Datatypes in Python

Datatype is a classification that specifies the type of value a variable can hold.

Basic data types

1. Numeric - Integer, Float, Complex
2. Sequence - string, list, Tuple
3. Dictionary
4. Set
5. Boolean
6. Binary : Bytes, Bytearray, Memoryview

It can be grouped under 4 Data Types.

i) Numeric Data type.

▶) "Integer (int)" type eg. age = 20, temp = -12

```
x = 10
y = 3
//Addition print("sum:", sum_result)        o/p = 13
//subtraction. difference = x - y
             print("difference :" difference)   o/p = 7
```

// Multiplication

```
product = x * y
print ("Product :", product)
```

// Division

```
Quotient = x/y
print ("Quotient :" quotient)
```

▶ Floating-Point (float) Type. - can have decimal points and represent actual numbers.

pi = 3.14159.

ii) Text Data Type.

Strings are used to store sequences of characters including letters, numbers, & symbols.

Creating strings

```
# single quotes   message = 'Hello, Nithya'
# double quotes   name = "Nithya"
# triple quotes   quote = ''' Hello, am Nithya"""
```

connecting string together

age = 28.

```
# using format method.
    info = "Name : {}, Age : {}". Format (name, age)
# using F-Strings or Formatted string Literal
    f"Name : {name}, Age : {age}"
```

String Methods

```
# getting the length of a string
length = len(name)
# converting to uppercase
uppercase_name = name.upper()
# lowercase = name.lower()
# capitalizing the first letter
capitalized_name = name.capitalize()
"String Manipulation" - removing leading and trailing whitespace
trimmed_message = message.strip()
"splitting and joining strings"
# splitting a string into a list of words
word_list = message.split(",")
# joining a list of words into single string
joined_words = "_".join(word_list)

"String formatting"
Print("The {animal} says {sound}.".format
(animal="dog", sound="woof"))
```

iii) Set Data type Set is a built in datatype in python that represent an unordered group of distinct items. Curly braces {} are used

```
vegetables = {"carrot", "spanish", "broccoli"}
fruits = {"apple", "banana", "orange", "banana"}
print(fruits)
# Adding elements to set
fruits.add("pear")
# removing elements to the set
fruits.remove("apple")

Basic set operations
# Union of sets
food = fruit.union(vegetables)
# intersection of sets
common_items = fruits.intersection(vegetables)
# difference of sets
unique_fruits = fruits.difference(vegetables)
```

iv) Boolean Data Type which may represent "True or false".