



AniPython

pandas选择数据

iloc

df.iloc[row_indexer, column_indexer]

		0	1	2	3	
			A	B	C	D
0	X	1	2	3	4	
1	Y	5	6	7	8	
2	Z	9	10	11	12	

- df.iloc[0], df.loc[0, 1]
- df.iloc[[0, 1], [1, 3]]
- df.iloc[[True, False, True],
[True, True, True, False]]
- df.iloc[0:2, 1:3]
- df.iloc[callable1, callable2]

传入整个df作为参数
返回值是前面的任一种情况

df.iloc[row_indexer, column_indexer]

		0	1	2	3	
			A	B	C	D
0	X	1	2	3	4	
1	Y	5	6	7	8	
2	Z	9	10	11	12	

- df.iloc[0], df.loc[0, 1]
- df.iloc[[0, 1], [1, 3]]
- df.iloc[[True, False, True],
[True, True, True, False]]
- df.iloc[0:2, 1:3]
- df.iloc[callable1, callable2]

传入整个df作为参数
返回值是前面的任一种情况

df.iloc[row_indexer, column_indexer]

		0	1	2	3	
			A	B	C	D
0	X	1	2	3	4	
1	Y	5	6	7	8	
2	Z	9	10	11	12	

- df.iloc[0], df.loc[0, 1]
- df.iloc[[0, 1], [1, 3]]
- df.iloc[[True, False, True],
[True, True, True, False]]
- df.iloc[0:2, 1:3]
- df.iloc[callable1, callable2]

传入整个df作为参数
返回值是前面的任一种情况

df.iloc[row_indexer, column_indexer]

		0	1	2	3	
			A	B	C	D
0	X	1	2	3	4	
1	Y	5	6	7	8	
2	Z	9	10	11	12	

- df.iloc[0], df.loc[0, 1]
- df.iloc[[0, 1], [1, 3]]
- df.iloc[[True, False, True],
[True, True, True, False]]
- df.iloc[0:2, 1:3]
- df.iloc[callable1, callable2]

传入整个df作为参数
返回值是前面的任一种情况

df.iloc[row_indexer, column_indexer]

		0	1	2	3	
			A	B	C	D
0	X	1	2	3	4	
1	Y	5	6	7	8	
2	Z	9	10	11	12	

- df.iloc[0], df.loc[0, 1]
- df.iloc[[0, 1], [1, 3]]
- df.iloc[[True, False, True],
[True, True, True, False]]
- df.iloc[0:2, 1:3]
- df.iloc[callable1, callable2]

传入整个df作为参数
返回值是前面的任一种情况

df.iloc[row_indexer, column_indexer]

		0	1	2	3	
			A	B	C	D
0	X	1	2	3	4	
1	Y	5	6	7	8	
2	Z	9	10	11	12	

- df.iloc[0], df.loc[0, 1]
- df.iloc[[0, 1], [1, 3]]
- df.iloc[[True, False, True],
[True, True, True, False]]
- df.iloc[0:2, 1:3]
- df.iloc[callable1, callable2]

传入整个df作为参数
返回值是前面的任一种情况

df.iloc[row_indexer, column_indexer]

		0	1	2	3	
			A	B	C	D
0	X	1	2	3	4	
1	Y	5	6	7	8	
2	Z	9	10	11	12	

- df.iloc[0], df.loc[0, 1]
- df.iloc[[0, 1], [1, 3]]
- df.iloc[[True, False, True],
[True, True, True, False]]
- df.iloc[0:2, 1:3]
- df.iloc[callable1, callable2]

传入整个df作为参数
返回值是前面的任一种情况

选择单个值时

`.loc == .at`

`.iloc == .iat`

		0	1	2	3	
			A	B	C	D
0	X	1	2	3	4	
1	Y	5	6	7	8	
2	Z	9	10	11	12	

```
df.loc['X', 'B'] == df.at['X', 'B']
```

```
df.iloc[1, 2] == df.iat[1, 2]
```


		0	1	2	3
		A	B	C	D
0	X	1	2	3	4
1	Y	5	6	7	8
2	Z	9	10	11	12

loc: location

.loc : 标签(label)

.iloc : 整数位置(integer position)

注意: 切片索引中 `loc` 和 `iloc` 的区别

`df.loc['X':'Y']`

结果包括 'Y'

`df.iloc[0:2]`

结果不包括 2

df.loc[row_indexer, column_indexer]

	A	B	C	D
X	1	2	3	4
Y	5	6	7	8
Z	9	10	11	12

- 单独的位置: df.loc['X'], df.loc['X', 'B']
- 位置的列表: df.loc[['X', 'Y'], ['B', 'D']]
- bool的列表: df.loc[[True, False, True],
[True, True, True, False]]
- 位置的切片: df.loc['X':'Y', 'B':'D']
- callable: df.loc[callable1, callable2]

传入整个df作为参数
返回值是前面的任一种情况

df.iloc[row_indexer, column_indexer]

		0	1	2	3
		A	B	C	D
0	X	1	2	3	4
1	Y	5	6	7	8
2	Z	9	10	11	12

- 单独的位置: df.iloc[0], df.loc[0, 1]
- 位置的列表: df.iloc[[0, 1], [1, 3]]
- bool的列表: df.iloc[[True, False, True],
[True, True, True, False]]
- 位置的切片: df.iloc[0:2, 1:4]
- callable: df.iloc[callable1, callable2]

传入整个df作为参数
返回值是前面的任一种情况

df.loc[row_indexer, column_indexer]

	A	B	C	D
X	1	2	3	4
Y	5	6	7	8
Z	9	10	11	12

- df.loc['X'], df.loc['X', 'B']
- df.loc[['X', 'Y'], ['B', 'D']]
- df.loc[[True, False, True],
[True, True, True, False]]
- df.loc['X':'Y', 'B':'D']
- df.loc[callable1, callable2]

传入整个df作为参数
返回值是前面的任一种情况

类型	索引
Series	s.loc[indexer] s.iloc[indexer]
DataFrame	df.loc[row_indexer, column_indexer] df.iloc[row_indexer, column_indexer]

谢谢观看

