



AniPython

时间序列 索引

DatetimeIndex



```
import pandas as pd
import numpy as np

index = pd.date_range(
    '2020-01-01',
    '2020-01-05',
    freq='D'
)

data = np.arange(10).reshape(5, 2)

df = pd.DataFrame(
    data,
    index=index,
    columns=['A', 'B']
)
```

	A	B
2020-01-01	0	1
2020-01-02	2	3
2020-01-03	4	5
2020-01-04	6	7
2020-01-05	8	9



# 时间序列索引优点

## 1. 获取日期范围(切片)非常快

```
df.loc['2020-01-02':'2020-01-04']
```

	A	B
2020-01-01	0	1
2020-01-02	2	3
2020-01-03	4	5
2020-01-04	6	7
2020-01-05	8	9



# 时间序列索引优点

## 2. shift(), tshift()等移位操作快

```
df.shift(1)
```

	A	B
2020-01-01	0	1
2020-01-02	2	3
2020-01-03	4	5
2020-01-04	6	7
2020-01-05	8	9



# 时间序列索引优点

2. shift(), tshift()等移位操作快

```
df.shift(1)
```

	A	B
2020-01-01		
2020-01-02	0	1
2020-01-03	2	3
2020-01-04	4	5
2020-01-05	6	7
	8	9



# 时间序列索引优点

2. shift(), tshift()等移位操作快

```
df.shift(1)
```

	A	B
2020-01-01	NaN	NaN
2020-01-02	0	1
2020-01-03	2	3
2020-01-04	4	5
2020-01-05	6	7
	8	9



# 时间序列索引优点

## 2. shift(), tshift()等移位操作快

```
df.shift(1)
```

	A	B
2020-01-01	NaN	NaN
2020-01-02	0	1
2020-01-03	2	3
2020-01-04	4	5
2020-01-05	6	7



# 时间序列索引优点

## 3. 相同频率的

DatetimeIndex对象

合并非常快

```
df1.join(df2)
```

	A
2020-01-01	0
2020-01-02	2
2020-01-03	4
2020-01-04	6
2020-01-05	8

	B
2020-01-01	1
2020-01-02	3
2020-01-03	5
2020-01-04	7
2020-01-05	9



# 时间序列索引优点

## 3. 相同频率的

DatetimeIndex对象

合并非常快

```
df1.join(df2)
```

	A	B
2020-01-01	0	1
2020-01-02	2	3
2020-01-03	4	5
2020-01-04	6	7
2020-01-05	8	9



# 时间序列索引优点

## 4. 通过年份, 月份等属性 快速访问

```
df['2019']  
df['2020']  
df['2020-01']  
df['2020-02-02']  
df['2019-02':'2020-01']
```

	A
2019-01-01	1
2019-02-02	2
2020-01-01	3
2020-01-02	4
2020-02-02	5



# 时间序列索引优点

4. 通过年份, 月份等属性  
快速访问

注意1

```
# 取值方式冲突  
# 优先取"列"  
df['2020-01']
```

	2020-01
2019-01-01	1
2019-02-02	2
2020-01-01	3
2020-01-02	4
2020-02-02	5



# 时间序列索引优点

4. 通过年份, 月份等属性  
快速访问

注意2

# 乱序切片

```
df['2020-02-01':'2020-04-01']
```

	A
2020-01-01	1
2020-02-01	2
2020-05-01	3
2020-04-01	4
2020-03-01	5



# 时间序列索引优点

**DatetimeIndex** of  
advanced time se

中文 (简体)

🔊 注意 虽然熊猫不会强迫您具有已排序的日期索引，但是如果日期未排序，则其中某些方法可能会出现意外或错误的行为。

See also

**Reindexing me**

扩展程序选项

更多 »

Note

While pandas does not force you to have a sorted date index, some of these methods may have unexpected or incorrect behavior if the dates are unsorted.

2020-03-01

5