Cheatography

Python Functions Cheat Sheet

by atinfosec (KilGrave) via cheatography.com/69386/cs/17507/

Math Module Functions

ceil(x)	Returns the smallest integer greater than or equal to x
copysign(x, y)	Returns x with the sign of y
fabs(x)	Returns the absolute value of x
factorial(x)	Returns the factorial of x
floor(x)	Returns the largest integer less than or equal to x
fmod(x, y)	Returns the remainder when x is divided by y
isfinite(x)	Returns True if x is neither an infinity nor a NaN (Not a Number)
isinf(x)	Returns True if x is a positive or negative infinity
isnan(x)	Returns True if x is a NaN
ldexp(x, i)	Returns x *(2*i)
modf(x)	Returns the fractional and integer parts of x
exp(x)	Returns e**x
expm1(x)	Returns e**x – 1
log(x[, base])	Returns the logarithm of x to the base (defaults to e)
log2(x)	Returns the base-2 logarithm of x
log10(x)	Returns the base-10 logarithm of x
pow(x, y)	Returns x raised to the power y
sqrt(x)	Returns the square root of x
acos(x)	Returns the arc cosine of x
asin(x)	Returns the arc sine of x
atan(x)	Returns the arc tangent of x
atan2(y, x)	Returns atan(y / x)
cos(x)	Returns the cosine of x
hypot(x, y)	Returns the Euclidean norm, sqrt(*xx + yy*)
sin(x)	Returns the sine of x
tan(x)	Returns the tangent of x
degrees(x)	Converts angle x from radians to degrees
radians(x)	Converts angle x from degrees to radians
gamma(x)	Returns the Gamma function at x
lgamma(x)	Returns the natural logarithm of the absolute value of the Gamma function at x
pi	Mathematical constant, the ratio of circumference of a circle to it's diameter (3.14159...)

Math Module Functions (cont)

e	mathematical constant e (2.71828...)

*** import math and use math.fun()**

Sets Functions

S.add(e)	Adds the element **e** to the set **S**
S1.update(S2)	Adds the items specified in the set **S2** to the set **S1**
S.remove(e)	Remove the element **e** from the set **S**
S.pop()	Removes any element from the set **S**
S.clear()	Remove all element from the set **S**
S.copy()	Creates a copy of the set **S**
S1.union(S2)	Returns a set containing elements from both **S1** and **S2**
S1.intersection(S2)	Returns a set containing elements common in set **S1** and **S2**
S1.difference(S2)	Returns a set containing elements in set **S1** but not in **S2**
S1.symmetric_difference(S2)	Returns a set containing elements which are in one of the either sets **S1** and **S2**, but not in both

String Functions

S.count(str)	Counts the number of times string **str** occurs in string **S**
S.find(str)	Returns index of first occurrence of string **str** in string **S**, and **-1** if **str** is not present int string **S**
S.rfind(str)	Returns index of last occurrence of string **str** in string **S**, and **-1** if **str** is not present in string **S**
S.capitalize()	Returns a string that has first letter of the string **S** in uppercase and rest of the characters in lowercase
S.title()	Returns a string that has first letter of every word in the string **S** in uppercase and rest of the characters in lowercase
S.lower()	Returns a string that has all uppercase characters in string **S** converted into lowercase characters

String Functions (cont)

S.upper()	Returns a string that has all lower characters in string **S** converted into uppercase characters
S.swapcase()	Returns a string that has all lowercase characters in string **S** converted into uppercase characters and vice versa
S.isupper()	Returns **True** if all alphabets in string **S** are in uppercase,else **False**
S.islower()	Returns **True** if all alphabets in string **S** are in lowercase,else **False**
S.istitle()	Returns **True** if string **S** is in titlecase
S.replace(str1,str2)	Returns a string that has every occurrence of string **str1** in **S** replaced by with the occurrence of string **str2**
S.strip()	Returns a string that has whitespaces in **S** removed from start and end
S.lstrip()	Returns a string that has whitespaces in **S** removed from start
S.rstrip()	Returns a string that has whitespaces in **S** removed from end
S.split(delimeter)	Returns a list formed by splitting the string **S** into various substring. The delimeter is used to mark the split points
S.partition(delimeter)	Partitions the string **S** into two parts base on **delimeter** and returns a tuple comprising of string before **delimeter**
S.join(sequence)	Returns a string comprising of elements of the sequence separated by delimeter **S**
S.isspace()	Returns **True** if all characters in string **S** comprise of whitespace characters only,i.e. **' '**, **'\n'**, **'\t'** else **False**
S.isalpha()	Returns **True** if all characters in string **S** comprise of alphabets only, else **False**

String Functions (cont)

S.isdigit()	Returns **True** if all characters in string **S** comprise of digits only, else **False**
S.isalnum()	Returns **True** if all characters in string **S** comprise of alphabets and digits only, else **False**
S.startswith(str)	Returns **True** if string **S** starts with string **str**,else **False**
S.endswith(str)	Returns **True** if string **S** ends with string **str**,else **False**
S.encode(str)	Returns **S** in encoded format according to the given encoding scheme
S.decode(str)	Returns the decoded string **S** according to the given encoding scheme

List

L.append(e)	Adds the element **e** to the end of the list **L**
L.extend(L2)	Adds the items specified in the list **L2** at the end of the list **L**
L.remove(e)	Remove the element **e** from the list **L**
L.pop(i)	Removes the element specified at index **i** ***from the list*** *L**
L.count(e)	Returns count of occurrence of element **e** in list **L**
L.index(e)	Returns index of element **e** from list **L**
L.insert(i,e)	Returns element **e** at the index **i** in list **L**
L.sort()	Sorts the elements of the list **L**
L.reverse()	Reverses the order elements in list **L**

File Handling

open(filename,mode)	Open a **file** and store it as an object
file.close()	Close a file which is opened
file.read()	Read whole data from **file**
file.readline()	Read a line from **file**
file.readlines()	Read all the lines in a list from **file**
file.write('data')	Write data in a file

*** Mode can be 'r', 'w' and 'a'**