Visualizing Time Series Data

Let's go through a few key points of creatng nice time visualizations!

In [1]:

```
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline

# Optional for interactive
# %matplotlib notebook (watch video for full details)
```

In [2]:

```
mcdon = pd.read_csv('mcdonalds.csv',index_col='Date',parse_dates=True)
```

In [3]:

```
mcdon.head()
```

Out[3]:

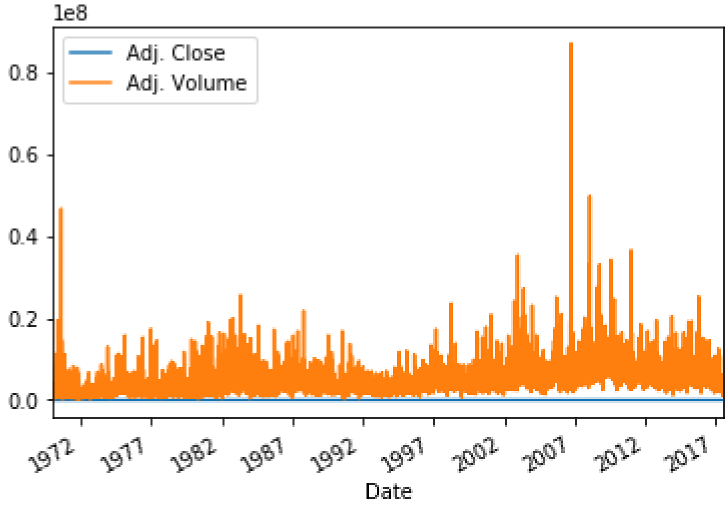
	Adj. Close	Adj. Volume
Date		
1970-01-02	0.209761	2825604.0
1970-01-05	0.213316	2210449.5
1970-01-06	0.214501	1951168.5
1970-01-07	0.213316	2728768.5
1970-01-08	0.213316	2242404.0

In [4]:

```
# Not Good!
mcdon.plot()
```

Out[4]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1955bcbea58>
```

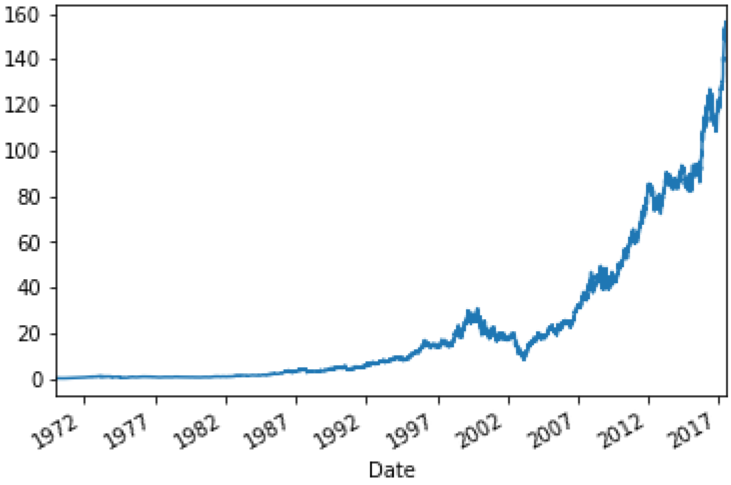


In [5]:

```
mcdon['Adj. Close'].plot()
```

Out[5]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1955be47e48>
```

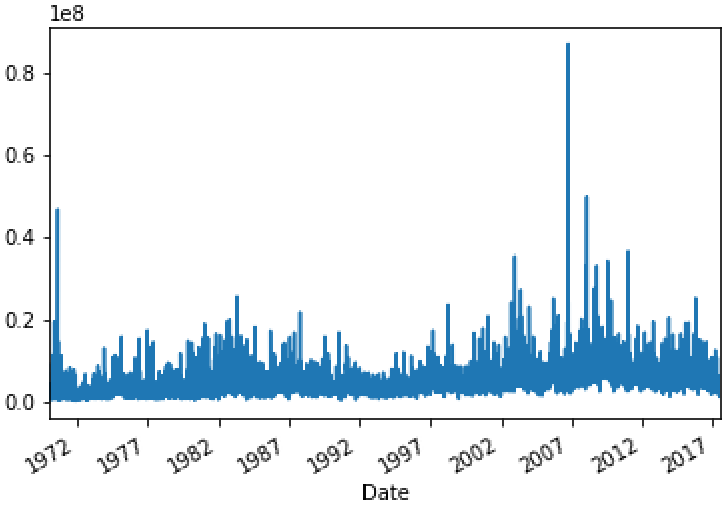


In [6]:

```
mcdon['Adj. Volume'].plot()
```

Out[6]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1955c3eb908>
```

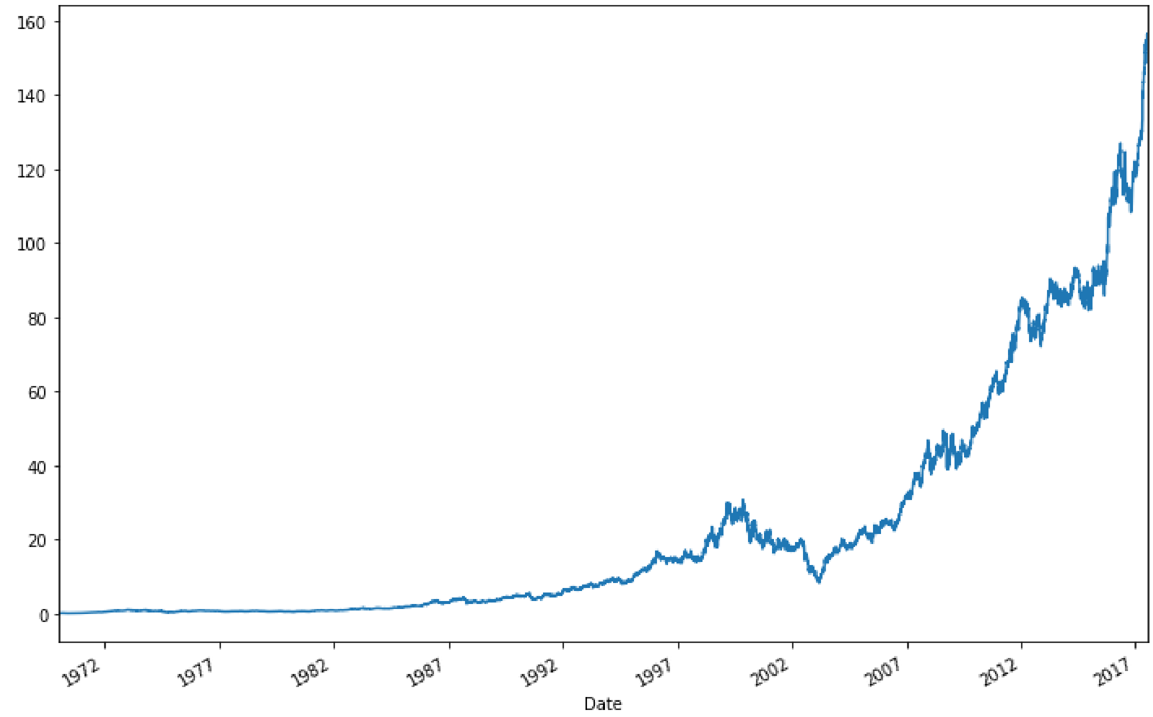


In [7]:

```
mcdon['Adj. Close'].plot(figsize=(12,8))
```

Out[7]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1955c3eb860>
```

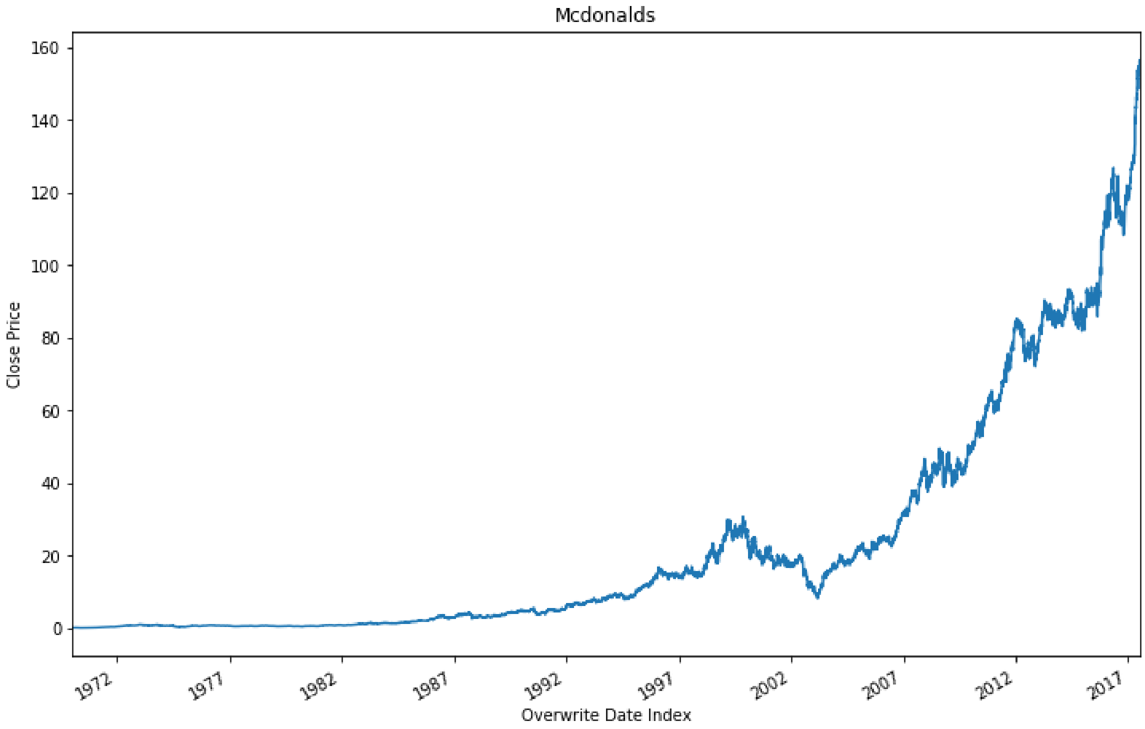


In [8]:

```
mcdon['Adj. Close'].plot(figsize=(12,8))
plt.ylabel('Close Price')
plt.xlabel('Overwrite Date Index')
plt.title('Mcdonalds')
```

Out[8]:

```
<matplotlib.text.Text at 0x1955c59b128>
```

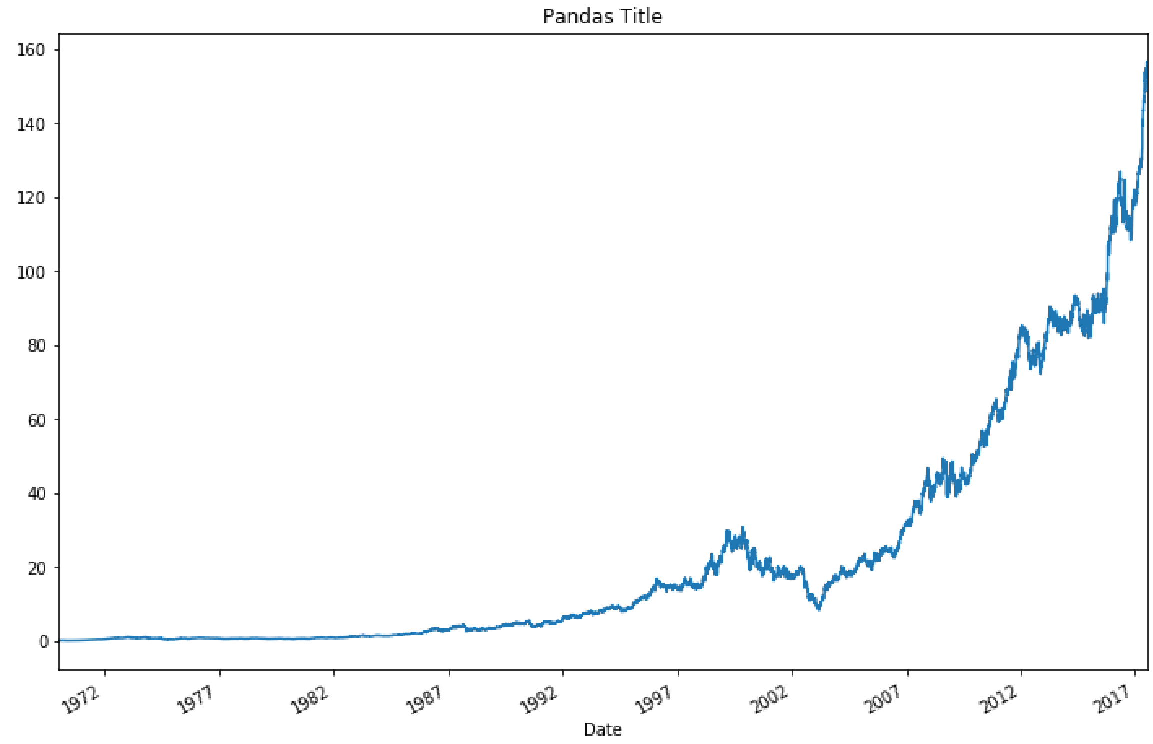


In [9]:

```
mcdon['Adj. Close'].plot(figsize=(12,8),title='Pandas Title')
```

Out[9]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1955c2eaef0>
```



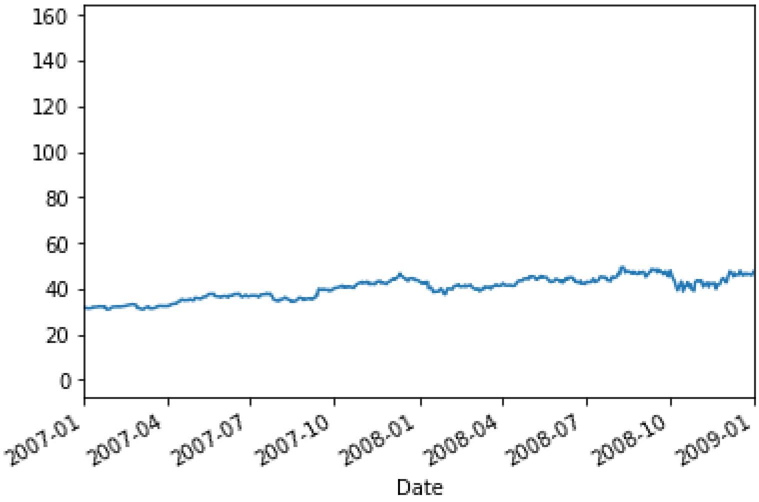
Plot Formatting

X Limits

```
In [10]:
```

```python
mcdon['Adj. Close'].plot(xlim=['2007-01-01','2009-01-01'])
```

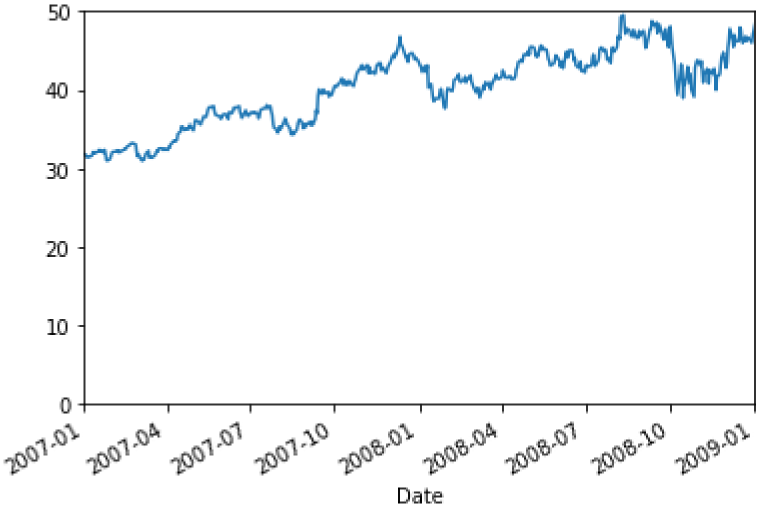
```
Out[10]:

<matplotlib.axes._subplots.AxesSubplot at 0x1955d7bb3c8>
```



```
In [11]:
```

```python
mcdon['Adj. Close'].plot(xlim=['2007-01-01','2009-01-01'],ylim=[0,50])
```

```
Out[11]:

<matplotlib.axes._subplots.AxesSubplot at 0x1955dbe0e80>
```



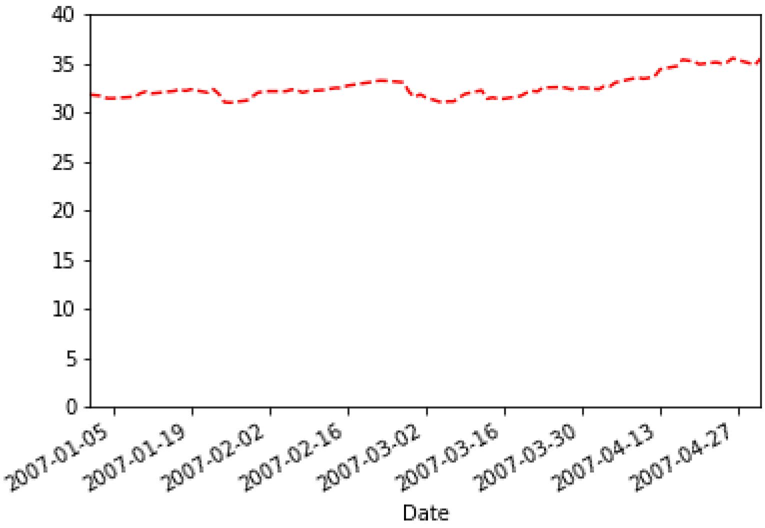
Color and Style

In [12]:

```
mcdon['Adj. Close'].plot(xlim=['2007-01-01','2007-05-01'],ylim=[0,40],ls='--',c=
'r')
```

Out[12]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1955dbeb1d0>
```



X Ticks

This is where you will need the power of matplotlib to do heavy lifting if you want some serious customization!

In [13]:

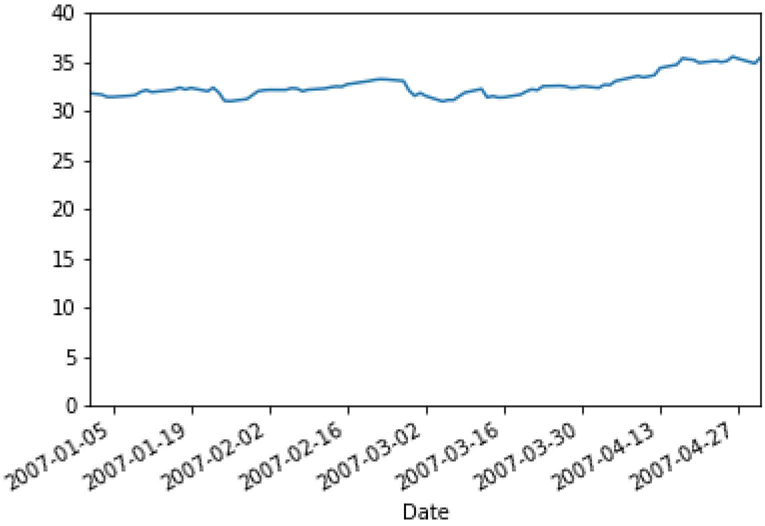
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.dates as dates
```

In [14]:

```
mcdon['Adj. Close'].plot(xlim=['2007-01-01','2007-05-01'],ylim=[0,40])
```

Out[14]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1955df6f780>
```



In [15]:

```
idx = mcdon.loc['2007-01-01':'2007-05-01'].index
stock = mcdon.loc['2007-01-01':'2007-05-01']['Adj. Close']
```

In [16]:

```
idx
```

Out[16]:

```
DatetimeIndex(['2007-01-03', '2007-01-04', '2007-01-05', '2007-01-0
8',
               '2007-01-09', '2007-01-10', '2007-01-11', '2007-01-1
2',
               '2007-01-16', '2007-01-17', '2007-01-18', '2007-01-1
9',
               '2007-01-22', '2007-01-23', '2007-01-24', '2007-01-2
5',
               '2007-01-26', '2007-01-29', '2007-01-30', '2007-01-3
1',
               '2007-02-01', '2007-02-02', '2007-02-05', '2007-02-0
6',
               '2007-02-07', '2007-02-08', '2007-02-09', '2007-02-1
2',
               '2007-02-13', '2007-02-14', '2007-02-15', '2007-02-1
6',
               '2007-02-20', '2007-02-21', '2007-02-22', '2007-02-2
3',
               '2007-02-26', '2007-02-27', '2007-02-28', '2007-03-0
1',
               '2007-03-02', '2007-03-05', '2007-03-06', '2007-03-0
7',
               '2007-03-08', '2007-03-09', '2007-03-12', '2007-03-1
3',
               '2007-03-14', '2007-03-15', '2007-03-16', '2007-03-1
9',
               '2007-03-20', '2007-03-21', '2007-03-22', '2007-03-2
3',
               '2007-03-26', '2007-03-27', '2007-03-28', '2007-03-2
9',
               '2007-03-30', '2007-04-02', '2007-04-03', '2007-04-0
4',
               '2007-04-05', '2007-04-09', '2007-04-10', '2007-04-1
1',
               '2007-04-12', '2007-04-13', '2007-04-16', '2007-04-1
7',
               '2007-04-18', '2007-04-19', '2007-04-20', '2007-04-2
3',
               '2007-04-24', '2007-04-25', '2007-04-26', '2007-04-2
7',
               '2007-04-30', '2007-05-01'],
              dtype='datetime64[ns]', name='Date', freq=None)
```

In [17]:

```
stock
```

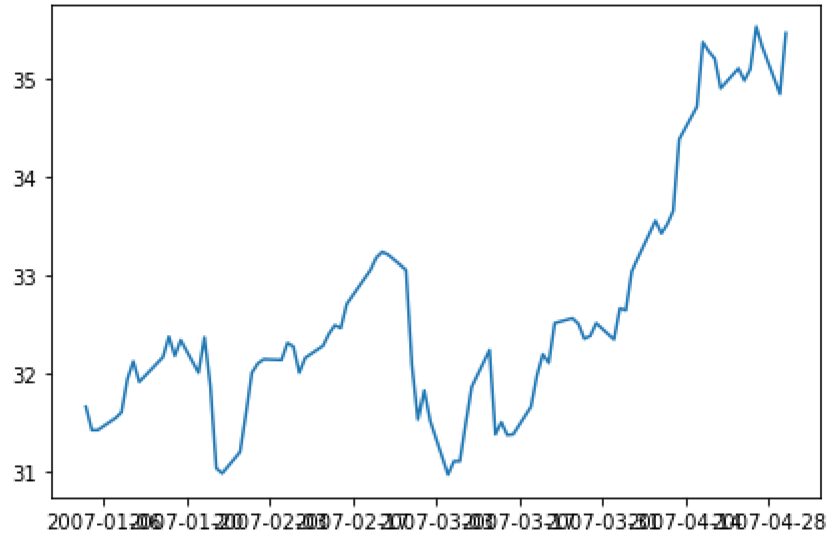
```
Out[17]:

Date
2007-01-03    31.662754
2007-01-04    31.424580
2007-01-05    31.424580
2007-01-08    31.547276
2007-01-09    31.605015
2007-01-10    31.944233
2007-01-11    32.124668
2007-01-12    31.915364
2007-01-16    32.167973
2007-01-17    32.377277
2007-01-18    32.182407
2007-01-19    32.341190
2007-01-22    32.009190
2007-01-23    32.370060
2007-01-24    31.872059
2007-01-25    31.034840
2007-01-26    30.984318
2007-01-29    31.200840
2007-01-30    31.590580
2007-01-31    32.009190
2007-02-01    32.103016
2007-02-02    32.146320
2007-02-05    32.139103
2007-02-06    32.312321
2007-02-07    32.276234
2007-02-08    32.009190
2007-02-09    32.160755
2007-02-12    32.283451
2007-02-13    32.406147
2007-02-14    32.492756
                ...
2007-03-20    31.980320
2007-03-21    32.196842
2007-03-22    32.110233
2007-03-23    32.514408
2007-03-26    32.564930
2007-03-27    32.507191
2007-03-28    32.355625
2007-03-29    32.384495
2007-03-30    32.514408
2007-04-02    32.348408
2007-04-03    32.665974
2007-04-04    32.644321
2007-04-05    33.041279
2007-04-09    33.553714
2007-04-10    33.423801
2007-04-11    33.517627
2007-04-12    33.654758
2007-04-13    34.383716
2007-04-16    34.715717
2007-04-17    35.372500
2007-04-18    35.278674
2007-04-19    35.206500
2007-04-20    34.903369
2007-04-23    35.105456
2007-04-24    34.982761
2007-04-25    35.098239
2007-04-26    35.531283
```

```
2007-04-27    35.329196
2007-04-30    34.845630
2007-05-01    35.466327
Name: Adj. Close, Length: 82, dtype: float64
```

Basic matplotlib plot

In [18]:

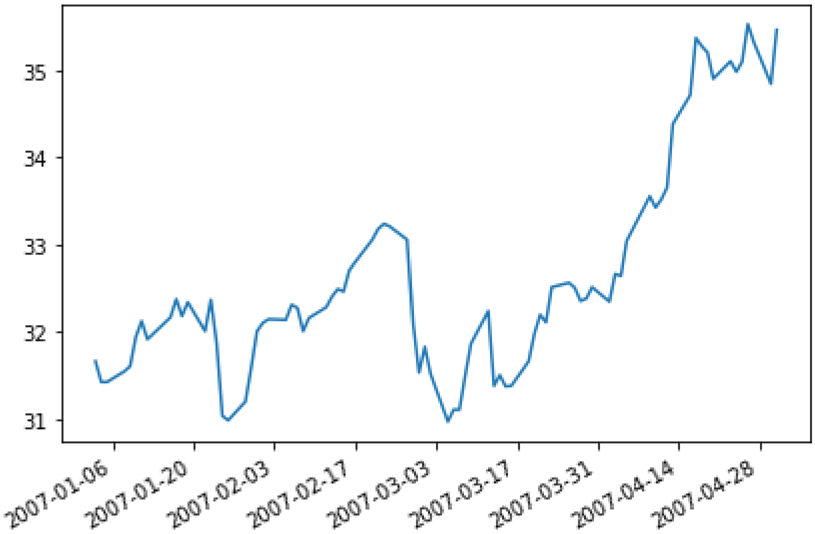
```
fig, ax = plt.subplots()
ax.plot_date(idx, stock,'-')
plt.tight_layout()
plt.show()
```



Fix the overlap!

In [19]:

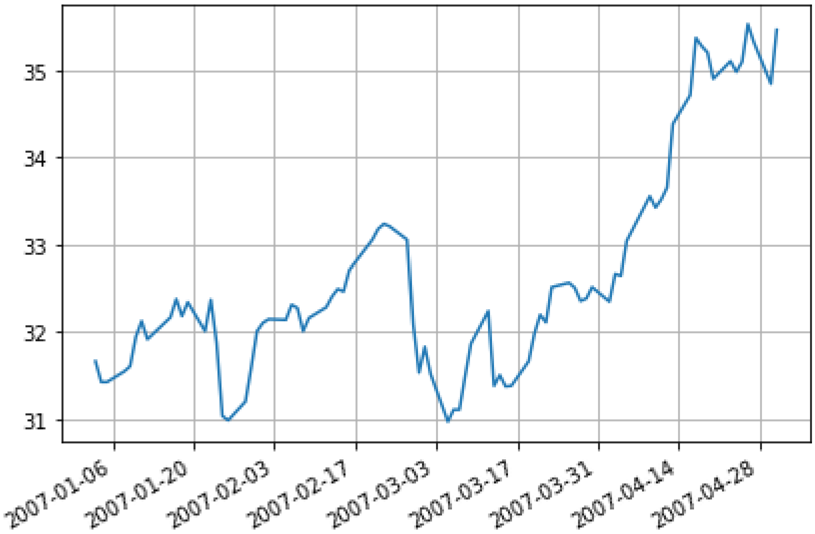
```
fig, ax = plt.subplots()
ax.plot_date(idx, stock,'-')

fig.autofmt_xdate() # Auto fixes the overlap!
plt.tight_layout()
plt.show()
```



Customize grid

In [20]:

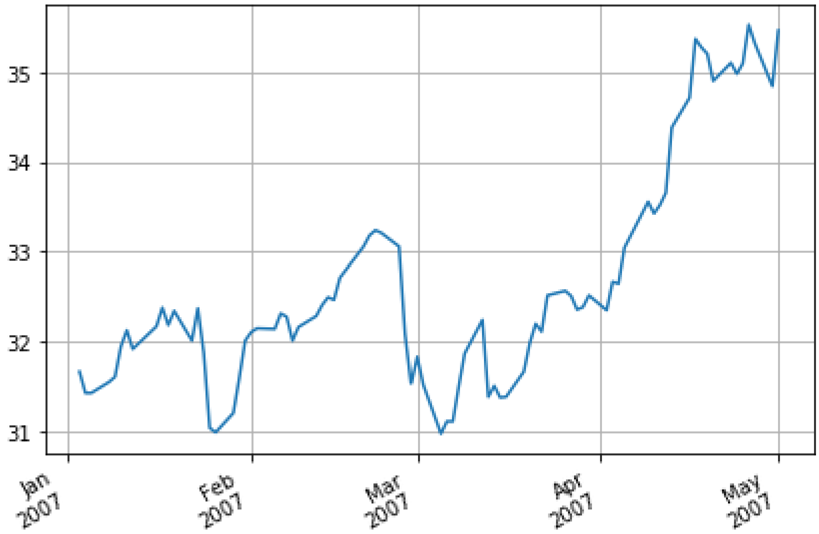
```
fig, ax = plt.subplots()
ax.plot_date(idx, stock,'-')
ax.yaxis.grid(True)
ax.xaxis.grid(True)
fig.autofmt_xdate() # Auto fixes the overlap!
plt.tight_layout()
plt.show()
```



Format dates on Major Axis

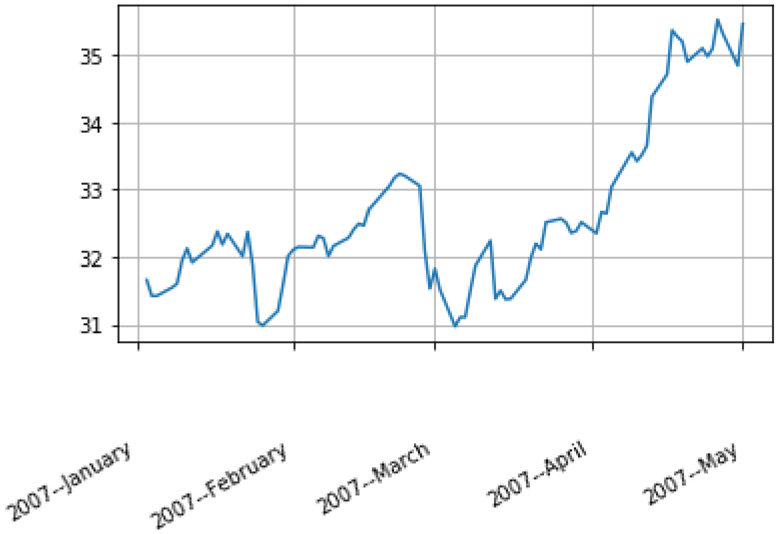
In [21]:

```
fig, ax = plt.subplots()
ax.plot_date(idx, stock,'-')

# Grids
ax.yaxis.grid(True)
ax.xaxis.grid(True)

# Major Axis
ax.xaxis.set_major_locator(dates.MonthLocator())
ax.xaxis.set_major_formatter(dates.DateFormatter('%b\n%Y'))

fig.autofmt_xdate() # Auto fixes the overlap!
plt.tight_layout()
plt.show()
```



In [22]:

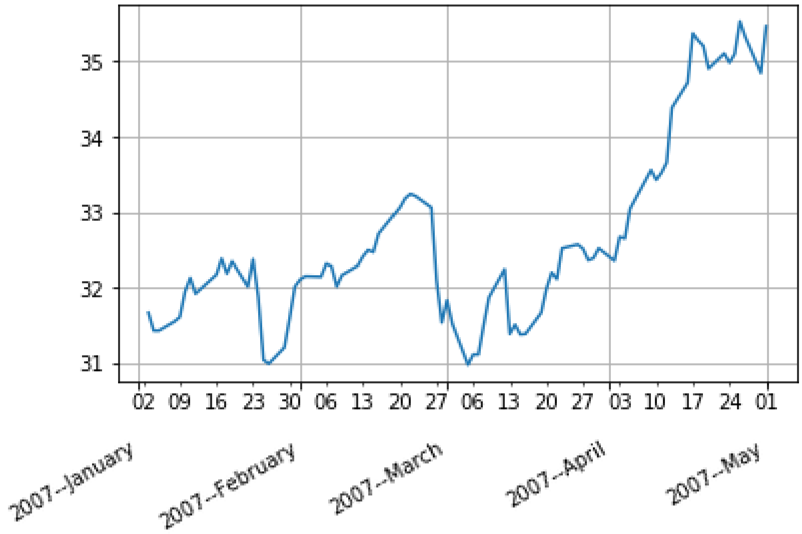
```
fig, ax = plt.subplots()
ax.plot_date(idx, stock,'-')

# Grids
ax.yaxis.grid(True)
ax.xaxis.grid(True)

# Major Axis
ax.xaxis.set_major_locator(dates.MonthLocator())
ax.xaxis.set_major_formatter(dates.DateFormatter('\n\n\n\n%Y--%B'))

fig.autofmt_xdate() # Auto fixes the overlap!
plt.tight_layout()
plt.show()
```



Minor Axis

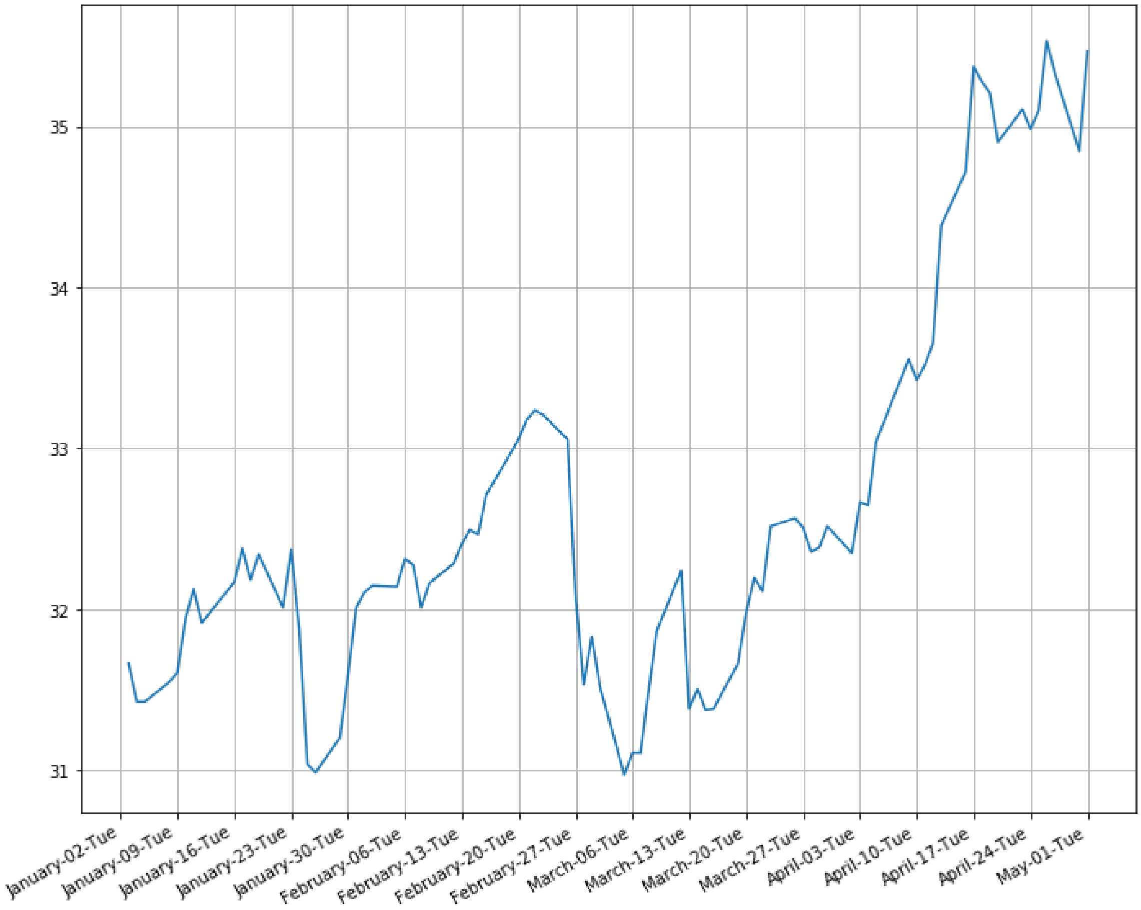
In [23]:

```
fig, ax = plt.subplots()
ax.plot_date(idx, stock,'-')


# Major Axis
ax.xaxis.set_major_locator(dates.MonthLocator())
ax.xaxis.set_major_formatter(dates.DateFormatter('\n\n%Y--%B'))

# Minor Axis
ax.xaxis.set_minor_locator(dates.WeekdayLocator())
ax.xaxis.set_minor_formatter(dates.DateFormatter('%d'))

# Grids
ax.yaxis.grid(True)
ax.xaxis.grid(True)

fig.autofmt_xdate() # Auto fixes the overlap!
plt.tight_layout()
plt.show()
```



In [33]:

```
fig, ax = plt.subplots(figsize=(10,8))
ax.plot_date(idx, stock,'-')


# Major Axis
ax.xaxis.set_major_locator(dates.WeekdayLocator(byweekday=1))
ax.xaxis.set_major_formatter(dates.DateFormatter('%B-%d-%a'))
# Grids
ax.yaxis.grid(True)
ax.xaxis.grid(True)

fig.autofmt_xdate() # Auto fixes the overlap!

plt.tight_layout()
plt.show()
```



Great job!