Python

27 February 2024

15:20

W3School

* Print("Hi")
* No need for semicolon
* Indentation
* Python helloworld.py
* # for comments, """ multiline hack for comment"
* X =int(3), str(3), float(3) // cast
* type(x) to get type
* Both '' and "" works
* Unpack a list
  + list1 = ["A","b", "c"]
  + X , y,z = list1
* Global variable,
  + Define outside
  + With global keyword , defined anywhere
* Data type : str, int, float, complex, list[], tuple(), range, dict{:}, set{}, frozenset{}, bool, byte, bytearray, memoryview, NoneType
* list(("a","b")), tuple(("a","b")), dict(name = "h", age = 5)
* In and not in to check in string , list etc.
* slicing : with all collection like string, list etc, strIp[startIndex:endIndex(not included)], -ve index for backward
* String func: upper(), lower(), strip(), replace(a,b), split(,),
* Operator : same as c++, and , or, not, is, is not, in, not in,
* Collection like list etc. does not need to contain data of same data type ;list1 = ["abc", 34, True, 40, "male"]
* List : append(), extend(), pop(index), del lost[1], remove("a"), clear()
* For x in list1:
* new list = [x for x in fruits if "a" in x] //list comprehension // newlist = [*expression* for *item* in *iterable* if *condition* == True]
* .sort() to sort any collection
* Tupples: same as list but immutable
* To create single element collection add , at end
* Set are list without duplicate
* Dictionary: list of key value pair: get(), keys(), values(), items(), pop(), popitem(), del dict1[model], clear(),
* If cond : elif, else, print("A") if a > b else print("B")
* While loop can have else , execute the exit cond
* Continue and break, pass
* Range()
* For item in list1: , for loop also can have else
* def myFunc():, positional args , keyword args
* \*args : represent a list
* x = lambda a : a + 10; y = x(5)
* Class MyClass: p1 = MyClass()
* \_\_init\_\_(): constr, \_\_str\_\_():
* class Student(Person): //inheritance
* Super() to access parent
* self. // to access
* \_name : protected , \_\_name:private
* To make things iterable define : \_\_iter\_\_(), \_\_next\_\_()
* All class function have self as first arg
* Import module , import module as mx, module.var/func, from module import entity
* Few lib: datetime, math, json,re, pip,
* Try, except(catch), finally, else . Raise(throw)
* x=input("message"), raw\_input()
* f"hello {name}"
* Create \_\_init\_\_.py to make that directory as package
* F = open(filename, mode), f.read(), readline(), for line in f: print(line),
* Os module and funct for create dir , file ,
* Lib : matplotlib
* Use numpy to calculate percentile, mean, median, mode, variance etc.