

2.1 - Indexes

April 11, 2017

```
In [1]: import pandas as pd
import numpy as np
```

```
In [2]: df = pd.concat([pd.DataFrame(np.random.rand(100, 7))] * 2)
```

```
In [3]: df.tail()
```

```
      0         1         2         3         4         5         6
95  0.382644  0.347368  0.178788  0.695637  0.128976  0.051965  0.973976
96  0.406776  0.790750  0.887662  0.493536  0.510413  0.259301  0.088332
97  0.859052  0.349884  0.174759  0.310978  0.198675  0.716746  0.475429
98  0.615220  0.883569  0.311990  0.401561  0.590069  0.177892  0.773087
99  0.126455  0.051023  0.754828  0.335267  0.927782  0.402497  0.603498
```

```
In [4]: df.loc[0]
```

```
      0         1         2         3         4         5         6
0  0.382934  0.517149  0.663191  0.869145  0.142919  0.462571  0.091122
0  0.382934  0.517149  0.663191  0.869145  0.142919  0.462571  0.091122
```

```
In [5]: df.loc[11] = range(7)
```

ValueError

Traceback (most recent call last)

```
<ipython-input-5-0de25612adeb> in <module>()
----> 1 df.loc[11] = range(7)
```

```
   /home/pietro/nobackup/repo/pandas/pandas/core/indexing.py in __setitem__(self,
176         key = com._apply_if_callable(key, self.obj)
177         indexer = self._get_setitem_indexer(key)
--> 178         self._setitem_with_indexer(indexer, value)
```

```

179
180     def _has_valid_type(self, k, axis):

/home/pietro/nobackup/repo/pandas/pandas/core/indexing.py in _setitem_with
616         self.obj._consolidate_inplace()
617         self.obj._data = self.obj._data.setitem(indexer=indexer,
--> 618                                             value=value)
619         self.obj._maybe_update_cacher(clear=True)
620

/home/pietro/nobackup/repo/pandas/pandas/core/internals.py in setitem(self,
3196
3197     def setitem(self, **kwargs):
-> 3198         return self.apply('setitem', **kwargs)
3199
3200     def putmask(self, **kwargs):

/home/pietro/nobackup/repo/pandas/pandas/core/internals.py in apply(self, f
3084
3085         kwargs['mgr'] = self
-> 3086         applied = getattr(b, f)(**kwargs)
3087         result_blocks = _extend_blocks(applied, result_blocks)
3088

/home/pietro/nobackup/repo/pandas/pandas/core/internals.py in setitem(self,
699         indexer.dtype == np.bool_ and
700         len(indexer[indexer]) == len(value)):
--> 701         raise ValueError("cannot set using a list-like indexe
702                             "with a different length than the
703

```

ValueError: cannot set using a list-like indexer with a different length than the

```
In [6]: df.loc[11]
```

```

      0      1      2      3      4      5      6
11  0.07784  0.839468  0.444689  0.927046  0.728596  0.566416  0.372417
11  0.07784  0.839468  0.444689  0.927046  0.728596  0.566416  0.372417

```

```
In [7]: df.loc[101]
```

```

-----

KeyError                                Traceback (most recent call last)

/home/pietro/nobackup/repo/pandas/pandas/core/indexing.py in _has_valid_type
1432         if key not in ax:
-> 1433             error()
1434         except TypeError as e:

/home/pietro/nobackup/repo/pandas/pandas/core/indexing.py in error()
1427         raise KeyError("the label [%s] is not in the [%s]" %
-> 1428                        (key, self.obj._get_axis_name(axis)))
1429

```

KeyError: 'the label [101] is not in the [index]'

During handling of the above exception, another exception occurred:

```

KeyError                                Traceback (most recent call last)

<ipython-input-7-de699f2a8324> in <module>()
----> 1 df.loc[101]

/home/pietro/nobackup/repo/pandas/pandas/core/indexing.py in __getitem__(self, key)
1325         else:
1326             key = com._apply_if_callable(key, self.obj)
-> 1327             return self._getitem_axis(key, axis=0)
1328
1329     def _is_scalar_access(self, key):

/home/pietro/nobackup/repo/pandas/pandas/core/indexing.py in _getitem_axis(self, key, axis)
1548
1549     # fall thru to straight lookup
-> 1550     self._has_valid_type(key, axis)
1551     return self._get_label(key, axis=axis)
1552

/home/pietro/nobackup/repo/pandas/pandas/core/indexing.py in _has_valid_type(self, key, axis)
1439         raise
1440     except:

```

```

-> 1441             error()
    1442
    1443             return True

/home/pietro/nobackup/repo/pandas/pandas/core/indexing.py in error()
1426                 "key")
1427                 raise KeyError("the label [%s] is not in the [%s]" %
-> 1428                               (key, self.obj._get_axis_name(axis)))
    1429
    1430             try:

```

```

KeyError: 'the label [101] is not in the [index]'

```

```
In [8]: df.loc[101] = range(7)
```

```
In [9]: df.tail()
```

	0	1	2	3	4	5	6
96	0.406776	0.790750	0.887662	0.493536	0.510413	0.259301	0.088332
97	0.859052	0.349884	0.174759	0.310978	0.198675	0.716746	0.475429
98	0.615220	0.883569	0.311990	0.401561	0.590069	0.177892	0.773087
99	0.126455	0.051023	0.754828	0.335267	0.927782	0.402497	0.603498
101	0.000000	1.000000	2.000000	3.000000	4.000000	5.000000	6.000000

```
In [10]: a = np.zeros(shape=(5,4))
```

```
In [11]: a
```

```

Out[11]: array([[ 0.,  0.,  0.,  0.],
                [ 0.,  0.,  0.,  0.],
                [ 0.,  0.,  0.,  0.],
                [ 0.,  0.,  0.,  0.],
                [ 0.,  0.,  0.,  0.]])

```

```
In [12]: a[4]
```

```

Out[12]: array([ 0.,  0.,  0.,  0.])

```

```
In [13]: a[5] = [1,2,3,4]
```

```

IndexError

```

```

Traceback (most recent call last)

```

```
<ipython-input-13-9524da7d37d0> in <module>()
----> 1 a[5] = [1,2,3,4]
```

IndexError: index 5 is out of bounds for axis 0 with size 5

```
In [14]: %timeit -n 1 -r 1 df.loc[11]
```

1 loop, best of 1: 713 μ s per loop

```
In [15]: %timeit -n 1 -r 1 df.loc[101]
```

1 loop, best of 1: 334 μ s per loop

```
In [16]: ddf = df.reset_index()
```

```
In [17]: ddf.tail()
```

	index	0	1	2	3	4	5	\
196	96	0.406776	0.790750	0.887662	0.493536	0.510413	0.259301	
197	97	0.859052	0.349884	0.174759	0.310978	0.198675	0.716746	
198	98	0.615220	0.883569	0.311990	0.401561	0.590069	0.177892	
199	99	0.126455	0.051023	0.754828	0.335267	0.927782	0.402497	
200	101	0.000000	1.000000	2.000000	3.000000	4.000000	5.000000	
		6						
196		0.088332						
197		0.475429						
198		0.773087						
199		0.603498						
200		6.000000						

```
In [18]: ddf.index.is_unique
```

```
Out[18]: True
```

```
In [19]: %timeit -n 1 -r 1 df.loc[11]
```

1 loop, best of 1: 610 μ s per loop

```
In [20]: %timeit -n 1 -r 1 ddf.loc[11]
```

1 loop, best of 1: 373 μ s per loop

```
In [21]: %timeit -n 1 -r 1 df.loc[101]
```

1 loop, best of 1: 253 μ s per loop

```
In [22]: %timeit -n 1 -r 1 ddf.loc[101]
```

1 loop, best of 1: 301 μ s per loop

```
In [23]: %timeit -n 1 -r 1 df.loc[[11]]
```

1 loop, best of 1: 837 μ s per loop

```
In [24]: %timeit -n 1 -r 1 ddf.loc[[11]]
```

1 loop, best of 1: 792 μ s per loop