PyPy3

– Indentation

The number of spaces is up to you as a programmer, but it has to be at least one.

You have to use the same number of spaces in the same block of code

Since Python will ignore string literals that are not assigned to a variable, you can add a multiline string (triple quotes) in your code, and place your comment inside it:

"""  
This is a string  
written in  
more than just one line  
"""

If you create a variable with the same name inside a function, this variable will be local, and can only be used inside the function. The global variable with the same name will remain as it was, global and with the original value.

A screenshot of a cell phone

Description automatically generated

Use global keyword if you want to change the value of global variable

Python has the following data types built-in by default, in these categories:

|  |  |  |
| --- | --- | --- |
| Text Type: | | str |
| Numeric Types: | | int, float, complex |
| Sequence Types: | | list, tuple, range |
| Mapping Type: | | dict |
| Set Types: | | set, frozenset |
| Boolean Type: | | bool |
| Binary Types: | | bytes, bytearray, memoryview |
|  | |  |
| x = b"Hello" | bytes | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_type_bytes) |
| x = bytearray(5) | bytearray |  |

x = list(("apple", "banana", "cherry")) // convert to list

random.randrange(1,10)

random.suffle(list\_var)

isinstance("this is a string", str) //returns true check type of variable

str.\_\_class\_\_ = str // true

**STRINGS—**

txt = "My {} is John, {} I {} am"

print(txt.format(age+3,age+1)) // tuple out of Index

The strip() method removes any whitespace from the beginning or the end:

a = " Hello, World! "  
print(a.strip()) # returns "Hello, World!"

a.lower()

a.upper()

a.replace("H", "J")

a.split(",")

txt = "The rain in Spain stays mainly in the plain"  
x = "ain" in txt

|  |  |
| --- | --- |
| [count()](https://www.w3schools.com/python/ref_string_count.asp) | Returns the number of times a specified value occurs in a string |
| [find()](https://www.w3schools.com/python/ref_string_find.asp) | Searches the string for a specified value and returns the position of where it was found |
| [index()](https://www.w3schools.com/python/ref_string_index.asp) | Searches the string for a specified value and returns the position of where it was found |

Boolean –

empty values, such as (), [], {}, "", the number 0, and the value None. And of course the value False evaluates to False.

50/3 = 16.666666666666668

50//3 == 16

Is : if they are actually the same object, with the same memory location

== : if content is equal

**LISTS --**

**6**

**thislist = ["apple", "banana", "cherry"]**

**thislist.append("orange")**

**thislist.insert(1, "orange")**

thislist.remove("banana") //remove only first item

|  |  |
| --- | --- |
| [reverse()](https://www.w3schools.com/python/ref_list_reverse.asp) | Reverses the order of the list |
| [sort()](https://www.w3schools.com/python/ref_list_sort.asp) | Sorts the list |

The pop() method removes the specified index, (or the last item if index is not specified):  
thislist.pop()

del thislist[0]

thislist.clear()

mylist = thislist.copy() // L2 = L1 will refferance to same list

mylist = list(thislist) // copy list

**SET --**6666666668

SET is unordered // no indexing

thisset = {"apple", "banana", "cherry"}  
thisset.add("orange")

thisset.update(["orange", "mango", "grapes"]) // add all items

thisset.remove("banana") // does not exist, remove() will raise an error

thisset.pop() // remove random element

set3 = set1.union(set2)

thisset = set(("apple", "banana", "cherry"))

**DICT –**

**thisdict = {  
  "brand": "Ford",  
  "model": "Mustang"  
}**

**for x in thisdict.values():  
  print(x)**

for x, y in thisdict.items():  
  print(x, y)

if "model" in thisdict:

thisdict.pop("model")

|  |  |
| --- | --- |
| [update()](https://www.w3schools.com/python/ref_dictionary_update.asp) | Updates the dictionary with the specified key-value pairs |
| [values()](https://www.w3schools.com/python/ref_dictionary_values.asp) | Returns a list of all the values in the dictionary |

while i < 6:  
  print(i)  
  i += 1  
else:  
  print("i is no longer less than 6")

Increment the sequence with 3 (default is 1):

for x in range(2, 30, 3):  
  print(x)

FUNC \_\_

It does not have to be named self , you can call it whatever you like, but it has to be the first parameter of any function in the class:

JSON is a data format (a string), Python dictionary is a data structure (in-memory object).

[-1:0:-1] means: start from the index len(string)-1 and move up to 0(not included) and take a step of -1(reverse) in short [::-1] ==🡺 slice[start:stop:step]

**Double babble method for converting binary to decimal**

5 = 101

Itr:101

Ans = ans\*2 +itr

# python code to demonstrate way to increment character

# ord returns Unicode character

  x = chr(ord(ch) + 3)