# Import package

from urllib.request import urlretrieve

# Import pandas

import pandas as pd

# Assign url of file: url

url = 'https://s3.amazonaws.com/assets.datacamp.com/production/course\_1606/datasets/winequality-red.csv'

# Save file locally

urlretrieve(url, 'winequality-red.csv')

# Read file into a DataFrame and print its head

df = pd.read\_csv('winequality-red.csv', sep=';')

print(df.head())

==

# Import packages

import matplotlib.pyplot as plt

import pandas as pd

# Assign url of file: url

url = 'https://s3.amazonaws.com/assets.datacamp.com/production/course\_1606/datasets/winequality-red.csv'

# Read file into a DataFrame: df

df = pd.read\_csv(url, sep=';')

# Print the head of the DataFrame

print(df.head())

# Plot first column of df

pd.DataFrame.hist(df.ix[:, 0:1])

plt.xlabel('fixed acidity (g(tartaric acid)/dm$^3$)')

plt.ylabel('count')

plt.show()

==

# Import package

import pandas as pd

# Assign url of file: url

url = 'http://s3.amazonaws.com/assets.datacamp.com/course/importing\_data\_into\_r/latitude.xls'

# Read in all sheets of Excel file: xl

xl = pd.read\_excel(url, sheetname=None)

# Print the sheetnames to the shell

print(xl.keys())

# Print the head of the first sheet (using its name, NOT its index)

print(xl['1700'].head())

==

# Import packages

from urllib.request import urlopen, Request

# Specify the url

url = "http://www.datacamp.com/teach/documentation"

# This packages the request: request

request = Request(url)

# Sends the request and catches the response: response

response = urlopen(request)

# Print the datatype of response

print(type(response))

# Be polite and close the response!

response.close()

==

# Import packages

from urllib.request import urlopen, Request

# Specify the url

url = "http://www.datacamp.com/teach/documentation"

# This packages the request

request = Request(url)

# Sends the request and catches the response: response

response = urlopen(request)

# Extract the response: html

html = response.read()

# Print the html

print(html)

# Be polite and close the response!

response.close()

==

# Import package

import requests

# Specify the url: url

url = "http://www.datacamp.com/teach/documentation"

# Packages the request, send the request and catch the response: r

r = requests.get(url)

# Extract the response: text

text = r.text

# Print the html

print(text)

==

# Import packages

import requests

from bs4 import BeautifulSoup

# Specify url: url

url = 'https://www.python.org/~guido/'

# Package the request, send the request and catch the response: r

r = requests.get(url)

# Extracts the response as html: html\_doc

html\_doc = r.text

# Create a BeautifulSoup object from the HTML: soup

soup = BeautifulSoup(html\_doc)

# Prettify the BeautifulSoup object: pretty\_soup

pretty\_soup = soup.prettify()

# Print the response

print(pretty\_soup)

==

# Import packages

import requests

from bs4 import BeautifulSoup

# Specify url: url

url = 'https://www.python.org/~guido/'

# Package the request, send the request and catch the response: r

r = requests.get(url)

# Extract the response as html: html\_doc

html\_doc = r.text

# Create a BeautifulSoup object from the HTML: soup

soup = BeautifulSoup(html\_doc)

# Get the title of Guido's webpage: guido\_title

guido\_title = soup.title

# Print the title of Guido's webpage to the shell

print(guido\_title)

# Get Guido's text: guido\_text

guido\_text = soup.get\_text()

# Print Guido's text to the shell

print(guido\_text)

==

# Import packages

import requests

from bs4 import BeautifulSoup

# Specify url

url = 'https://www.python.org/~guido/'

# Package the request, send the request and catch the response: r

r = requests.get(url)

# Extracts the response as html: html\_doc

html\_doc = r.text

# create a BeautifulSoup object from the HTML: soup

soup = BeautifulSoup(html\_doc)

# Print the title of Guido's webpage

print(soup.title)

# Find all 'a' tags (which define hyperlinks): a\_tags

a\_tags = soup.find\_all('a')

# Print the URLs to the shell

for link in a\_tags:

print(link.get('href'))

==