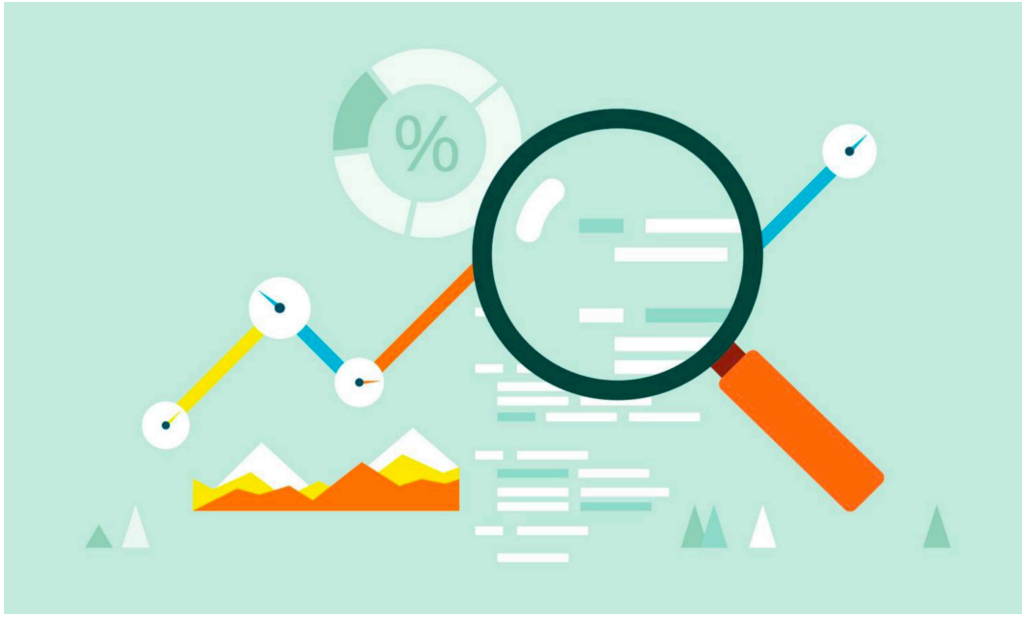
Data Science



Selenium

Selenium

- Selenium is basically a Web Browser Automation Tool, which simulates a user surfing the Internet.
- It allows -
  - Clicking buttons
  - Entering information in forms
  - Searching for specific information on the web pages

BeautifulSoup Limitations

- This process is suitable for static content
- Sometimes the data we want to extract is hidden behind JavaScript objects, objects that need to be clicked on to reveal the hidden data.

Installation

- Selenium package
- Webdriver
- Supported browsers are -
  - Chrome
  - Firefox
  - Internet Explorer
  - Safari
  - Opera
  - PhantomJS (invisible)

Browser Interaction

Methods & Properties

- driver.get(url)
- driver.back()
- driver.forward()
- driver.title
- driver.page_source

Methods & Properties

- maximize_window()
- driver.current_url
- driver.refresh()
- driver.get(driver.current_url)
- driver.close()
- driver.quit()

Locate Element

Locate element

- find_element_by_link_text()
- find_element_by_partial_link_text()
- find_element_by_id()
- find_element_by_class_name()
- find_element_by_name()
- find_element_by_tag_name()
- find_element_by_xpath()
- find_element_by_css_selector()

Locate element

- find_elements_by_link_text()
- find_elements_by_partial_link_text()
- find_elements_by_id()
- find_elements_by_class_name()
- find_elements_by_name()
- find_elements_by_tag_name()
- find_elements_by_xpath()
- find_elements_by_css_selector()

Methods & Properties

- click()
- clear()
- get_attribute(name)
- is_displayed()
- is_enabled()
- is_selected()
- send_keys(*value)
- submit()
- text
- tag_name

Locate Element

Locate element

- find_element_by_link_text()
- find_element_by_partial_link_text()
- find_element_by_id()
- find_element_by_class_name()
- find_element_by_name()
- find_element_by_tag_name()
- find_element_by_xpath()
- find_element_by_css_selector()

Using XPath

- Path
  - Absolute
  - Relative

Selecting Nodes

- nodename - Selects all nodes with the name "nodename"
- / - Selects from the root node
- // - Selects nodes in the document from the current node that match the selection no matter where they are
- @ - Selects attributes

Selecting Nodes

- Syntax -
  - //tag[@attribute = ‘value’]
- Examples -
  - //div[@id = ‘navbar]
  - //div[@id = ‘navbar]/div/li
  - //div[@class = ‘index’]//div
  - //*[@id = ‘navbar]
  - //*[@id = ‘navbar]/div/div/a
  - /bookstore/*
  - //*
  - //title[@*]

Partial match

- Syntax -
  - //tag[contains(@attribute, ‘value’)]
    - Helpful when there are multiple classes
  - //tag[starts-with(@attribute, ‘value’)]

Predicates

- /div/book[1]
- /div/book[last()]
- /div/book[last()-1]
- /div/book[position()<3]

Locate Element using CSS Selector

CSS Selector

- Absolute Path
- Relative Path
- Using class
- Using id