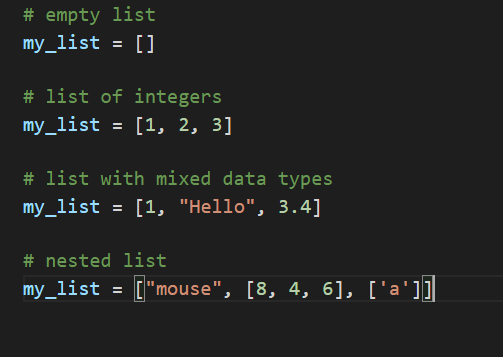
List in Python

The list is a most versatile datatype available in Python which can be written as a list of comma-separated values (items) between square brackets. Important thing about a list is that items in a list need not be of the same type.

## How to create a list?

In Python programming, a list is created by placing all the items (elements) inside square brackets [], separated by commas.

It can have any number of items and they may be of different types (integer, float, string etc.).



## Access List Elements

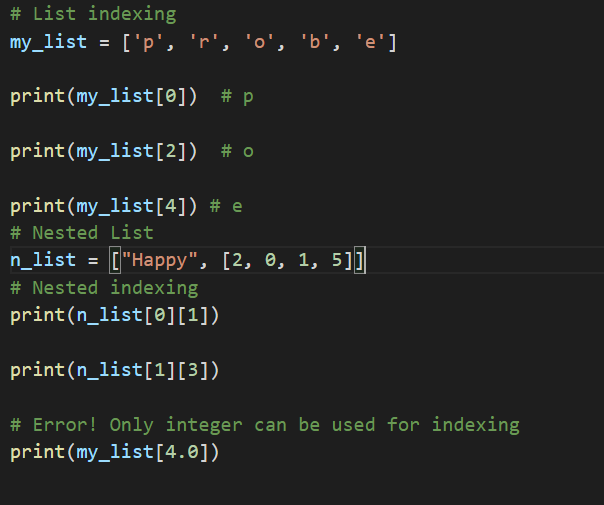
There are various ways in which we can access the elements of a list.

### **List Index**

We can use the index operator [] to access an item in a list. In Python, indices start at 0. So, a list having 5 elements will have an index from 0 to 4.

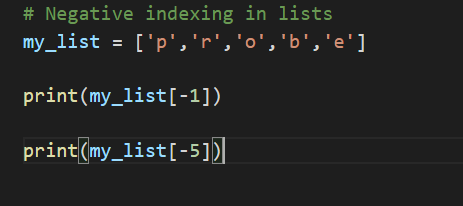
Trying to access indexes other than these will raise an IndexError. The index must be an integer. We can't use float or other types, this will result in TypeError.

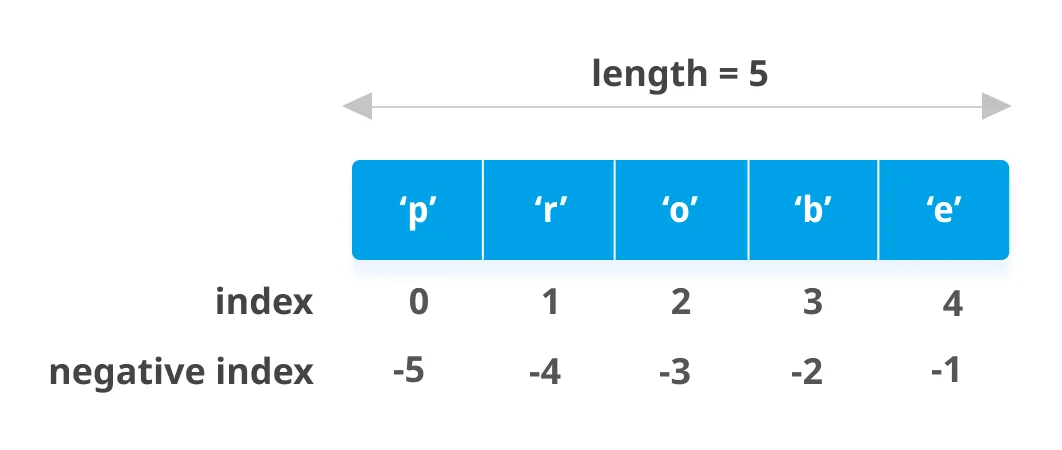
Nested lists are accessed using nested indexing.



### **Negative indexing**

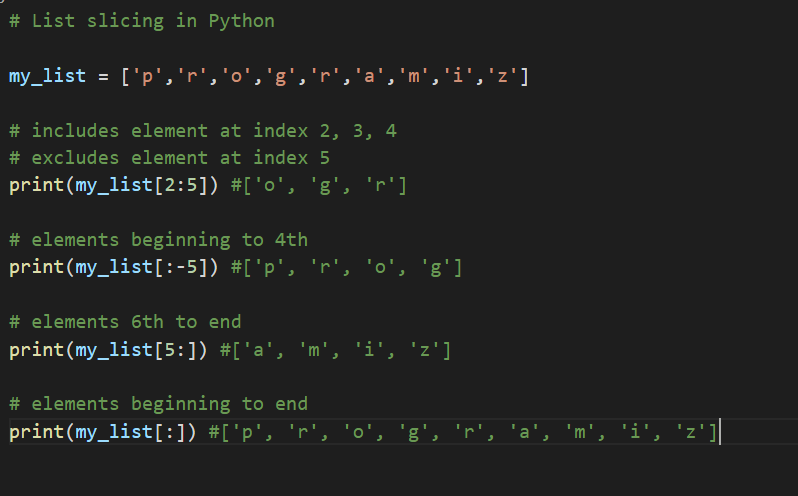
Python allows negative indexing for its sequences. The index of -1 refers to the last item, -2 to the second last item and so on.

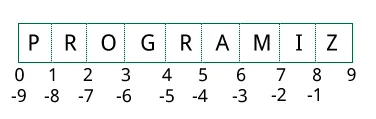




## How to slice lists in Python?

We can access a range of items in a list by using the slicing operator :(colon).



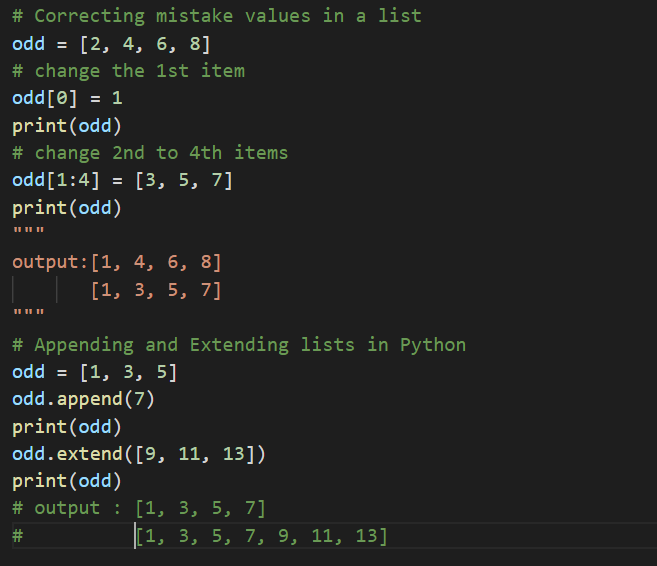


## Add/Change List Elements

Lists are mutable, meaning their elements can be changed unlike string or tuple.

We can use the assignment operator = to change an item or a range of items.

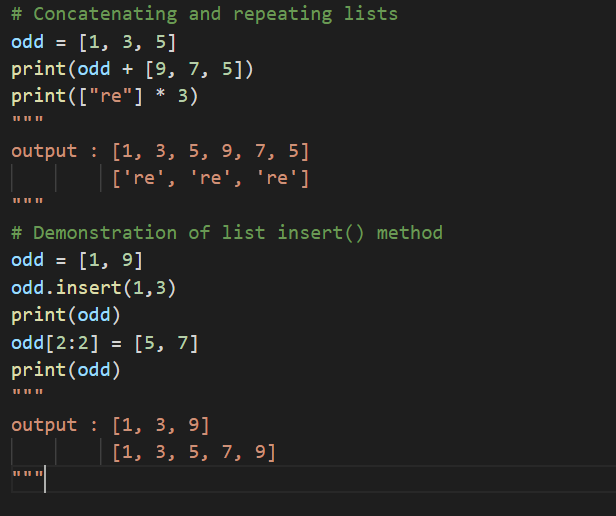
We can add one item to a list using the append() method or add several items using extend() method.



We can also use + operator to combine two lists. This is also called concatenation.

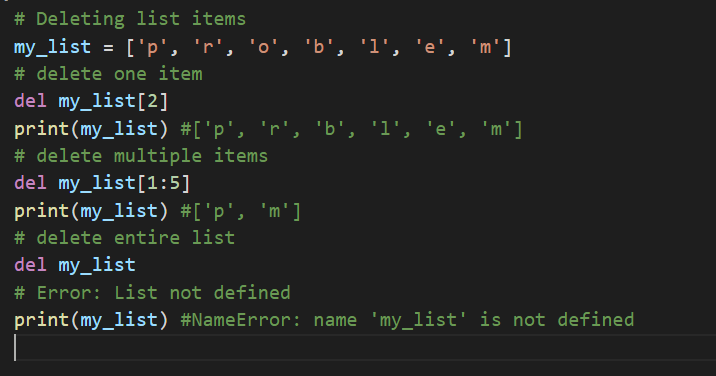
The \* operator repeats a list for the given number of times.

Furthermore, we can insert one item at a desired location by using the method insert() or insert multiple items by squeezing it into an empty slice of a list.



## Delete/Remove List Elements

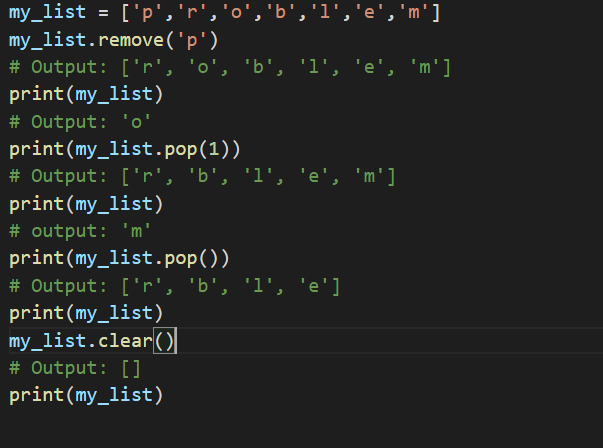
We can delete one or more items from a list using the keyword del. It can even delete the list entirely.



We can use remove() method to remove the given item or pop() method to remove an item at the given index.

The pop() method removes and returns the last item if the index is not provided. This helps us implement lists as stacks (first in, last out data structure).

We can also use the clear() method to empty a list.



## Python List Methods

Methods that are available with list objects in Python programming are tabulated below.

They are accessed as list.method(). Some of the methods have already been used above.

|  |  |
| --- | --- |
| Methods | Descriptions |
| append() | adds an element to the end of the list |
| extend() | adds all elements of a list to another list |
| insert() | inserts an item at the defined index |
| remove() | removes an item from the list |
| pop() | returns and removes an element at the given index |
| clear() | removes all items from the list |
| index() | returns the index of the first matched item |
| count() | returns the count of the number of items passed as an argument |
| sort() | sort items in a list in ascending order |
| reverse() | reverse the order of items in the list |
| copy() | returns a shallow copy of the list |

