

Cheat Sheet - Day 1

Introduction to Data Analysis with Python

<https://github.com/janekfleper/Workshop-Konstanz-2024>

Basics

<code>x = 1</code> <code>y = "abc"</code>	Assign a value to a <i>variable</i>
<code>print("hello")</code> <code>print(x, 2, 3)</code>	Print one or multiple values or variables
<code>list(y)</code> <code>len([0, 1, 2])</code>	A <i>function</i> is called with (optional) arguments
<code>y.count("a")</code> <code>"abc".upper()</code>	A <i>method</i> is called on a value or on a variable
<code># a comment</code>	A <code>#</code> starts a <i>comment</i> that will not be evaluated
<code>type(x)</code>	Get the <i>type</i> of a value or of a variable

Data types

<code>"hello", 'abc',</code> <code>"0.9", str(123)</code>	A <i>string</i> is a sequence of characters in quotation marks
<code>12, -4, int("5")</code>	An <i>integer</i> is a number without a decimal part
<code>0.9, -3.1415,</code> <code>float("-0.1")</code>	A <i>float</i> is a number with a decimal part
<code>True, False,</code> <code>bool(0), x < 1</code>	A <i>boolean</i> can only take the values <code>True</code> or <code>False</code>
<code>[0, "abc", 0.1]</code> <code>list("hello")</code>	A <i>list</i> is a mutable, sorted collection of values
<code>{"a": 1, "b": 2}</code> <code>dict(a=1, b=2)</code>	A <i>dictionary</i> is a mutable collection of key-value pairs
<code>(0, "0.9", True)</code> <code>tuple([0, 1, 2])</code>	A <i>tuple</i> is an immutable, sorted collection of values

Strings

<code>s = "hello"</code>	Initialize a string
<code>len(s)</code>	Get the length of a string
<code>s[2], s[1:-1]</code>	Get characters from a string
<code>s[::-1]</code>	Get a string in reverse order
<code>"he" in s</code>	Check if a string contains a substring
<code>s + "abc"</code>	Concatenate/append strings
<code>s * 10</code>	Repeat a string <i>N</i> times
<code>s.isalpha()</code> <code>s.isnumeric()</code>	Check properties of the characters in a string

Lists

<code>x = [1, 2, 3]</code>	Initialize a list with values
<code>len(x)</code>	Get the length of a list
<code>x[2], x[0:-1]</code>	Get values from a list
<code>x[::-1]</code>	Get a list in reverse order
<code>x[0] = 3</code>	Change a value in a list
<code>x.append(4)</code>	Append a value to a list
<code>x.extend([4, 5])</code>	Extend a list by another list
<code>y = x + [4, 5]</code>	Add lists to create a new list

Dictionaries

<code>d = {"a": 1, "b": 2}</code> <code>d = dict(a=1, b=2)</code>	Create a dictionary with key-value pairs
<code>d["a"]</code>	Get the value of a key from a dictionary
<code>d["a"] = 0</code> <code>d["c"] = 3</code>	Update/add a key-value pair in/to a dictionary
<code>d.keys()</code> <code>d.values()</code> <code>d.items()</code>	Get all keys, all values or all key-value pairs in a dictionary

for loops

<code>range(n)</code> <code>range(m, n)</code>	Create a sequence from 0 / m to n-1 in steps of 1
<code>for i in range(n):</code> <code>print(i)</code>	Iterate over the integer values in a sequence
<code>for c in "hello":</code> <code>print(c)</code>	Iterate over the characters in a string
<code>for v in [0, -1, 2]:</code> <code>print(v)</code>	Iterate over the values in a list
<code>for k in d.keys():</code> <code>print(k)</code>	Iterate over the keys in a dictionary
<code>x = [1, -5, 3, 0]</code> <code>y = [v+1 for v in x]</code>	Create a new list with a list comprehension

Conditions and if - elif - else statements

<code>a == b, a != b</code> <code>a > b, a >= b</code> <code>a < b, a <= b</code>	Compare two values, the result will be <code>True</code> or <code>False</code>
<code>if a < 0:</code> <code> a = -a</code> <code>elif a < 10:</code> <code> a = a - 10</code> <code>else:</code> <code> a = a * 5</code>	Check a condition with an <code>if</code> statement. Check multiple conditions with <code>if</code> and <code>elif</code> (as many as you want). The (optional) <code>else</code> block handles all other cases.

Jupyter shortcuts

<code>Enter</code> / <code>Esc</code>	Start/exit the edit mode
<code>Shift</code> + <code>Enter</code>	Run cell(s) and select next
<code>A</code> / <code>B</code>	Insert new cell above/below
<code>X</code> , <code>C</code> , <code>V</code>	Cut, copy or paste cell(s)
<code>Z</code> / <code>Shift</code> + <code>Z</code>	Undo/redo cell operation