

# Python Cheat Sheet

“A puzzle a day to learn, code, and play” → Visit [finxter.com](https://finxter.com)

| Keyword        | Description                                                                                                                                                       | Code example                                                                                                                                              |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| False, True    | Boolean data types                                                                                                                                                | <code>False == (1 &gt; 2), True == (2 &gt; 1)</code>                                                                                                      |
| None           | Empty value constant                                                                                                                                              | <pre>def f():<br/>    x = 2<br/>f() == None # True</pre>                                                                                                  |
| and, or, not   | Logical operators:<br>“x and y” → both x and y must be True<br>“x or y” → either x or y must be True<br>“not x” → x must be false                                 | <pre>x, y = True, False<br/>x or y == True      # True<br/>x and y == False   # True<br/>not y == True       # True</pre>                                 |
| break          | Ends loop prematurely                                                                                                                                             | <pre>while(True):<br/>    break # no infinite loop<br/>print("hello world")</pre>                                                                         |
| continue       | Finishes current loop iteration                                                                                                                                   | <pre>while(True):<br/>    continue<br/>print("43") # dead code</pre>                                                                                      |
| class          | Defines a new class → a real-world concept (object oriented programming)                                                                                          | <pre>class beer:<br/>    x = 1.0 # litre<br/>    def drink():<br/>        x = 0.0</pre>                                                                   |
| def            | Defines a new function                                                                                                                                            |                                                                                                                                                           |
| if, elif, else | Conditional program execution: program starts with “if” branch, tries all “elifs” branches, and finishes with “else” branch (until one branch evaluates to True). | <pre>x = int(input("your value: "))<br/>if x &gt; 3:<br/>    print("Big")<br/>elif x == 3:<br/>    print("Medium")<br/>else:<br/>    print("Small")</pre> |
| for, while     | <pre># For loop declaration<br/>for i in [0,1,2]:<br/>    print(i)</pre>                                                                                          | <pre># The same while loop<br/>j = 0<br/>while j &lt; 3:<br/>    print(j)<br/>    j = j + 1</pre>                                                         |
| in             | Tests if element is in sequence                                                                                                                                   | <code>42 in [2, 39, 42] # True</code>                                                                                                                     |
| is             | Tests whether both element point to the same object                                                                                                               | <pre>3 is 3 # True<br/>[3] is [3] # False</pre>                                                                                                           |
| lambda         | Function with no name                                                                                                                                             | <pre>f = lambda x: x + 3<br/>f(3) # return value is 6</pre>                                                                                               |
| return         | Defines the result of a function                                                                                                                                  | <pre>def incrementor(x):<br/>    return x + 1<br/>incrementor(4) # returns 5</pre>                                                                        |