Python Data Types Fundamentals for Beginners

Python has a variety of built-in data types that allow developers to work efficiently with different types of data. Understanding data types is a key part of learning Python programming.

1. Basic Data Types

Python has several built-in data types that can be categorized as follows: - Numeric Types: int, float, complex - Text Type: str - Sequence Types: list, tuple, range - Mapping Type: dict - Set Types: set, frozenset - Boolean Type: bool - Binary Types: bytes, bytearray, memoryview

2. Numeric Types

Numeric types store numbers and allow arithmetic operations. **Examples:** x = 10 # int y = 3.14 # float z = 3 + 2j # complex print(type(x), type(y), type(z))

3. String Type (str)

Strings are used to store text data. Strings are immutable, meaning they cannot be changed after creation. **Examples:** name = "Python" print(name[0]) # Access character print(name.lower()) # Convert to lowercase print(name.upper()) # Convert to uppercase print(len(name)) # Get length

4. List Type

Lists are ordered, mutable collections that can store elements of different data types. **Examples:** fruits = ["apple", "banana", "cherry"] fruits.append("orange") print(fruits) print(fruits[1]) # Access by index **Properties:** - Ordered - Mutable (can be changed) - Allows duplicate elements

5. Tuple Type

Tuples are ordered but immutable collections of elements. **Examples:** numbers = (1, 2, 3, 4) print(numbers[0]) **Properties:** - Ordered - Immutable (cannot be modified) - Allows duplicates

6. Dictionary Type

Dictionaries store data in key-value pairs. Each key must be unique. **Examples:** student = {"name": "Alice", "age": 25, "course": "Python"} print(student["name"]) student["age"] = 26 print(student) **Properties:** - Unordered (in Python 3.6+, insertion order is preserved) - Mutable - Keys must be unique and immutable

7. Set Type

Sets are unordered collections of unique elements. **Examples:** numbers = {1, 2, 3, 3, 4} print(numbers) # Duplicate 3 will be removed numbers.add(5) print(numbers) **Properties:** - Unordered - Mutable (but elements must be immutable) - Does not allow duplicates

8. Boolean Type

Boolean values represent one of two possible states: True or False. **Examples:** x = True y = False print(x and y) # False print(x or y) # True

9. Type Conversion

Python allows conversion between compatible data types using type casting functions. **Examples:** x = 5 y = float(x) # int to float z = str(x) # int to string print(type(y), type(z))

10. Summary Table

Data Type	Mutable	Ordered	Duplicates Allowed
int	No	N/A	N/A
float	No	N/A	N/A
str	No	Yes	Yes
list	Yes	Yes	Yes
tuple	No	Yes	Yes
dict	Yes	Yes	Keys No
set	Yes	No	No
bool	No	N/A	N/A

11. Summary

- Python provides several built-in data types for handling different kinds of data.
- Data types like lists and dictionaries are mutable, while tuples and strings are immutable.
- Sets are useful for storing unique items.
- Knowing when to use each data type is crucial for writing efficient Python programs.