

ITProToday Python Commands Cheat Sheet

Action	Command
Strings	
Define a string	<code>string = "string value"</code>
Print a string	<code>string = "string value"</code> <code>print (string)</code>
Concatenate strings	<code>string1 = "some value"</code> <code>string2 = "another value"</code> <code>concatenated_strings = string1 + string2</code>
Variables	
Set a variable as integer	<code>var = 1</code>
Set a variable as string	<code>var = "value"</code>
Math	
Add numbers	<code>num1 = 1</code> <code>num2 = 2</code> <code>sum = int(num1) + int(num2)</code>
Subtract numbers	<code>num1 = 1</code> <code>num2 = 2</code> <code>difference = num1 - num2</code>
Multiply numbers	<code>num1 = 1</code> <code>num2 = 2</code> <code>product = num1 * num2</code>
Divide numbers	<code>num1 = 1</code> <code>num2 = 2</code> <code>quotient = num1 / num2</code>
Compare numbers	<code>num1 = 1</code> <code>num2 = 2</code> <code>print(num1 > num2) # will output False</code> <code>print(num1 < num2) # will output True</code> <code>print(num1 == num2) # will output False</code> <code>print(num1 != num2) #will output True</code>
Lists	
Create a list	<code>list = [1, 2, 3]</code>
Add item to list	<code>list = [1, 2, 3]</code> <code>list.append(4)</code>
Remove item from list	<code>list = [1, 2, 3]</code> <code>list.remove(1)</code>
Sort a list	<code>list = [4, 1, 9, 3]</code> <code>list.sort() # list values are now 1, 3, 4, 9</code>
Comments	
Insert a comment	<code>x = 1 # code that precedes the # sign will be interpreted</code> <code># none of this line will be interpreted</code> Note that Python doesn't support multi-line comments (unless you preface each line with a # sign)
Files	
Open a file in read-only mode	<code>file = open("/path/to/file", "r")</code>
Open a file in read-write mode	<code>file = open("/path/to/file", "r+")</code> Note: You can also use "w+" in place of "r+" for read-write mode, but "w+" treats the file a bit differently; check out the open() function documentation for details.
Open a file in append mode	<code>file = open("/path/to/file", "a")</code>
Command line arguments	
Read a command line argument	<code>import sys</code> <code>print(sys.argv[0]) # prints first command-line argument (which is usually the name of the Python program)</code> <code>print(sys.argv[1]) # prints second command-line argument (i.e., the first argument to the program)</code>
Searches and regexes	
Search for regex in string	<code>import re</code> <code>string = "This is your string!"</code> <code>search_result = re.search("^This.*", string)</code>
Search for string in list	<code>list = ["a", "b", "c"]</code> <code>matches = [match for match in list if "a" in match]</code>
Conditionals	
Create a for loop	<code>list = ["a", "b", "c"]</code> <code>for value in list:</code> <code>print(value)</code>
Create a while loop	<code>i = 10</code> <code>while i < 10:</code> <code>print(i)</code> <code>i += 1</code>
Create if and elif statements	<code>a = 1</code> <code>b = 2</code> <code>if b > a:</code> <code>print("b is greater than a")</code> <code>elif a > b:</code> <code>print(a is greater than a)</code> <code>elif a == b:</code> <code>print("a and b are the same")</code>
Functions	
Define a function	<code>def function_name():</code> <code>def function_name():</code> <code>print("Some text")</code> <code>print("Some more text")</code>
Call a function	<code>function_name()</code>