nums = Series([7,8,9], index=[-1,0,1])

x = Series({"A":1, "B":2, "C":3})

y = Series({"A":2, "C":12, "D":4})

|  |  |
| --- | --- |
| **Expression** | **Result(s)** |
| nums[0] |  |
| nums.loc[0], nums.iloc[0] |  |
| nums.loc[-1], nums.iloc[-1] |  |
| x / y |  |

s = Series(["A", "B", "C", "D"])

**note**: Series.loc[X] looks for label X in the **index**. Series.iloc[X] looks for the **int position** X. These names are confusing. iloc supports negative indexing.

letters = Series(["x", "y", "z"], index=[1, 0, 3])

|  |  |
| --- | --- |
| **Expression** | **Result(s)** |
| s[-1] |  |
| s[-2:] |  |
| s + s |  |
| letters[0] |  |
| s + letters |  |
| s[1:] + s[:-1] |  |

v = Series([-1, 1, 200, 191, 4])

|  |  |
| --- | --- |
| **Expression** | **Result(s)** |
| v < 0 |  |
| v \* v == 1 |  |
| v[v > 100] |  |
| v[v % 2 == 0] |  |
| v[(v>0) & (v<100)] |  |

|  |  |
| --- | --- |
| **Code:** | **storms.csv:** |
| path = "storms.csv"tab = pd.read\_csv(path)map = DataFrame({ "code": ["o","p","a"], "where": ["other","Pacific","Atlantic"]}) | name,year,type,speed,placealice,2016,tornado,100,obob,2016,hurricane,200,pcindy,2017,tornado,150,odan,2018,tornado,300,oeve,2018,hurricane,250,a |

|  |  |
| --- | --- |
| **Expression** | **Result(s)** |
| map["code"] |  |
| map.code |  |
| type(map.code), type(map.where) |  |
| tab.year.mean() |  |
| tab.year == 2018 |  |
| tab.name[tab.year == 2018] |  |
| map["where"] == "Atlantic" |  |
| b = map["where"] == "other"code = map.code[b].item()nms = tab.name[tab.place==code] | # what are b, code, nms? |

**note**: s.COL is a shortcut for s["COL"], unless COL collides with a method name

**also**: when a Series s contains exactly one one item, s.item() extracts it

|  |  |
| --- | --- |
| **Expression** | **Result(s)** |
| tab.loc[0] |  |
| tab.loc[4, "type"] |  |
| map.loc[0,"where"] = "mainland"place = map["where"][0] | # what is place? |
| tab.loc[:, "speed"] += 1col = tab.speed | # what is col? |