# course_1_assessment_9

## Due: 2018-11-25 01:23:00

Description: Assessment for Methods on Strings and Lists lesson. Score: 0 of 7 = 0.0%

## Questions

seqmut-1-2: Which method would you use to figure out the position of an item in a list?

**Not yet graded**

- A. .pop()
- B. .insert()
- C. .count()
- D. .index()

Check me | Compare me

✔ Yes, index will return the position of the first occurance of an item.

Multiple Choice (assess_question4_1_1_2)

seqmut-1-3: Which method is best to use when adding an item to the end of a list?

**Not yet graded**

- A. .insert()
- B. .pop()
- C. .append()
- D. .remove()

Check me | Compare me

✔ Yes, though you can use insert to do the same thing, you don't need to provide the position.

Multiple Choice (assess_question4_1_1_3)

**Not yet graded**

Write code to add 'horseback riding' to the third position (i.e., right before volleyball) in the list `sports`.

Save & Run | 5/13/2021, 4:13:05 PM - 11 of 11 | Show in CodeLens

```
1 sports = ['cricket', 'football', 'volleyball', 'baseball', 'softball', 'track and field',
2 sports.insert(2,'horseback riding')
3
4
```

ActiveCode (assess_ac4_1_1_4)

| Result | Actual Value | Expected Value | Notes | |
|---|---|---|---|---|
| Pass | ['cri...key'] | ['cri...key'] | Testing that sports is set correctly. | Expand Differences |
| Pass | '.insert(' | "sport...g')\n\n" | Testing that insert was used in your code. | Expand Differences |

You passed: 100.0% of the tests

**Not yet graded**

Write code to take 'London' out of the list `trav_dest`.

Save & Run | 5/13/2021, 4:13:30 PM - 2 of 2 | Show in CodeLens

```
1 trav_dest = ['Beirut', 'Milan', 'Pittsburgh', 'Buenos Aires', 'Nairobi', 'Kathmandu', 'Osa
2 trav_dest.pop(7)
3
4
```

ActiveCode (assess_ac4_1_1_5)

| Result | Actual Value | Expected Value | Notes | |
|---|---|---|---|---|
| Pass | ['Bei...rne'] | ['Bei...rne'] | Testing that trav_dest is set correctly. | Expand Differences |
| Pass | True | True | Testing that a method invocation was used in your code. | |

You passed: 100.0% of the tests

**Not yet graded**

Write code to add 'Guadalajara' to the end of the list `trav_dest` using a list method.

Save & Run 5/13/2021, 4:13:48 PM - 2 of 2 Show in CodeLens

```
1 trav_dest = ['Beirut', 'Milan', 'Pittsburgh', 'Buenos Aires', 'Nairobi', 'Kathmandu', 'Osa
2 trav_dest.append('Guadalajara')
3
4
```

ActiveCode (assess_ac4_1_1_6)

| Result | Actual Value | Expected Value | Notes | |
|---|---|---|---|---|
| Pass | ['Bei...ara'] | ['Bei...ara'] | Testing that trav_dest is set correctly. | Expand Differences |
| Pass | '+' | "trav_...a')\n\n" | Testing that you are not using concatenation (+). | Expand Differences |
| Pass | '.' | "trav_...a')\n\n" | Testing that a method invocation was used in your code. | Expand Differences |

You passed: 100.0% of the tests

**Not yet graded**

Write code to rearrange the strings in the list `winners` so that they are in alphabetical order from A to Z.

Save & Run 5/13/2021, 4:14:07 PM - 2 of 2 Show in CodeLens

```
1 winners = ['Kazuo Ishiguro', 'Rainer Weiss', 'Youyou Tu', 'Malala Yousafzai', 'Alice Munro
2 winners.sort()
3
4
```

ActiveCode (assess_ac4_1_1_7)

| Result | Actual Value | Expected Value | Notes | |
|---|---|---|---|---|
| Pass | ['Ali... Tu'] | ['Ali... Tu'] | Testing that winners is set correctly (Don't worry about actual and expected values). | Expand Differences |

You passed: 100.0% of the tests

**Not yet graded**

Write code to switch the order of the `winners` list so that it is now Z to A. Assign this list to the variable `z_winners`.

Save & Run 5/13/2021, 4:15:24 PM - 2 of 2 Show in CodeLens

```
1 winners = ['Alice Munro', 'Alvin E. Roth', 'Kazuo Ishiguro', 'Malala Yousafzai', 'Rainer W
2 winners.sort(reverse=True)
3 z_winners = winners
4
5
```

ActiveCode (assess_ac4_1_1_8)

| Result | Actual Value | Expected Value | Notes |
|---|---|---|---|
| Pass | ['You...nro'] | ['You...nro'] | Testing that z_winners is set correctly (Don't worry about actual and expected values). |

Expand Differences

You passed: 100.0% of the tests

Score Me