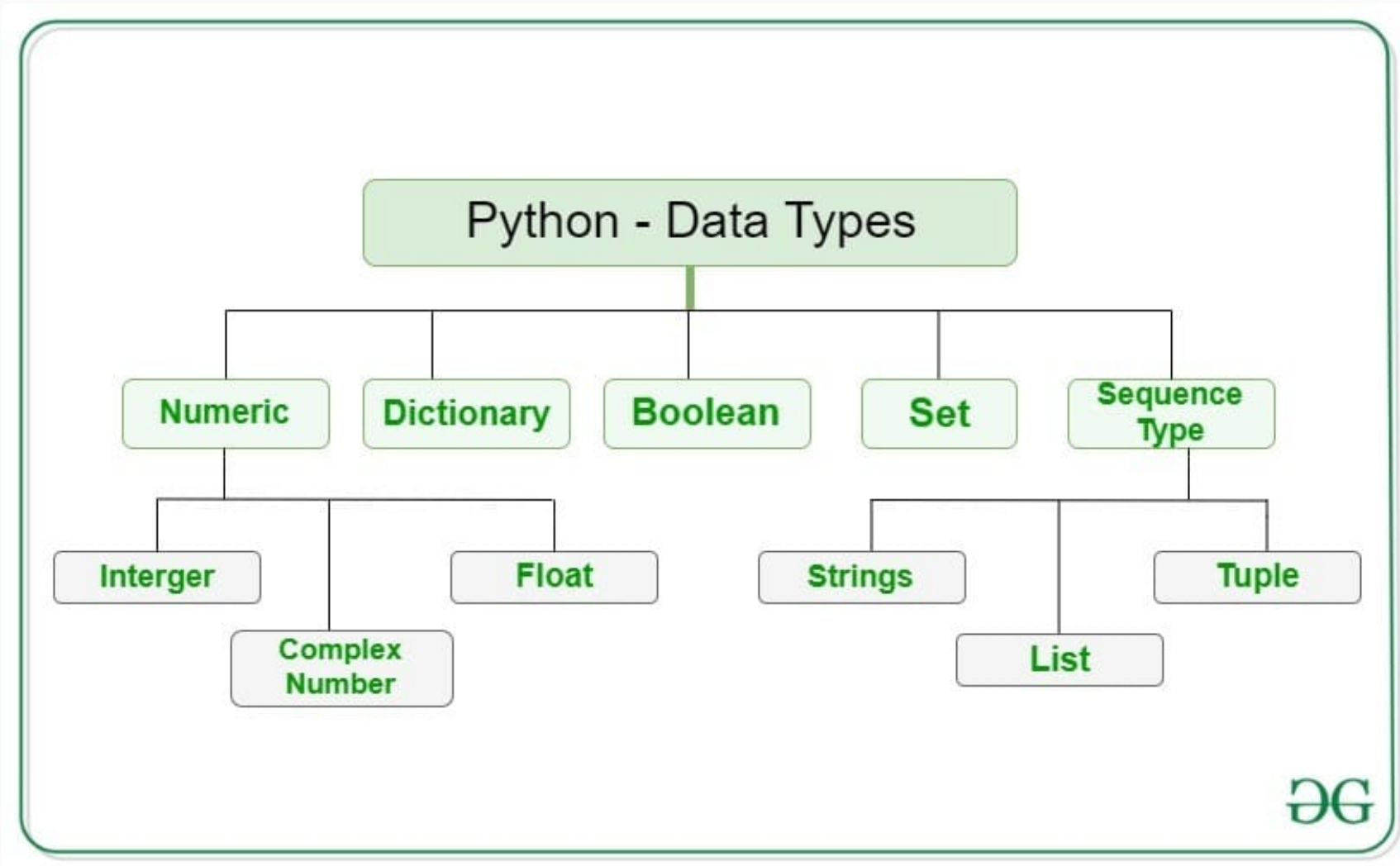


Understand Datatypes in Python

Python is a dynamic, high-level programming language that supports a variety of datatypes. Each data type has its own characteristics and uses. Understanding the datatypes in Python is important for effective programming.



Numeric Data Types

Python supports various numeric data-types, such as int, float and complex. In Python, numbers are treated as objects and the operations can be performed on them similar to other objects. For instance, arithmetic, relational and logical operations.

The distinguishing feature is that python never raises any exceptions in case of overflow and underflow of numeric types.

String Data Type

Strings in Python are identified with single and double quotes. Python's string module provides many functions to manipulate strings. In addition, Python's format strings let us embed expressions in strings, including the values of variables and expressions. In brief, Python's strings are rich, versatile and expressive.

Single Quote

'Hello World!'

Double Quote

"Python is awesome!"

List Data Type

List is a collection of items that can be stored sequentially in Python. In other words, List is an ordered set of items enclosed in square brackets separated by commas. We can store different types of items in a list to create a complex collection data structures.

Syntax	index	types
[10, 20, 30]	0, 1, 2	Integer
['A','B','C']	0, 1, 2	String
['apple', 'banana']	0, 1	Nested

Dictionary Data Type

A dictionary comprises key-value pairs. Python's dictionary values have no restrictions in terms of type. They can span multiple data types or have single data type values, making them flexible. We can perform operations on them just like strings and lists.

1 Key

apple

3 Key

carrot

2 Value

"A fruit that is red or green in colour"

4 Value

"A vegetable that is usually orange in colour"

Tuple Data Type

Tuple is an ordered sequence of items, enclosed in parenthesis. Unlike lists, tuples are immutable, meaning they cannot be changed once created.

1 Ordered

Items in tuples are ordered, so the items have a distinct position in the sequence.

2 Immutable

Tuples are immutable, meaning they cannot be updated or deleted once created.

Set Data Type

Set is an unordered collection of unique elements, found in curly braces '{'. You can perform mathematical set operations on them, such as union, intersection and difference.

Syntax	Output
{1, 2, 3}	{1,2,3}
{1,2,'Sam'}	{'Sam',1,2}

Conclusion

Python offers a variety of datatypes that support a broad range of applications. Understanding datatypes is critical to effective programming. Identification of the right datatype can make the code more concise and easier to read and maintain.

```
message", m => {  
  a = m.split(" ")  
  h(a[0]){  
    case "connect":  
      f(a[1]){  
        if(clients.has(a[1])){  
          ws.send("connected");  
          ws.id = a[1];  
        }else{  
          ws.id = a[1]  
          clients.set(a[1], {client: {position: {x: 0, y: 0},  
            ws.send("connected")  
        }  
      }else{  
        let id = Math.random().toString().slice(2, 8)  
        ws.id = id;  
        clients.set(id, {client: {position: {x: 0, y: 0},  
          ws.send("connected")  
        }  
      }  
    }  
  }  
}
```

Concise

Python's rich data types make code less bulky and more readable.



Effective

Accurate use of data types makes Python code more effective, enabling faster execution and minimised memory management.