python Cheat Sheet

by noeyarp via cheatography.com/25806/cs/6980/

Functions

print()	displays information on the screen.
input()	receives information from the user.
int()	converts a value to an integer.
float()	change number to be decimal number.
str()	a list of number,letter and symbols.
len()	the length of the string.
#	Comment or no effect
long (long integers)	Also called longs, they are integers of unlimited size, written like integers and followed by an uppercase or lowercase L.

Functions (cont)

complex (complex numbers)	are of the form a + bJ, where a and b are floats and J (or j) represents the square root of -1 (which is an imaginary number). The real part of the number is a, and the imaginary part is b. Complex numbers are not used much in Python programming.

Code

```
name = "noey RAWIDA"
print (name.upper())
print (name.lower())
print (name.capitalize())
print (name.title())
```

Conditional

if	A statement that the writer given a condition
else	A statement that can be combined with an if statement.
elif	A statement that allows you to check multiple expressions for TRUE and execute a block of code as soon as one of the conditions evaluates to TRUE.
while	A statement that acting resembles like a loop.

Python Identifier

Definition	is a name used to identify a variable, function, class, module or other object. An identifier starts with a letter A to Z or a to z or an underscore (_) followed by zero or more letters, underscores and digits (0 to 9).

Python Assignment Operators

Operator	Description	**E

Python Assignment Operators

Operator	Description	Example
=	Assigns values from right side operands to left side operand	c = a + b assigns value of a + b into c
+=Add AND	It adds right operand to the left operand and assign the result to left operand	c += a is equivalent to c = c + a
-=Subtract AND	It subtracts right operand from the left operand and assign the result to left operand	c -= a is equivalent to c = c - a

Python Assignment Operators (cont)

*=Multiply AND	It multiplies right operand with the left operand and assign the result to left operand	c = *a is equivalent to c = c* a
/=Divide AND	It divides left operand with the right operand and assign the result to left operand	c /= a is equivalent to c = c / ac /= a is equivalent to c = c / a
%=Modulus AND	It takes modulus using two operands and assign the result to left operand	c %= a is equivalent to c = c % a
**=Exponent AND	Performs exponential (power) calculation on operators and assign value to the left operand	c = **a is equivalent to c = c** a
//=Floor Division	It performs floor division on operators and assign value to the left operand	c //= a is equivalent to c = c // a

Random Code

```
import random
mylist = ['Dog','Fish', 'Cat', 'Bear']
counter = 0
while counter < 10:
      random_item = random.choice (mylist)
      print (random_item)
      counter = counter + 1
```

Number to Binary Code

```
mystring = "hello"
print (mystring)
firstname = input( "what is your first name?")
lastname = input( "what is your last name?")
fullname = firstname + " " + lastname
print (fullname)
letternumber = int(input( " what is letter number? " ))
if letternumber >len(fullname):
    print ( " invalid letter number, try again! " )
else:
    letter = ( fullname[letternumber] )
    print (letter)
    numberletter = int(input( "how many times to print letter " ))
    if numberletter >100:
       print ( " too many letters to print! " )
    else:
        print (letter * numberletter )
```

Number to Binary Code

```
mystring = "hello"
print (mystring)
firstname = input( "what is your first name?")
lastname = input( "what is your last name?")
fullname = firstname + " " + lastname
print (fullname)
letternumber = int(input( " what is letter number? " ))
if letternumber >len(fullname):
    print ( " invalid letter number, try again! " )
else:
    letter = ( fullname[letternumber] )
    print (letter)
    numberletter = int(input( "how many times to print letter " ))
    if numberletter >100:
       print ( " too many letters to print! " )
    else:
        print (letter * numberletter )
```

Addition

string + string	combine together
string + number	crash
number + number	math - addition

Multiplication and Exponents

string * number	combine that string multiple times.
string * string	crash
number * number	math - multiply
string ** string	crash
number ** number	math - exponents
string ** number	crash

Loop

While Loop	Repeats a statement or group of statements while a given condition is TRUE. It tests the condition before executing the loop body.
For Loop	Executes a sequence of statements multiple times and abbreviates the code that manages the loop variable.
Nested Loop	You can use one or more loop inside any another while, for or do..while loop.

Python Variables Types

Number
String
List
Tuple
Dictionary

Data Type Conversion

Function	Description
int(x [,base])	Converts x to an integer. base specifies the base if x is a string.
float(x)	Converts x to a floating-point number.
long(x [,base])	Converts x to a long integer. base specifies the base if x is a string.
str(x)	Converts object x to a string representation.
repr(x)	Converts object x to an expression string.
complex(real [,imag])	Create a complex number.
eval(str)	Evaluates a string and returns an object.
tuple(s)	Converts s to a tuple.
list(s)	Converts s to a list
set(s)	Converts s to a set.

Data Type Conversion (cont)

dict(d)	Creates a dictionary, d must be a sequence of (key,value) tuples.
frozenset (s)	Converts to a frozen set.
chr(x)	Converts an integer to a character.
unichr(x)	Converts an integer to a Unicode character.
ord(x)	Converts a single character to its interger value.
hex(x)	Converts an integer to a hexadecimal string.
oct(x)	Converts an interger to an octal string.

Data Types

Integer	-256, 15
Float	-253.23, 1.253e-10
String	" Hel lo", 'Goodbye', " " " Mul til ine " " "
Boolean	True,False
List	[value, ...]
Tuple	(value, ...)1
Dictionary	{ key: value, ... }
Set	{ value, value, ... }2

python Cheat Sheet

by noeyarp via cheatography.com/25806/cs/6980/

Python Shop Code

```
print ("welcone to our shop")
price=0
size=('s','m','l','xl')
colour=('red','black','white')
sock=('want','not want')
print (size)
shirt = (input('what shirt size do you want?'))
if shirt == ('s'):
        price = price+70
        print( "the price now is",price)
elif shirt ==('m'):
        price = price+80
        print( "the price now is",price)
elif shirt ==('l'):
        price = price+90
        print( "the price now is",price)
elif shirt ==('xl'):
        price = price+100
        print( "the price now is",price)
else:
        print("our shop doesn't have this size.")
print (colour)
shirtcolour= (input('what colour of shirt do you want?'))
if shirtcolour == ('red'):
        price = price+70
        print( "the price now is",price)
elif shirtcolour ==('black'):
```

Python Shop Code (cont)

```
        price = price+80
        print( "the price now is",price)
elif shirtcolour ==('white'):
        price = price+90
        print( "the price now is",price)
else:
        print("our shop don't have this colour")
print (size)
pant = (input('what pant size do you want?'))
if pant == ('s'):
        price = price+70
        print( "the price now is",price)
elif pant ==('m'):
        price = price+80
        print( "the price now is",price)
elif pant ==('l'):
        price = price+90
        print( "the price now is",price)
elif pant ==('xl'):
        price = price+100
        print( "the price now is",price)
else:
        print("our shop doesn't have this size.choose again")
```

Random Choice Code

```
import random
mylist = ['beagle','pomeranian','pug','golden','chihuahua']
score = 0
chances = 3
start_over = 0
random_item = random.choice(mylist)
while chances > 0:
    start_over = 0
    random_item = random.choice(mylist)

    while start_over < 1:
        print ("-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-")
        print ("Guessing Game")
        print ("-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-")
        print("words:", mylist)
        guess = input("Guess a word: ")
        if (guess in mylist):

            if(guess == random_item ):
                print("That's correct!")
                score = score + 100
                print("Score:", score)
                start_over = 2
            else:
                print("Sorry, wrong choice! ")
                chances = int(chances) -1
        else:
            print("Sorry, that is not even in the list")
            chances = int(chances) -1
```

Random Choice Code (cont)

```
        if(chances > 0):
            print("Chances
remaining:",chances)
        else:
            start_over = 2
            print("Game Over! The
word was ", random_item)
            print("Chance
remaining:", chances)
            print("Final score:",
score)
```

Loop list Code

```
def creatlist(quitword) :
    print ('Keep entering words to
add to the list')
    print ('Quit when word =',
quitword)
    mylist = []
    while True:
        user_word = input('Enter a
word to add to the list:')
        if user_word == quitword
            return mylist
        duplicateword = False
        for item in mylist:
            if item == user_word:
                    duplicateword =
True
        for item == user_word:
                duplicateword =
True
        if duplicateword == True:
            print ('Duplicate
Word')
        else:
            mylist.append(user_word
)
userlist = createList("stop")
```

Loop list Code (cont)

```
print(userlist)
```

Math

!=	unequal or not equal to	
==	equal to	example;(a == b) is not true.
<	less than	example;(a < b) is true.
>	more than	example;(a > b) is not true.
<=	less than or equal	example;(a <= b) is true.
>=	more than or equal	example;(a >= b) is not true.
%	modulo or find the remainder	
<>	If values of two operands are not equal, then condition becomes true.	example;(a <> b) is true. This is similar to != operator.

Vocabulary

variable	holds a value and can be changed.
string	a list of characters such as numbers, letters, symbols.

Vocabulary (cont)

integer number	whole number or counting number.
float number	the number in decimal.
syntax	grammar structure of language.
value	the number or the string can be store in valuable.
loop	
module	the text for storing for python code.
blank	
comment	
input	receives information from the user.
code	
print	to show information.
syntax error	make possible to the parse
boolean	true/false

Python Arithmetic Operators

Operator	Description	Example
+ Addition	Adds values on either side of the operator.	a + b = 30

Python Arithmetic Operators (cont)

- Subtraction	Subtracts right hand operand from left hand operand.	a - b = 30
* Multiplication	Multiplies values on either side of the operator.	a * b = 200
/ Division	Divides left hand operand by right hand operand.	b / a = 2
% Modulus	Divides left hand operand by right hand operand and returns remainder.	b % a = 0
** Exponent	Performs exponential (power) calculation on operators.	a ** b = 10 to the power 20
//	Floor Division - The division of operands where the result is the quotient in which the digits after the decimal point are removed.	9//2 = 4 and 9.0/2.0 = 4.0

Statements

If Statement

if expression:
statements
elif expression:
statements
else:
statements

While Loop

while expression:
statements

For Loop

for var in collection:
statements

Counting For Loop

for i in range(s t art, end [, step]):
statements
(start is included; end is not)

Area of circle Code

```
while True:

    user_radius = input("What is the radius?")
    radius = float(user_radius)
    pi = 3.1415
    area= pi radius * 2
    print ("The area of the circle is", area)
```

Code

```
mystring = "hello"
print (mystring)
firstname = input( "what is your first name?")
lastname = input( "what is your last name?")
fullname = firstname + " " + lastname
print (fullname)
letternumber = int(input( " what is letter number? " ))
if letternumber >len(fullname):
    print ( " invalid letter number, try again! " )
else:
    letter = ( fullname[letternumber] )
    print (letter)
    numberletter = int(input( "how many times to print letter " ))
    if numberletter >100:
       print ( " too many letters to print! " )
    else:
        print (letter * numberletter )
```

Print Code

```
name = "noey RAWIDA"
print (name.upper())
print (name.lower())
print (name.capitalize())
print (name.title())
```

List Code

```
shoppinglist = ['tshirt' , 'pants'
, 'socks']
for myvariable in shoppinlist:
    print (myvariable)
print (shoppinglist[1])
for number in range(5):
    print (number)
```

Count Down Code

```
#create a program that receives a
number from the user and count down
from that number on the same line
#recive the number from the user as
a string
user_number= input("enter number")
#convert the user number to an
integer
number = int(user_number)
#setup the countdown string
countdown_string = ""
while number > 0:
    #add the number to the string
    #subtract 1 from the number
    countdown_string =
countdown_string + str(number) +
""
    number = number-1
print (countdown_string)
#output should look like this
# if the user enter 5:
#5 4 3 2 1
#print (countdown_string)
```

Random Code 2

```
import random
intlist = [1,2,3,4]
random_int =
random.choice(intlist)
print(intlist,random_int)
fplist = [1.0, 2.0, 3.0, 4.0]
random_fp = random.choice(fplist)
print(fplist,random_fp)
strlist =
['book','pen','bag','pencil']
random_str =
random.choice(strlist)
print (strlist,random_str)
mylist = [1, 1.0, 'beagle' ]
random_item =
random.choice(mylist)
print(mylist,random_item)
myvar1 = 1
myvar2 = 2
myvar3 = 3
varlist =[myvar1, myvar2, myvar3]
random_var =
random.choice(varlist)
print(varlist,random_var)
```