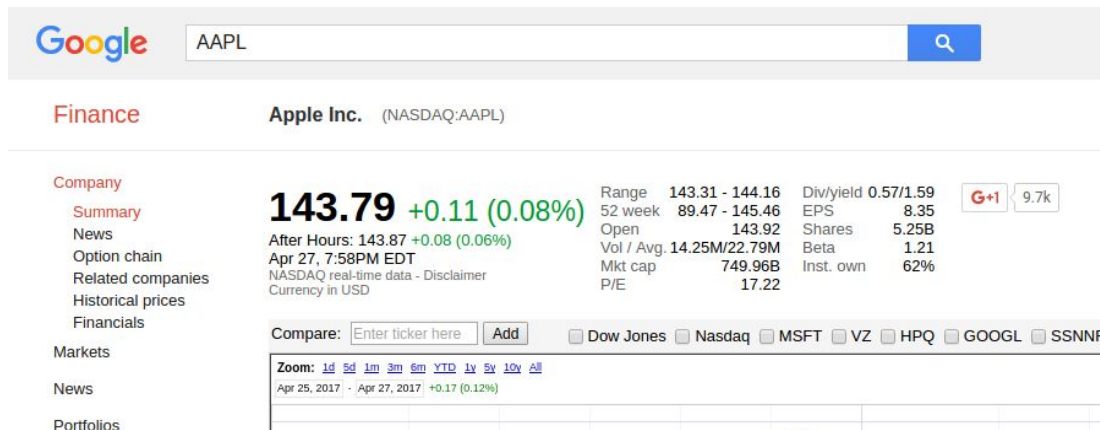# Exercise 1: Scrape the Stock Price for Apple

Below is a screenshot for the Apple Stock page on Google Finance

Below is a screenshot of part of the HTML source for the Apple stock page

----

In the space below, write the code that would extract ONLY the stock price for Apple. Some starter code is provided. Your code should indicate which tag and id contains the Apple stock price, and save that information to a variable called content. Use the screenshot of the HTML source to let you know what tag information should be specified to extract the price.

```
url = "https://www.google.com/finance?q=AAPL"
sourcecode = urllib.urlopen(url)
soup = BeautifulSoup(sourcecode, 'html.parser')
```

Exercise 2: Scrape the Headlines from Google News

In the space below, write the code that would extract ALL of the headline titles from the Google News homepage.
Save the list of headlines as a variable called contents, and then for each headline in contents, print them.

```
url = "https://news.google.com/"
sourcecode = urllib.urlopen(url)
soup = BeautifulSoup(sourcecode, 'html.parser'
```

Scrape the Links from the White House Press Briefing

BRIEFING ROOM   ISSUES   THE ADMINISTRATION   PARTICIPATE   1600 PENN

HOME · BRIEFING ROOM

From the Press Office

Speeches & Remarks
Press Briefings
Statements & Releases
Nominations & Appointments
Presidential Actions
Legislation
Disclosures

Press Briefings

April 27, 2017
Daily Press Briefing by Press Secretary Sean Spicer -- #42

April 26, 2017
Press Briefing by Secretary of Veterans Affairs David Shulkin on Executive Order Improving Accountability and Whistleblower Protections

FOLLOW US:
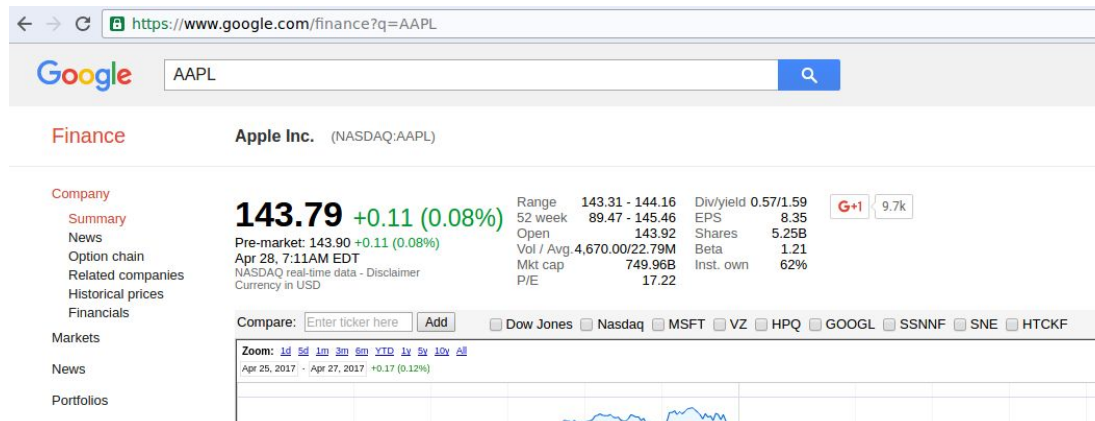TWITTER
FACEBOOK
INSTAGRAM
YOUTUBE
EMAIL

```
▶<p id="skip-link">…</p>
  <!-- !$page_manager_control: Only show these things if the page isn't a node under the control of page manager. -->
  <!-- /!$page_manager_control -->
  <!-- Main page content -->
▼<section id="page">
  <div id="nav-overlay"></div>
  ▼<div class="panelizer-view-mode node node-full node-forall-basic-page node-6776">
    ▼<div class="panel-display panel-basic-single-content-col">
      ▶<header class="panel-panel panel-header">…</header>
      ▼<div class="panel-panel panel-wrapper panel-col-section-all">
        ▶<div class="panel-panel panel-col panel-col-section-menu">…</div>
        ▼<div id="content-start" class="panel-panel panel-col panel-col-section-content" tabindex="-1">
          ▶<div class="panel-pane pane-node-title">…</div>
          ▼<div class="panel-pane pane-views-panes pane-press-office-listings-panel-pane-1">
            ▼<div class="view view-press-office-listings view-id-press_office_listings view-display-id-panel_pane_1 view-dom-id-
            e966ee0fb177efa117c9e858dce6064b">
              ▼<div class="view-content">
                ▼<div class="views-row views-row-1 views-row-odd views-row-first">
                  ▶<div class="views-field views-field-created">…</div>
                  ▼<div class="views-field views-field-title">
                    ▼<h3 class="field-content">
                        <a href="/the-press-office/2017/04/27/daily-press-briefing-press-secretary-sean-spicer-42">Daily Press Briefing by Press
                        Secretary Sean Spicer -- #42</a>
                      </h3>
                    </div>
                  </div>
                ▼<div class="views-row views-row-2 views-row-even">
                  ▶<div class="views-field views-field-created">…</div>
                  ▼<div class="views-field views-field-title">
                    ▼<h3 class="field-content">
                      ▶<a href="/the-press-office/2017/04/26/press-briefing-secretary-veterans-affairs-david-shulkin-executive-order">…</a>
                      </h3>
                    </div>
                  </div>
                ▼<div class="views-row views-row-3 views-row-odd">
                  ▶<div class="views-field views-field-created">…</div>
                  ▼<div class="views-field views-field-title">
                    ▼<h3 class="field-content">
                        <a href="/the-press-office/2017/04/26/background-briefing-first-100-days">Background Briefing on the First 100 Days</a>
                      </h3>
                    </div>
                  </div>
                ▶<div class="views-row views-row-4 views-row-even">…</div>
                ▶<div class="views-row views-row-5 views-row-odd">…</div>
```

In the space below, write the code that would extract ALL of the links to the press briefings from the White House website. Save the list of links as a variable called contents, and then for each link in contents, print them.

```
url = "https://www.whitehouse.gov/briefing-room/press-briefings"
sourcecode = urllib.urlopen(url)
soup = BeautifulSoup(sourcecode, 'html.parser')



```

Exercise 4: Scrape the Stock Price for Apple, Google, and Microsoft

----

In the space below, write code that loops through a list containing the stock symbols for Google ("GOOG"), Apple ("AAPL"), and Microsoft ("MSFT"). It should visit the webpage for each of the stocks, and then extract the stock price for each one, and append each price to a list called "data". Hint: Look at the url for a stock price in the example above

```
stocks = ["GOOG", "AAPL", "MSFT"]
data = []
```