Python Arrays

June 5, 2024

0.1 Python Arrays

Dr. Labeed Al-Saad

**Note: Python does not have built-in support for Arrays, but Python Lists can be used instead.

0.2 Arrays

**Note: This page shows you how to use LISTS as ARRAYS, however, to work with arrays in Python you will have to import a library, like the NumPy library.

**Arrays are used to store multiple values in one single variable:

Example:

Create an array containing car names:

[1]:
```
cars = ["Ford", "Volvo", "BMW"]
print(cars)
```

```
['Ford', 'Volvo', 'BMW']
```

0.3 What is an Array?

An array is a special variable, which can hold more than one value at a time.

If you have a list of items (a list of car names, for example), storing the cars in single variables could look like this:

car1 = “Ford”

car2 = “Volvo”

car3 = “BMW”

However, what if you want to loop through the cars and find a specific one? And what if you had not 3 cars, but 300?

The solution is an array!

An array can hold many values under a single name, and you can access the values by referring to an index number.

0.4 Access the Elements of an Array

You refer to an array element by referring to the index number.

Example:

Get the value of the first array item:

[2]:
```
cars = ["Ford", "Volvo", "BMW"]

x = cars[0]

print(x)
```

```
Ford
```

Example:

Modify the value of the first array item:

[3]:
```
cars = ["Ford", "Volvo", "BMW"]

cars[0] = "Toyota"

print(cars)
```

```
['Toyota', 'Volvo', 'BMW']
```

0.5 The Length of an Array

Use the len() method to return the length of an array (the number of elements in an array).

Example Return the number of elements in the cars array:

[4]:
```
cars = ["Ford", "Volvo", "BMW"]

x = len(cars)

print(x)
```

```
3
```

0.6 Looping Array Elements

You can use the for in loop to loop through all the elements of an array.

Example:

Print each item in the cars array:

[5]:
```
for x in cars:
  print(x)
```

```
Ford
Volvo
BMW
```

0.7 Adding Array Elements

You can use the append() method to add an element to an array.

Example:

Add one more element to the cars array:

[6]:
```
cars = ["Ford", "Volvo", "BMW"]

cars.append("Honda")

print(cars)
```

```
['Ford', 'Volvo', 'BMW', 'Honda']
```

0.8 Removing Array Elements

You can use the pop() method to remove an element from the array.

Example:

Delete the second element of the cars array:

[7]:
```
cars = ["Ford", "Volvo", "BMW"]

cars.pop(1)

print(cars)
```

```
['Ford', 'BMW']
```

You can also use the remove() method to remove an element from the array.

**Note: The list's remove() method only removes the first occurrence of the specified value.

Example:

Delete the element that has the value "Volvo":

[9]:
```
cars = ["Ford", "Volvo", "BMW", "Volvo"]

cars.remove("Volvo")

print(cars)
```

```
['Ford', 'BMW', 'Volvo']
```

0.9 Array Methods

Python has a set of built-in methods that you can use on lists/arrays.

Method Description

append() Adds an element at the end of the list

clear() Removes all the elements from the list

copy() Returns a copy of the list

count() Returns the number of elements with the specified value

extend() Add the elements of a list (or any iterable), to the end of the current list

index() Returns the index of the first element with the specified value

insert() Adds an element at the specified position

pop() Removes the element at the specified position

remove() Removes the first item with the specified value

reverse() Reverses the order of the list

sort() Sorts the list

**Note: Python does not have built-in support for Arrays, but Python Lists can be used instead.