-------------|
| A. 0 | ✓ Correct |
| B. 1 | ✗ Incorrect |
| C. 2 | ✗ Incorrect |
| D. -1 | ✗ Incorrect |

Explanation:

It's not 1 like maybe you are used to.

2. What kind of parentheses do we use when accessing an item in a list in Python?

MULTIPLE CHOICE

Correct Answer:

- | | |
|-------|-------------|
| A. () | ✗ Incorrect |
| B. {} | ✗ Incorrect |
| C. [] | ✓ Correct |
| D. | ✗ Incorrect |

Explanation:

They are sometimes called "square brackets"

3. What will separate a string where the spaces are and put each word into a list as individual strings?

MULTIPLE CHOICE

Correct Answer:

A. `split()` method

✓ Correct

B. indexing

✗ Incorrect

C. `listname[]`

✗ Incorrect

D. if "_____" in listname

✗ Incorrect

Explanation:

This is a certain method in Python.

4. Which of the following ways is indexing specifically used for? Select all that apply.

SELECT MULTIPLE

Correct Answers:

A. Finding the position of a character in a string.

✓ Correct

B. Finding the position of an item in a list.

✓ Correct

C. Committing a piece of code to memory.

✗ Incorrect

D. Documenting code

✗ Incorrect

Explanation:

Indexing is usually used for locating something.

5. What will the following code print out?

MULTIPLE CHOICE

Correct Answer:

A. shoes

✓ Correct

B. socks

✗ Incorrect

C. 3

✗ Incorrect

D. ["shirt", "pants", "hat", "shoes", "socks"]

✗ Incorrect

E. shirt

✗ Incorrect

F. pants

✗ Incorrect

G. hat

✗ Incorrect

Explanation:

Indexing starts counting at 0.

6. What will the following code print out?

MULTIPLE CHOICE

Correct Answer:

A. 0

✗ Incorrect

B. ["shirt", "pants", "hat", "shoes", "socks"]

✗ Incorrect

C. shirt

✓ Correct

D. pants

✗ Incorrect

E. hat

✗ Incorrect

F. shoes

✗ Incorrect

G. socks

✗ Incorrect

Explanation:

Indexing starts counting at 0.

7. What is at index value of 4 in this list?

MULTIPLE CHOICE

Correct Answer:

A. ["square", "triangle", "circle", "rectangle", "star"]

✗ Incorrect

B. square

✗ Incorrect

C. triangle

✗ Incorrect

D. circle

✗ Incorrect

E. rectangle

✗ Incorrect

F. star

✓ Correct

Explanation:

Indexing starts counting at 0.

8. What will the following code print out?

MULTIPLE CHOICE

Correct Answer:

A. Today I will wear shoes

✓ Correct

B. Today I will wear clothes

✗ Incorrect

C. Today I will wear

✗ Incorrect

D. Today I will wear hat

✗ Incorrect

Explanation:

Indexing starts counting at 0.

9. What will the following code print out?

MULTIPLE CHOICE

Correct Answer:

A. Yes

✗ Incorrect

B. pants

✗ Incorrect

C. ["shoes", "hat", "socks", "shirt"]

✗ Incorrect

D. Nothing will print

✓ Correct

Explanation:

The string "pants" is not in the list and there is no else statement.

10. What will the following code print out?

Correct Answer:

- A. ['I', 'love', 'to', 'eat', 'jelly', 'beans'] ✓ Correct
- B. ["I love to", "eat jelly beans"] ✗ Incorrect
- C. "I love to eat jelly beans" ✗ Incorrect
- D. myList ✗ Incorrect

Explanation:

The `split()` method creates a new list of the string separated at the spaces.

11. What will the following code print out?

Correct Answer:

- A. ['I', 'love', 'to', 'eat', 'jelly', 'beans'] ✗ Incorrect
- B. to ✓ Correct
- C. love ✗ Incorrect
- D. ["I love to", "eat jelly beans"] ✗ Incorrect

Explanation:

The `split()` method creates a new list of the string separated at the spaces. Then, it pulls out the item at index value of 2.

12. Edit the text box below to debug (fix) the code:

Incorrect Code:

```
1 clothes = ["shirt", "pants", "hat", "shoes", "socks"]
2
3 prin(clothes[3])
```

Correct Solution:

```
1 clothes = ["shirt", "pants", "hat", "shoes", "socks"]
2
3 print(clothes[3])
```


Explanation:

There's a spelling error.

13. Edit the text box below to debug (fix) the code:

DEBUG CODE

Incorrect Code:

```
1 clothes = ["shoes", "hat", "socks", "shirt"]
2
3 if "pants" in lucky_list:
4     print("Yes")
```

Correct Solution:

```
1 clothes = ["shoes", "hat", "socks", "shirt"]
2
3 if "pants" in clothes:
4     print("Yes")
```

Explanation:

This code is looking at a list named lucky_list that doesn't exist.

14. Edit the text box below to debug (fix) the code:

DEBUG CODE

Incorrect Code:

```
1 treat = "I love to eat jelly beans"
2
3 myList = treat-split()
4
5 print(myList)
```

Correct Solution:

```
1 treat = "I love to eat jelly beans"
2
3 myList = treat.split()
4
5 print(myList)
```

Explanation:

The hyphen needs to be replaced with something else.

1. Multiply Center Number

Solution:

```
1 my_list = [int(n) for n in input("Create a list of 9 numbers").split()]
2
3 print(my_list[4] * 100)
```

2. Getting Around

Solution:

```
1 my_list = input().split()
2
3 print("Some animals " + my_list[0] + " and some " + my_list[4])
```

3. First Two, Last Two

Solution:

```
1 my_list = [int(n) for n in input("Create a list of 6 numbers").split()]
2
3 print(my_list[0] + my_list[1] + my_list[4] + my_list[5])
```

4. Missing Number

Solution:

```
1 my_list = [int(n) for n in input("Input a list of numbers").split()]
2
3 for x in my_list:
4     missing = x + 1
5     if missing not in my_list:
6         print(missing)
```

5. Boat Cruise

Solution:

```
1 choice = input("Choose a letter")
2
```

```
3 boat = "On our boat cruise we saw crocodiles, flamingos, turtles, fish, and even a  
manatee!"  
4  
5 if choice in boat:  
6     words = boat.split(choice)  
7     print(words)  
8 else:  
9     print("The letter " + choice + " is not in the string")
```