Python Programming

Jian Zhang

Nov. 23, 2023@PHBS

Comments

Comments can be used to explain Python code.

Comments can be used to make the code more readable.

Comments can be used to prevent execution when testing code.

Comments

```
#This is a comment
print("Hello, World!")

print("Hello, World!") #This is a comment
```

Comments

```
#This is a comment
#written in
#more than just one line
print("Hello, World!")

"""
This is a comment
written in
more than just one line
"""
print("Hello, World!")
```

Complex Types

List
Tuple
Dictionary
Set

List

```
# Create a List:
thislist = ["apple", "banana", "cherry"]
print(thislist)

# List items are ordered, changeable, and allow duplicate values
thislist = ["apple", "banana", "cherry", "apple", "cherry"]
print(thislist)

# List Length
thislist = ["apple", "banana", "cherry"]
print(len(thislist))
```

List

```
# List Items - Data Types
list1 = ["apple", "banana", "cherry"]
list2 = [1, 5, 7, 9, 3]
list3 = [True, False, False]
list4 = ["abc", 34, True, 40, "male"]

# List Type
print(type(list1))
print(type(list2))
print(type(list3))
print(type(list4))
```

List

```
## list() Constructor
thislist = list(("apple", 12, "ww"))
print(thislist)

# Access Items
thislist = ["apple", "banana", "cherry"]
print(thislist[1])

# Negative Indexing
thislist = ["apple", "banana", "cherry"]
print(thislist[-1])
```

List

```
# Range of Indexes
thislist = ["apple", "banana", "cherry", "orange", "kiwi", "melon", "mango"]
print(thislist[2:3])
print(thislist[2])

thislist = ["apple", "banana", "cherry", "orange", "kiwi", "melon", "mango"]
print(thislist[:4])

thislist = ["apple", "banana", "cherry", "orange", "kiwi", "melon", "mango"]
print(thislist[2:-1])
```

List

```
# Range of Negative Indexes
thislist = ["apple", "banana", "cherry", "orange", "kiwi", "melon", "mango"]
print(thislist[-4:-1])

# Check if Item Exists
thislist = ["apple", "banana", "cherry"]
if "appl" in thislist:
    print("Yes, 'apple' is in the fruits list")
else:
    print('Wrong')
```

List

```
# Change Item Value
thislist = ["apple", "banana", "cherry"]
thislist[1] = [12,'apple']
print(thislist)

# Change a Range of Item Values
thislist = ["apple", "banana", "cherry", "orange", "kiwi", "mango"]
thislist[1:3] = ["blackcurrant", "watermelon"]
print(thislist)
```

List

```
thislist = ["apple", "banana", "cherry"]
thislist[1:2] = ["blackcurrant", "watermelon"]
print(thislist)

thislist = ["apple", "banana", "cherry"]
thislist[1] = ["blackcurrant", "watermelon"]
print(thislist)
```

List

```
thislist = ["apple", "banana", "cherry"]
print(thislist[1:3])
thislist[1:3] = ["watermelon"]
print(thislist)

# Append Items
thislist = ["apple", "banana", "cherry"]
thislist.append(12)
print(thislist)
```

List

```
# Insert Items
thislist = ["apple", "banana", "cherry"]
thislist.insert(2, 12)
print(thislist)

# Extend List
thislist = ["apple", "banana", "cherry"]
tropical = ["mango", "pineapple", "papaya"]
# thislist.extend(tropical)
# print(thislist)
tropical.extend(thislist)
print(tropical)
print(thislist)
```

List

```
# Add Any Iterable
thislist = ["apple", "banana", "cherry"]
thistuple = ("kiwi", "orange")
thislist.extend(thistuple)
print(thislist)

# Remove Specified Item
thislist = ["apple", "banana", "cherry"]
thislist.remove("banana")
print(thislist)
```

List

```
# Remove Specified Index
thislist = ["apple", "banana", "cherry"]
thislist.pop(1)
print(thislist)

thislist = ["apple", "banana", "cherry"]
thislist.pop()
print(thislist)

thislist = ["apple", "banana", "cherry"]
del thislist[0]
print(thislist)
```

List

```
thislist = ["apple", "banana", "cherry"]
print(thislist)
del thislist
print(thislist)

# Clear the List
thislist = ["apple", "banana", "cherry"]
print(thislist)
thislist.clear()
print(thislist)
```

List

```
# Loop Through a List
thislist = [1,2,3,4,5]
sum = 0
for x in thislist:
    sum = sum + x
    print(x)
print(sum)

mylist = []
mylist.append('apple')
print(mylist)
```

List

```
# Loop Through the Index Numbers
thislist = ["apple", "banana", "cherry", 4]
for i in range(len(thislist)):
    print(thislist[i])
for x in thislist:
    print(x)

# Using a While Loop
thislist = ["apple", "banana", "cherry"]
i = 0
while i < len(thislist):
    print(thislist[i])
    i = i + 1
```

List

```
# List Comprehension
fruits = ["apple", "banana", "cherry", "kiwi", "mango"]
newlist = []

for x in fruits:
  if "a" in x:
    newlist.append(x)

print(newlist)
```

List

```
fruits = ["apple", "banana", "cherry", "kiwi", "mango"]

newlist = [x for x in fruits if "k" in x]
# newlist = [x for x in fruits]
print(newlist)

fruits = ["apple", "banana", "cherry", "kiwi", "mango"]
newlist = [x.upper() for x in fruits]
print(newlist)

newlist = ['hello' for x in fruits]
print(newlist)

newlist = [x for x in range(10) if x < 5]
print(newlist)
```

List

```
# Sort List Alphanumerically
thislist = ["orange", "mango", "kiwi", "pineapple", "banana"]
thislist.sort()
print(thislist)

thislist = [100, 50, 65, 82, 23]
thislist.sort()
print(thislist)

# Sort Descending
thislist = ["orange", "mango", "kiwi", "pineapple", "banana"]
thislist.sort(reverse = True)
print(thislist)
```

List

```
thislist = [100, 50, 65, 82, 23]
thislist.sort(reverse = True)
print(thislist)

# Customize Sort Function
def myfunc(n):
    return abs(n - 50)

thislist = [100, 50, 65, 82, 23]
thislist.sort(key = myfunc)
print(thislist)
```

List

```
# Case Insensitive Sort
thislist = ["banana", "Orange", "Kiwi", "cherry"]
thislist.sort()
print(thislist)

thislist = ["banana", "Orange", "Kiwi", "cherry"]
thislist.sort(key = str.lower)
print(thislist)

# Copy a List
thislist = ["apple", "banana", "cherry"]
print(thislist)
```

List

```
mylist = thislist.copy()
print(mylist)
a = list([1])
print(a)

mylist[0] = 1
print(mylist)

# Make a copy of a list with the list() method
thislist = ["apple", "banana", "cherry"]
mylist = thislist.copy()
print(mylist)
```

List

```
# Join Two Lists
list1 = ["a", "b", "c"]
list2 = [1, 2, 3]

list3 = list1 + list2
print(list3)

list1 = ["a", "b" , "c"]
list2 = [1, 2, 3]

for x in list2:
    list1.append(x)
print(list1)
```

List

```
list1 = ["a", "b" , "c"]
list2 = [1, 2, 3]

list1.extend(list2)
print(list1)

list1.append(list2)
print(list1)
```

Tuple

```
# Create a Tuple
thistuple = ("apple", "banana", "cherry")
print(thistuple)

# Tuple items are ordered, unchangeable, and allow duplicate values.
thistuple = ("apple", "banana", "cherry", "apple", "cherry")
print(thistuple)

# Tuple Length
thistuple = ("apple", "banana", "cherry")
print(len(thistuple))
```

Tuple

```
thistuple = ("apple",)
print(type(thistuple))
print(thistuple)

#NOT a tuple
thistuple = ("apple")
print(type(thistuple))
print(thistuple)
```

Tuple

```
 # Tuple Items - Data Types
tuple1 = ("apple", "banana", "cherry")
tuple2 = (1, 5, 7, 9, 3)
tuple3 = (True, False, False)

tuple1 = ("abc", 34, True, 40, "male")
print(tuple1)

mytuple = ("apple", "banana", "cherry")
print(type(mytuple))
```

Tuple

```
# tuple() Constructor
thistuple = tuple(("apple", "banana", "cherry")) # note the double round-brackets
print(thistuple)

# Access Tuple Items
thistuple = ("apple", "banana", "cherry")
print(thistuple[1])

# Negative Indexing
thistuple = ("apple", "banana", "cherry")
print(thistuple[-1])
```

Tuple

```
# Range of Indexes
thistuple = ("apple", "banana", "cherry", "orange", "kiwi", "melon", "mango")
print(thistuple[2:5])

thistuple = ("apple", "banana", "cherry", "orange", "kiwi", "melon", "mango")
print(thistuple[:4])

thistuple = ("apple", "banana", "cherry", "orange", "kiwi", "melon", "mango")
print(thistuple[2:])

# Range of Negative Indexes
thistuple = ("apple", "banana", "cherry", "orange", "kiwi", "melon", "mango")
print(thistuple[-4:-1])
```

Tuple

```
# Check if Item Exists
thistuple = ("apple", "banana", "cherry")
if "apple" in thistuple:
    print("Yes, 'apple' is in the fruits tuple")

# Once a tuple is created, you cannot change its values. Tuples are
unchangeable, or immutable as it also is called.
x = ("apple", "banana", "cherry")
x[0] = 'kiwi’
```

Tuple

```
# Change Tuple Values
x = ("apple", "banana", "cherry")
y = list(x)
y[1] = "kiwi"
x = tuple(y)
print(x)

# Add Items
thistuple = ("apple", "banana", "cherry")
y = list(thistuple)
y.append("orange")
thistuple = tuple(y)
print(thistuple)
```

Tuple

```
thistuple = ("apple", "banana", "cherry")
y = ("orange",)
thistuple += y

print(thistuple)


# Remove Items
thistuple = ("apple", "banana", "cherry")
y = list(thistuple)
y.remove("apple")
thistuple = tuple(y)
print(thistuple)
```

Tuple

```
thistuple = ("apple", "banana", "cherry")
print(thistuple)
del thistuple
print(thistuple)

# Unpack Tuples
fruits = ("apple", "banana", "cherry")
(green, yellow, red) = fruits
print(green)
print(yellow)
print(red)
```

Tuple

```
# Using Asterisk
fruits = ("apple", "banana", "cherry", "strawberry", "raspberry")
(green, yellow, *red) = fruits
print(green)
print(yellow)
print(type(red))

fruits = ("apple", "mango", "papaya", "pineapple", "cherry")
(green, *tropic, red) = fruits
print(green)
print(tropic)
print(red)
```

Tuple

```
# Loop Through a Tuple
thistuple = ("apple", "banana", "cherry")
for x in thistuple:
  print(x)

# Loop Through the Index Numbers
thistuple = ("apple", "banana", "cherry")
for i in range(len(thistuple)):
  print(thistuple[i])
```

Tuple

```
# Using a While Loop
thistuple = ("apple", "banana", "cherry")
i = 0
while i < len(thistuple):
  print(thistuple[i])
  i = i + 1

# Join Two Tuples
tuple1 = ("a", "b" , "c")
tuple2 = (1, 2, 3)

tuple3 = tuple1 + tuple2
print(tuple3)
```

Tuple

```
# Multiply Tuples
fruits = ("apple", "banana", "cherry")
mytuple = fruits * 2

print(mytuple)
```

Questions?