



Lists vs Arrays in Python

We'll cover the following



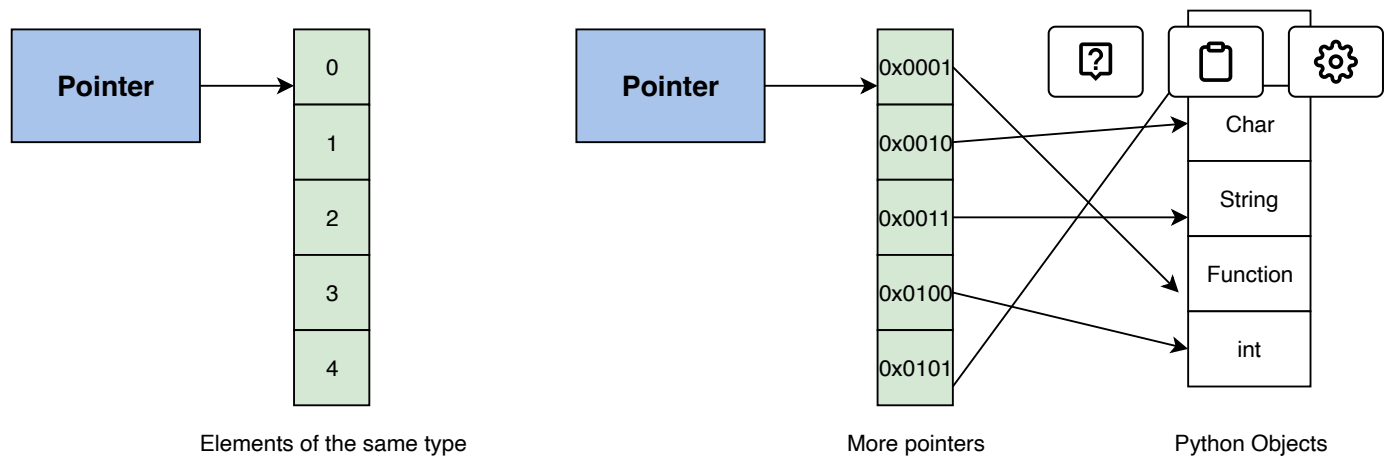
- Lists vs Arrays
 - Differences
 - Similarities
 - When to use arrays and when to use lists
 - The NumPy Library

Lists vs Arrays

Differences

Python lists are very flexible and can hold completely heterogeneous arbitrary data, but they use a lot more space than Python arrays. Each list contains pointers to a block of pointers, each of which in turn points to a full Python object. Again, the advantage of the list is flexibility: because each list element is a full structure containing both data and type information, the list can be filled with data of any desired type. Arrays lack this flexibility but are much more efficient for storing and manipulating data.

The differences between the two largely exist because of the aforementioned backend implementation. Arrays in Python are implemented just like C arrays, with a pointer pointing to the first element of the array with the rest existing contiguously in the memory. ↻

**Python Arrays****Python Lists**

Similarities

Arrays and lists are very similar in syntactical usage and functionality. Both are used to store data. Both are mutable, i.e., the data in both is not constant and can be modified. Both can also be indexed and iterated through. Both can be sliced and are in fact sliced the same way.

When to use arrays and when to use lists

Because of the way that lists are implemented, elements can be appended to them very efficiently. If you want your collection to grow and shrink in a time-efficient manner, then lists should be used. It's also better to use lists if you need to manage lots of different data types, however, arrays are better when you need to store a lot of data and perform a large number of computationally intensive mathematical operations. Note that the arrays from the NumPy library of Python are better suited for mathematical use cases.

The NumPy Library

Here is how a NumPy array is initialized. You can read more about them on the [official documentation](#). They're used in algorithms where a lot of

mathematical operations are performed on large sets of data and machine learning algorithms.



```
import numpy as np
numpy_array = np.array([1, 2, 3])
print(numpy_array)
```



Interviewing soon? We've partnered with Hired so that companies apply to you instead of you applying to them. [See how](#) ⓘ



← Back

Next →

Arrays

Challenge 1: Remove Even Integers fr...

✓ Completed



Report an Issue

