Understanding the Implementation of Basic Data Structures in Python

March 26, 2025

If you're curious to see **how basic data structures like lists, dictionaries, sets, and tuples are implemented in Python**, there are a few ways to explore this depending on how deep you want to go:

1. Quick Info via `help()` or `__doc__`

You won't see the full implementation, but you'll get a summary of how it works.

```
help(list)
help(dict)
```

2. Browse the CPython Source Code (Actual Implementation)

Python's most widely used implementation is **CPython**, and most of the core data structures are written in **C** for performance.

You can browse the source code here: `https://github.com/python/cpython`

Key files:

`list`	listobject.c
`dict`	dictobject.c
`set`	setobject.c
`tuple`	tupleobject.c
`str`	unicodeobject.c

3. Use the `collections` Module for Higher-Level Structures

Some data structures like `deque`, `defaultdict`, and `OrderedDict` are implemented in Python or C extensions.

```
from collections import deque
help(deque)
```

You can find their implementation in:

- collections/__init__.py
- _collectionsmodule.c (C backend)

4. Use Pure Python Versions (For Educational Purposes)

There are many educational re-implementations of Python data structures in pure Python online. For instance:

- cpython-data-structures on GitHub – great for understanding the logic behind them.

Would you like to see a simplified Python version of, say, a `list` or `dict`?

5. Official Reference

`https://wiki.python.org/moin/TimeComplexity`