



SEOUL NATIONAL UNIVERSITY
Graduate School of Data Science

M3239.003100: Data Analysis and Visualization

Lecture 2

Data Science Pipeline and Data Collection

Hyunwoo Park

Graduate School of Data Science

Seoul National University

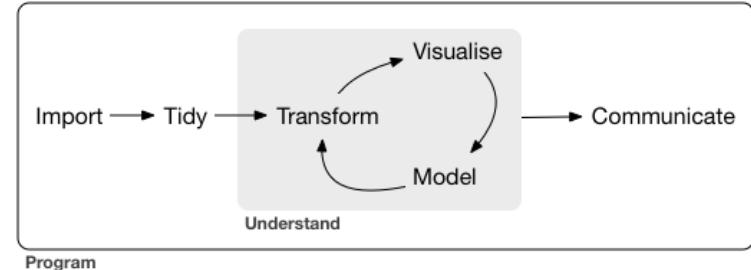
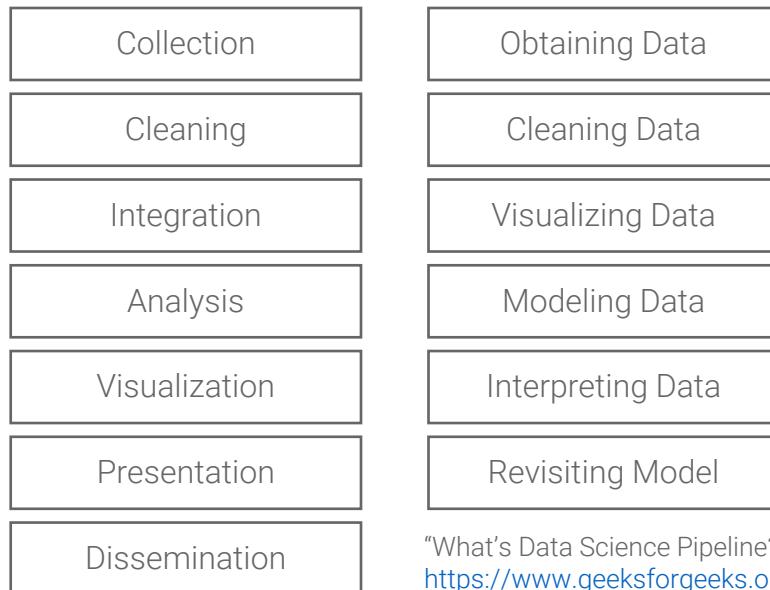
Agenda

- Data Science Pipeline
- Data Collection
- Data Parsing
- Simple Exercise
- Class Survey (Due 9/13 before class)
 - Submitted via Google Forms
- Things to do
 - Install Stata
 - Install the Anaconda distribution of Python 3
 - Launch Jupyter Notebook
 - Get to know the terminal interface
 - Pick a text/code editor

Data Science Pipeline

Data science pipeline

- A typical process that a data science project goes through



What a typical data science project looks like
<https://r4ds.had.co.nz/introduction.html>

"What's Data Science Pipeline?"
<https://www.geeksforgeeks.org/whats-data-science-pipeline/>

Polo Chau's

Analytics Building Blocks

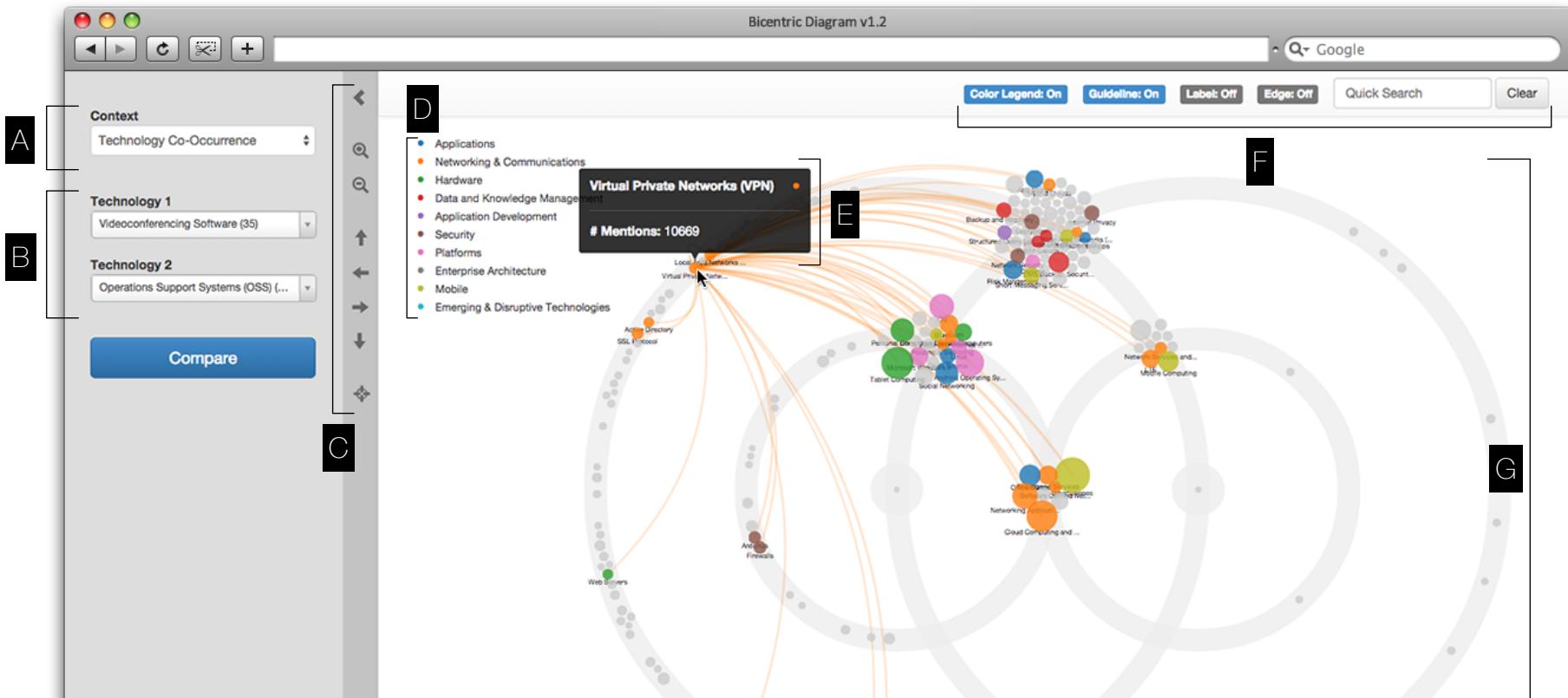
<https://poloclub.github.io/cse6242-2018fall-campus/slides/CSE6242-010-AnalyticsBuildingBlocks.pdf>

Common theme

- You may skip some steps.
- You can go back and forth. (Blocks are linked by two-way street.)
- For example,
 - Data types inform visualization design.
 - Data size informs the choice of algorithms.
 - Analysis motivates more data collection/cleaning/integration.
 - Visualization motivates more data cleaning.
 - Visualization challenges algorithm assumptions. (e.g., users may find results do not make sense.)

An example: Bicentric Diagrams

- <https://youtu.be/TFBf3Nm5XQ8>



Data Collection / Cleaning / Integrating

The screenshot shows the ACM Digital Library interface. At the top, there are navigation links for Journals, Magazines, Proceedings, Books, SIGs, Conferences, and People. On the right, there are links for Seoul National University, Browse, About, Sign in, and Register. A search bar at the top right says "Search ACM Digital Library". Below the header, a navigation bar has tabs for Institution's Profile, Award Winners, Authors, Collaborative Institutions (which is highlighted in grey), and Publication Archive.

In the center, a large blue banner features the text "Georgia Institute of Technology". To its right is a search bar with the placeholder "Search for all publications from Georgia Institute of Technology" and a magnifying glass icon.

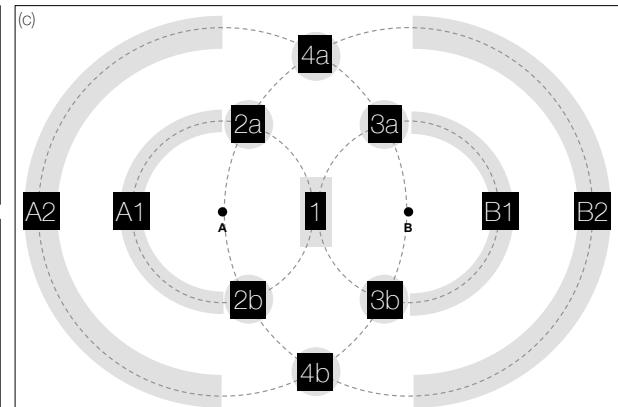
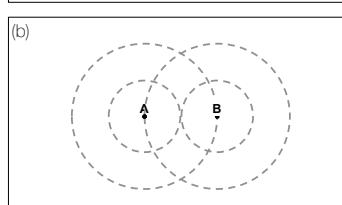
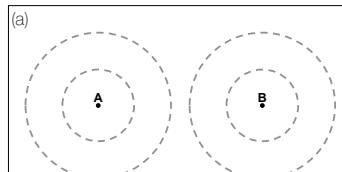
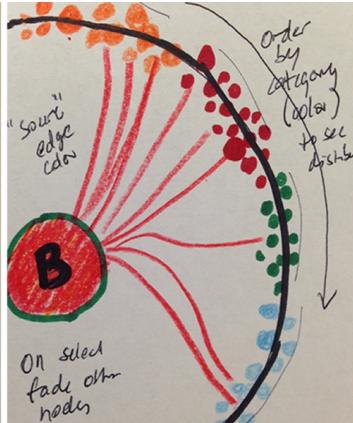
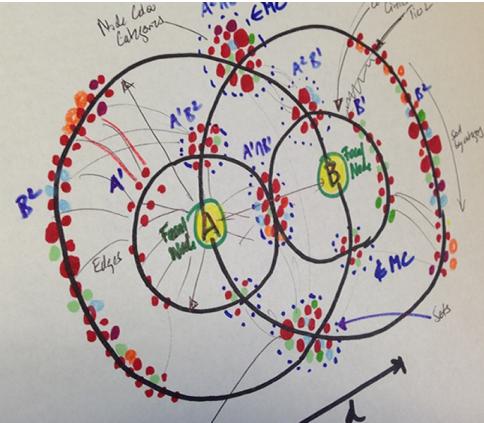
Home > Georgia Institute of Technology > Collaborative Institutions

The main content area displays the Georgia Institute of Technology logo and the text "Atlanta, GA, United States". On the left, there is a sidebar titled "Applied Filters" containing a single filter: "Georgia Institute Of Technology". Below it is another sidebar titled "Authors" listing three individuals: James R Wilson (30), Roberto Perdisci (20), and Scott Alan Klasky (20).

The main table lists collaborative institutions with their names, logos, paper counts, and a "View Details" link. The first two rows are shown below:

Name	Paper Counts
Carnegie Mellon University	241
Microsoft Research	189

Analysis / Visualization



_PRESENTATION / DISSEMINATION

- <http://bicentric.herokuapp.com/>

Decision Support Systems 84 (2016) 64–77



Bicentric diagrams: Design and applications of a graph-based relational set visualization technique



Hyunwoo Park ^{a,*}, Rahul C. Basole ^{b,1}

^a Tenenbaum Institute, Georgia Institute of Technology, 85 Fifth Street NW, Atlanta, GA 30308, USA

^b School of Interactive Computing & Tenenbaum Institute, Georgia Institute of Technology, 85 Fifth Street NW, Atlanta, GA 30308, USA

ARTICLE INFO

Article history:

Received 20 February 2015

Received in revised form 22 December 2015

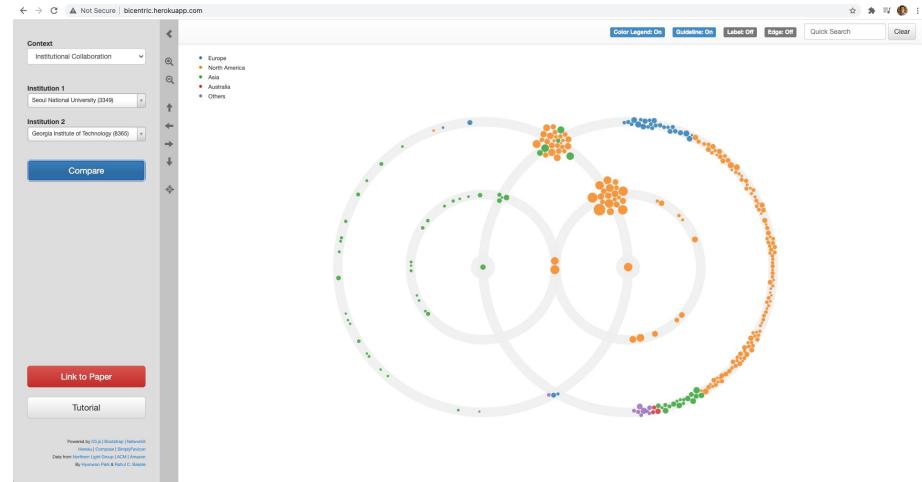
Accepted 2 February 2016

Available online 12 February 2016

ABSTRACT

In an era where data on social, economic, and physical networks are proliferating at a rapid pace, the ability to understand the underlying complex structural connections, discover prominent entities, and identify clusters is becoming increasingly important. It is also well-established that interactive visualizations can amplify human cognition and augment decision making. Motivated by a practical need articulated by corporate decision makers and limitations of existing visual representations, this research presents our journey in designing and implementing bicentric diagrams, a novel graph-based set visualization technique. A bicentric diagram enables simultaneous identification of sets, set relationships, and set member reach in integrated egonetworks of two focal entities. Our technique builds on the well-established sociological theory of tie strength to visually group and position nodes. We illustrate the broad applicability of bicentric diagrams with examples from four diverse sample domains: university collaboration, technology co-occurrence, health app purchases, and interfirm alliance networks. We assess the value of our technique using an expert-based evaluation approach. The paper concludes with implications and a discussion of opportunities for implementation in real-world decision support settings.

© 2016 Elsevier B.V. All rights reserved.



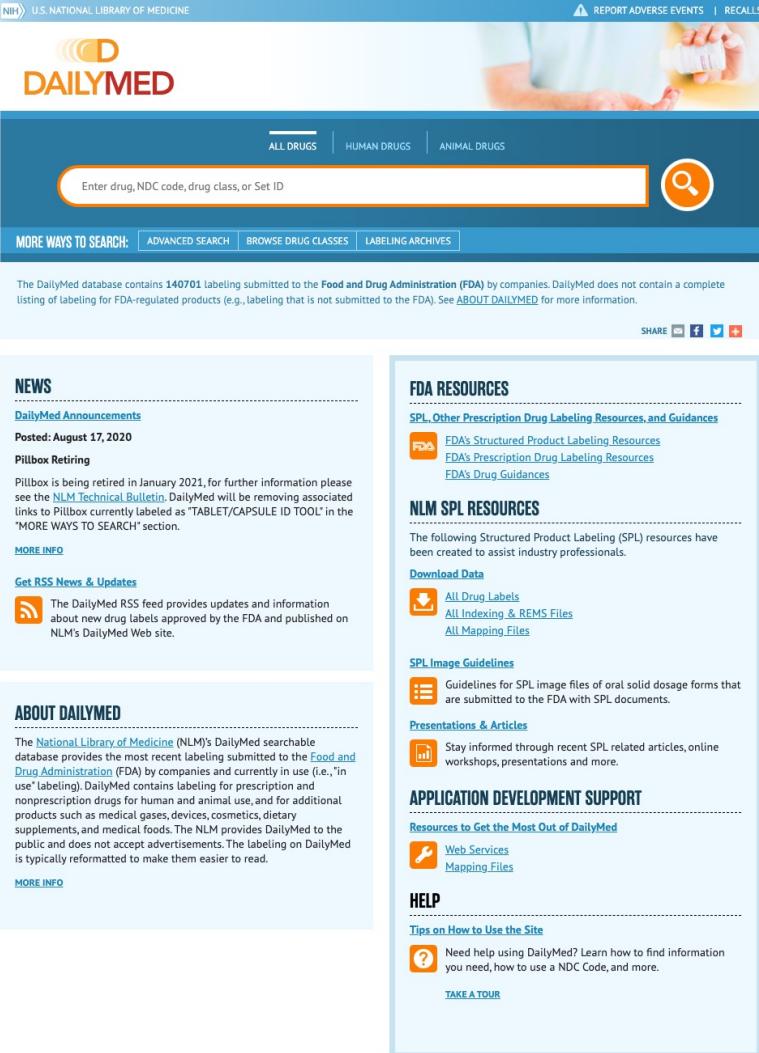
Data Collection

Three ways you can collect data

1. Bulk data download – Easy
2. Application Programming Interface (API) – Medium

3. Scraping or crawling – Hard

- Combining multiple sources of data is one way to develop a unique dataset.
- Let me use this website as an example.
<https://dailymed.nlm.nih.gov/>



The screenshot shows the homepage of the DailyMed website. At the top, there's a navigation bar with links for "ALL DRUGS", "HUMAN DRUGS", and "ANIMAL DRUGS". Below that is a search bar with the placeholder "Enter drug, NDC code, drug class, or Set ID" and a magnifying glass icon. Under the search bar are buttons for "MORE WAYS TO SEARCH", "ADVANCED SEARCH", "BROWSE DRUG CLASSES", and "LABELING ARCHIVES". A message below the search bar states: "The DailyMed database contains 140701 labeling submitted to the Food and Drug Administration (FDA) by companies. DailyMed does not contain a complete listing of labeling for FDA-regulated products (e.g., labeling that is not submitted to the FDA). See [ABOUT DAILYMED](#) for more information." To the right of this message are social media sharing icons for LinkedIn, Facebook, Twitter, and YouTube. The main content area has several sections: "NEWS" (with a "DailyMed Announcements" link), "FDA RESOURCES" (with links to "FDA's Structured Product Labeling Resources", "FDA's Prescription Drug Labeling Resources", and "FDA's Drug Guidances"), "SPL SPL RESOURCES" (with links to "All Drug Labels", "All Indexing & REMS Files", and "All Mapping Files"), "SPL Image Guidelines" (with a link to "Guidelines for SPL image files of oral solid dosage forms that are submitted to the FDA with SPL documents"), "PRESENTATIONS & ARTICLES" (with a link to "Stay informed through recent SPL related articles, online workshops, presentations and more."), "APPLICATION DEVELOPMENT SUPPORT" (with a link to "Resources to Get the Most Out of DailyMed"), "HELP" (with a link to "Tips on How to Use the Site"), and "TAKE A TOUR" (with a link to "Need help using DailyMed? Learn how to find information you need, how to use a NDC Code, and more."). The top right corner of the page features a small image of a hand holding a prescription bottle.

1) Bulk data download

- Data is money these days. Proprietary data is often not available for download.
- Bulk data download is often offered by non-profit organizations or governments.
- Because anyone can obtain such data, your analysis or visual analytics system for such data may not be seen novel and interesting by itself.
- This type of data may provide some supplementary information for your main dataset.
- Examples include:
 - StackOverflow dump, Wikipedia, Data.gov, etc.
 - <https://poloclub.github.io/cse6242-2019spring-campus/#datasets>
 - <http://va.gatech.edu/courses/cs4460/resources/> [“Data Sources” section]
 - Please feel free to post a data source in Korea on the eTL Q&A board.

- <https://dailymed.nlm.nih.gov/dailymed/spl-resources-all-drug-labels.cfm>

NLM SPL RESOURCES

The following Structured Product Labeling (SPL) resources have been created to assist industry professionals.

Download Data



- [All Drug Labels](#)
- [All Indexing & REMS Files](#)
- [All Mapping Files](#)

FULL RELEASES

Warning: The full human prescription and OTC archive files, dm_spl_release_human_rx.zip and dm_spl_release_human_otc.zip, are no longer available due to size considerations. Instead, these archives have been split into multiple parts. The remainder archive files consist of bulk ingredient labels, vaccine labels, and some labels for medical devices.

HUMAN PRESCRIPTION LABELS

[dm_spl_release_human_rx_part1.zip](#) [[HTTPS](#) / [FTP](#)]
Number of files: 14,658 | File size: 3.00GB | MD5 checksum: 721232ade9eee11bff4a0f2922878199 | Last Modified: Sep 3, 2021

[dm_spl_release_human_rx_part2.zip](#) [[HTTPS](#) / [FTP](#)]
Number of files: 10,924 | File size: 3.00GB | MD5 checksum: 50e0cf7cd2c991e09cc7ad6f050313 | Last Modified: Sep 3, 2021

[dm_spl_release_human_rx_part3.zip](#) [[HTTPS](#) / [FTP](#)]
Number of files: 10,867 | File size: 3.00GB | MD5 checksum: 4f6656da03c980ed46c5f416e47953a3 | Last Modified: Sep 3, 2021

[dm_spl_release_human_rx_part4.zip](#) [[HTTPS](#) / [FTP](#)]
Number of files: 7,636 | File size: 2.35GB | MD5 checksum: 745d803cd844b83e1ff143b88c2becc2 | Last Modified: Sep 3, 2021

2) API (Application Programming Interface)

- Some web sites expose API endpoints to be accessed.
- API endpoints typically return a structured response in the XML or JSON format.
- For developers, it is easier because data parsing and cleaning work is reduced.
- Instead, the web site retains a greater control and monitoring over what data to be shared and who accesses the data.
- API is often intended for providing real-time data for third-party applications.
- You may need to get an API key to authenticate your request.
- Rate limiting or IP monitoring is typically in place. (a certain number of requests per day or per hour)

 APPLICATION DEVELOPMENT SUPPORT: Web Services: RESTful Resources

PRINT  SHARE    

"/splis/{SETID}"
Returns the XML document for the specified SET ID.
API Version: 2

NOTE: This Web Service is not meant to be viewed using an internet browser. If you wish to view using an internet browser, consider installing an add-on or extension to your browser that will allow your browser to query RESTful APIs.

PATH PARAMETERS

- **SETID** - The SET ID of a specific SPL. This parameter is **required**.

RETURN FORMATS

- **XML**: <https://dailymed.nlm.nih.gov/dailymed/services/v2/splis/{SETID}.xml>

RETURN FORMATS EXAMPLES

XML: <https://dailymed.nlm.nih.gov/dailymed/services/v2/splis/1efe378e-feel-4ae9-8ea5-0fe2265fe2d8.xml>

Returns

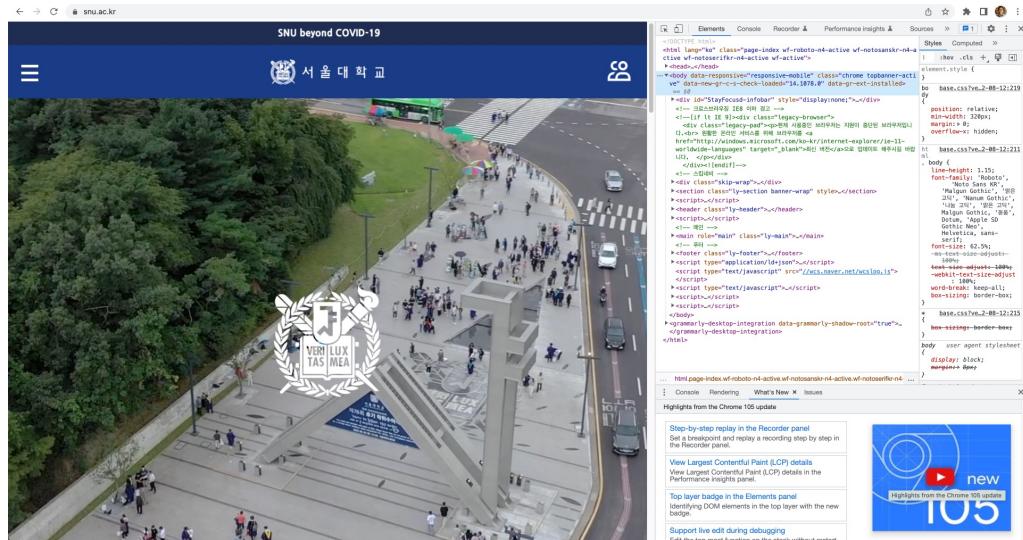
```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet href="https://www.accessdata.fda.gov/spl/stylesheets/spl.xsl" type="text/xsl"?>
<Document xmlns="urn:hl7-org:v3" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:hl7-org:v3 https://www.accessdata.fda.gov/spl/schema/spl.xsd">
  <id root="9187708f-cc7b-4fe9-828e-9a77e98354ad" />
  <code code="34391-3" codeSystem="2.16.840.1.113883.6.1" displayName="HUMAN
PRESCRIPTION DRUG LABEL" />
  <title>
```

...(Rest of SPL Document)...

[CLOSE](#)

3) Scraping or crawling

- The most brute-force and messy way to get data is to scrape a web site.
- You will get a bunch of HTML files and parse/clean them out to get structured data.
- Figuring out the structure of URLs and getting the list of IDs to scrape are the first step in scraping.



Popular Python packages for scraping

- Requests
 - <https://docs.python-requests.org/en/latest/user/quickstart/>
- Scrapy
 - <https://docs.scrapy.org/en/latest/>
- Selenium
 - Not part of Anaconda distribution
 - <https://selenium-python.readthedocs.io/getting-started.html#simple-usage>
- Note
 - You may have to log in with username and password to get the data you want depending on the website.
 - You may need to change the “user agent” of your requests to trick the server.
 - Be courteous to the server load. Space out your requests using something like “time.sleep(1).”
 - Otherwise, your IP may be blocked from the server.

Data Parsing

XML (or HTML) and JSON

- XML (or HTML)

```
1  <?xml version="1.0" encoding="UTF-8"?><?xml-stylesheet href="https://www.accessdata.fda.gov/spl/stylesheets/spl.xsl" type="text/xsl"?>
2  <document xmlns="urn:hl7-org:v3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:hl7-org:v3 https://www.accessdata.fda.gov/spl/schema/spl.xsd">
3    <id root="874057c0-a437-4a08-9105-38c1b2f3307b"/>
4    <code code="34391-3" displayName="HUMAN PRESCRIPTION DRUG LABEL" codeSystem="2.16.840.1.113883.6.1"/>
5    <title>
6      <content styleCode="bold">These highlights do not include all the information needed to use HUMIRA safely and effectively. See full prescribing information for HUMIRA.</content>
7      <br/>
8      <content styleCode="bold"><0xa0><0xa0><0xa0><0xa0><0xa0><0xa0></content>
9      <br/>
10     <content styleCode="bold">HUMIRA</content>
11     <content styleCode="bold">
12       <sup>®</sup>
13     </content>
14     <content styleCode="bold"> (adalimumab) injection, for subcutaneous use</content>
15     <br/>
16     <content styleCode="bold">Initial U.S. Approval: 2002</content>
17     <br/>
18   </title>
19   <effectiveTime value="20210224"/>
20   <setId root="608d4ff0d-b19f-46d3-749a-7159aa5f933d"/>
21   <versionNumber value="2135"/>
22   <author>
23     <time/>
24     <assignedEntity>
25       <representedOrganization>
26         <id extension="078458370" root="1.3.6.1.4.1.1519.1"/>
27         <name>AbbVie Inc.</name>
28         <assignedEntity>
29           <representedOrganization/>
30         </assignedEntity>
31       </representedOrganization>
32     </assignedEntity>
```

- JSON

```
1  {
2    "users": [
3      {
4        "name": "John",
5        "age": 25
6      },
7      {
8        "name": "Mark",
9        "age": 29
10     },
11     {
12       "name": "Sarah",
13       "age": 22
14     }
15   ],
16   "dataTitle": "JSON Tutorial!",
17   "swiftVersion": 2.1
18 }
```

- Basics

- <https://www.geeksforgeeks.org/xml-basics/>
- <https://www.geeksforgeeks.org/javascript-json/>
- <https://www.geeksforgeeks.org/difference-between-json-and-xml/>

Parsing XML (and HTML)

- BeautifulSoup (package name: bs4)
 - <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

Beautiful Soup 4.9.0 documentation » Beautiful Soup Documentation

Table of Contents

- Beautiful Soup Documentation
 - Getting help
 - Quick Start
 - Installing Beautiful Soup
 - Problems after installation
 - Installing a parser
 - Managing the soup
 - Kinds of objects
 - Tag
 - Name
 - Attributes
 - Multi-Child attributes
 - NavigableString
 - BeautifulSoup
 - Comments and other spannables
 - Navigating the tree
 - Going down
 - Navigating using tag names
 - .children and .descendants
 - .string
 - .strings and .prettify_strings
 - Going up
 - .parent
 - .parents
 - Going sideways
 - .next_sibling and .previous_sibling
 - .next_siblings and .previous_siblings
 - Going back and forth
 - .next_element and .previous_element
 - .next_elements and .previous_elements
 - Searching the tree
 - Kinds of filters
 - A string
 - A regular expression
 - A list
 - True
 - Action
 - filter
 - The same argument
 - The keyword arguments

Beautiful Soup Documentation

Beautiful Soup is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers hours or days of work.

These instructions illustrate all major features of Beautiful Soup 4, with examples. I show you what the library is good for, how it works, how to use it, how to make it do what you want, and what to do when it violates your expectations.

This document covers Beautiful Soup version 4.9.3. The examples in this documentation should work the same way in Python 2.7 and Python 3.8.

You might be looking for the documentation for Beautiful Soup 3. If so, you should know that Beautiful Soup 3 is no longer being developed and that support for it will be dropped on or after December 31, 2020. If you want to learn about the differences between Beautiful Soup 3 and Beautiful Soup 4, see Porting code to BS4.

This documentation has been translated into other languages by Beautiful Soup users:

- 这篇文档当然还有中文版。
- このページは日本語で利用できます(外部リンク)
- 이 문서는 한국어 번역도 가능합니다.
- Este documento também está disponível em Português do Brasil.
- Эта документация доступна на русском языке.

Getting help

If you have questions about Beautiful Soup, or run into problems, send mail to the discussion group. If your problem involves parsing an HTML document, be sure to mention what the `diagnose()` function says about that document.

Quick Start

Here's an HTML document I'll be using as an example throughout this document. It's part of a story from *Alice in Wonderland*.

```
html_doc = """<html><head><title>The Dormouse's story</title></head>
<body>
<p class="title"><b>The Dormouse's story</b></p>

<p class="story">Once upon a time there were three little sisters; and their names were
<a href="http://example.com/elsie" class="sister" id="link1">Elsie</a>,"
```

- ElementTree (built-in)
 - <https://docs.python.org/3.8/library/xml.etree.elementtree.html>

Python » English ▾ 3.8.11 ▾ Documentation » The Python Standard Library » Structured Markup Processing Tools »

Table of Contents

`xml.etree.ElementTree` — The ElementTree XML API

- Tutorial
 - XML tree and elements
 - Parsing XML
 - Pull API for non-blocking parsing
 - Finding interesting elements
 - Modifying an XML File
 - Building XML documents
 - Parsing XML with Namespaces
 - Additional resources
- XPath support
 - Example
 - Supported XPath syntax
- Reference
 - Functions
 - `xinclude` support
 - Example
 - Reference
 - Functions
 - Element Objects
 - ElementTree Objects
 - QName Objects
 - TreeBuilder Objects
 - XMLParser Objects
 - XMLPullParser Objects
 - Exceptions

Source code: [Lib/xml/etree/ElementTree.py](#)

The `xml.etree.ElementTree` module implements a simple and efficient API for parsing and creating XML data.

Changed in version 3.3: This module will use a fast implementation whenever available. The `xml.etree.ElementTree` module is deprecated.

Warning: The `xml.etree.ElementTree` module is not secure against maliciously constructed data. If you need to parse untrusted or unauthenticated data see [XML vulnerabilities](#).

Tutorial

This is a short tutorial for using `xml.etree.ElementTree` (ET in short). The goal is to demonstrate some of the building blocks and basic concepts of the module.

XML tree and elements

XML is an inherently hierarchical data format, and the most natural way to represent it is with a tree. ET has two classes for this purpose – `ElementTree` represents the whole XML document as a tree, and `Element` represents a single node in this tree. Interactions with the whole document (reading and writing to/from files) are usually done on the `ElementTree` level. Interactions with a single XML element and its sub-elements are done on the `Element` level.

Parsing XML

We'll be using the following XML document as the sample data for this section:

```
<?xml version="1.0"?>
<data>
    <country name="Liechtenstein">
        <rank>1</rank>
```

Parsing JSON

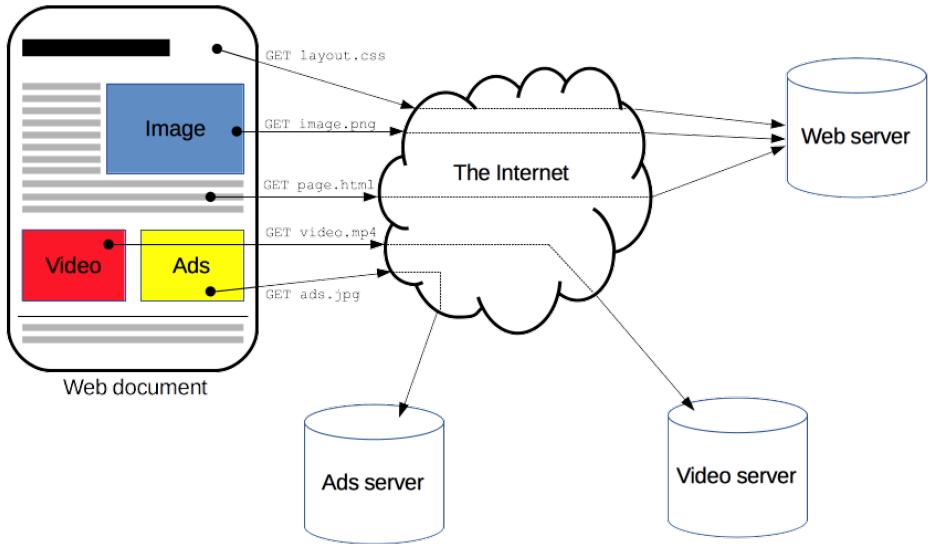
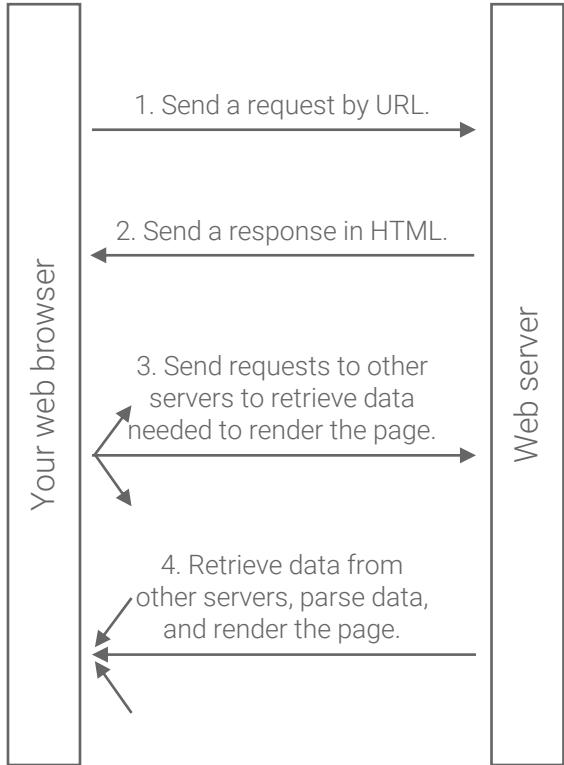
- json (built-in)

```
1 import json
2 a = json.loads(open('simple.json').read())
3 a['mylist']
4 a['mydict']
5 with open('simple2.json', 'w') as f: f.write(json.dumps(a))
```

```
simple.json
{
    "mylist": [ 1, 2, "a", "b" ],
    "mydict": { "3": "c", "4": "d" }
}
```

Simple Exercise

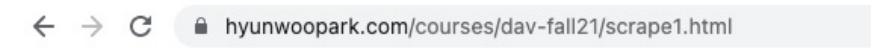
Web request flow



<https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>

❶ Three different situations when scraping

- Scenario 1
 - <https://hyunwoopark.com/courses/dav-fall21/scrape1.html>
- Scenario 2
 - <https://hyunwoopark.com/courses/dav-fall21/scrape2.html>
- Scenario 3
 - <https://hyunwoopark.com/courses/dav-fall21/scrape3.html>



This is a test page for scraping exercise.



This is a test page for scraping exercise.



This is a test page for scraping exercise.

Let's try.

- First example: scrape1.html

```
import requests
r = requests.get('https://hyunwoopark.com/courses/dav-fall21/scrape1.html')
print(r.content.decode('ascii'))
```

```
<!doctype html>
<html>
<body>
<h1>This is a test page for scraping exercise.</h1>
</body>
</html>
```

```
from bs4 import BeautifulSoup as soup
s = soup(r.content)
s.h1.text
```

```
'This is a test page for scraping exercise.'
```

↗ IFRAME and asynchronous DOM update

- Second example

```
u = 'https://hyunwoopark.com/courses' \
+'/dav-fall21/scrape2.html'
r = requests.get(u)
print(r.content.decode('ascii'))
```

```
<!doctype html>
<html>
<head>
  <style type="text/css">
    body { margin: 0px; }
    iframe {
      width: 100%;
      border: 0px;
    }
  </style>
</head>
<body>
<iframe src="scrape1.html"></iframe>
</body>
</html>
```

- Third example

```
u = 'https://hyunwoopark.com/courses/dav-fall21/scrape3.html'
r = requests.get(u)
print(r.content.decode('ascii'))
```



```
<!doctype html>
<html>
<head>
  <script type="text/javascript">
    document.addEventListener("DOMContentLoaded", function(event) {
      var h1 = document.createElement("h1");
      h1.innerHTML = "This is a test page for scraping exercise.";
      document.getElementsByTagName('body')[0].appendChild(h1);
    });
  </script>
</head>
<body>
</body>
</html>
```

Real-world case

- Related shop items are fetched as JSON asynchronously.



- Comments are in an IFRAME.



Another exercise

- https://www.yestrade.go.kr/common/common.do?jPath=/im/imCt010L&CURRENT_MENU_CODE=MENU1120

 전략물자관리제도 안내 판정허가신청 기타민원 신청 자율준수무역거래자 일립/정보마당 이용안내 

일립/정보마당 > 일립/정보마당 > 전략물자(이중용도)

전략물자(이중용도)

• 게시물 검색 HSKCODE 

※ 본 정보는 HSK 연개표 정보로써 전략물자 해당 가능성이 높은 HS 번호의 품목군과 관련 통제번호의 정보를 제공하는 것임에 유의하시기 바랍니다.

총 8959 개 / 현재페이지: [1/896]

품목분류(HS)	HSK품목명	HSK영문명	통제번호
7218101000	잉곳(ingot)	Ingots	1C118.
7218109000	기타	Other	1C118.
7218912000	시트비(sheet bar)	Sheet bars	1C118.
7218991000	블룸(bloom)	Blooms	1C216.
7218992000	빌릿(billet)	Billets	1C216.
7218999000	기타	Other	1C216.
7219129000	기타	Other	1C116.
7219129000	기타	Other	1C216.
7219131010	니켈 함유량이 6% 미만이고, 망간 함유량이 3% 이상인 것	Containing nickel less than 6% and containing manganese at least 3% by weight	1C116.
7219131010	니켈 함유량이 6% 미만이고, 망간 함유량이 3% 이상인 것	Containing nickel less than 6% and containing manganese at least 3% by weight	1C118.

« « 1 / 2 / 3 / 4 / 5 / 6 / 7 / 8 / 9 / 10 » »

 전략물자 기본정보

품목분류(HS)	7218101000
HSK품목명	잉곳(ingot)
통제번호	1C118.

Class Survey
Due 9/13
before class 2:00pm KST

Example Response

<https://forms.gle/SBsgZpFcAciLYRjP8>

Class Survey for Data Analysis and Visualization Fall 2021

Your personal information will be used for project group formation. Your responses on Kaggle datasets will be used as a dataset for teaching material.

All questions must be answered before you submit the form.

You must be signed in with your SNU email to access and submit this form.

If you want to change your response, please submit this form again. If you submit multiple responses, only your last response will be used.

hyunwoopark@snu.ac.kr Switch account  Draft saved

* Required

SNU Email *
Your response must ends with @snu.ac.kr
hyunwoopark@snu.ac.kr

Student Number *
2002-11992

Name (in English) *
The same name as in eTL. Please use [Last, First] format.
Park, Hyunwoo

Gender *
 Female
 Male
 Prefer not to say
 Other: _____

Age Group *
 0 ~ 19
 20 ~ 24
 25 ~ 29
 30 ~ 34
 35 ~ 39
 40 and above
 Prefer not to say

Undergraduate Major (If multiple, separate by semicolon.) *
Electrical Engineering

Graduate Major (If multiple, separate by semicolon. If none, say None.) *
Information Management and Systems; Indust

Choose languages that you have experience programming more than 100 lines. *
 Python
 JavaScript
 HTML
 CSS
 Other: _____

What is your self-evaluated proficiency in Python? *
1 2 3 4 5 6 7
No experience Very comfortable

What is your self-evaluated proficiency in web programming? *
1 2 3 4 5 6 7
No experience Very comfortable

Choose your five favorite Kaggle datasets with weights. *
Your answer to this question should be five lines. Each line should be URL [space] weight. The weights should sum to 100. Example response is as attached. Your score for this class survey will be deducted if your response does not follow the instructions.

Choose your five favorite Kaggle datasets with weights. *
Your answer to this question should be five lines. Each line should be URL [space] weight. The weights should sum to 100. Example response is as attached.

https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings 20
https://www.kaggle.com/neuromusic/avocado-prices 30
https://www.kaggle.com/jessical9530/honey-production 15
https://www.kaggle.com/uclm/red-wine-quality-cortez-et-al-2009 5
https://www.kaggle.com/nickhould/craft-cans 30

https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings 20
https://www.kaggle.com/neuromusic/avocado-prices 30
https://www.kaggle.com/jessical9530/honey-production 15
https://www.kaggle.com/uclm/red-wine-quality-cortez-et-al-2009 5
https://www.kaggle.com/nickhould/craft-cans 30

Submit Clear form

Never submit passwords through Google Forms.

This form was created inside of 서울대학교. [Report Abuse](#)

Google Forms

_datasets for this course

- Kaggle (<https://www.kaggle.com/>)

kaggle Search Competitions Datasets Kernels Discussion Learn ...

Documentation New Dataset

Datasets

Public Your Datasets Favorites Sort by Most Votes

13,194 Datasets Sizes File types Licenses Tags Search datasets

Credit Card Fraud Detection
Anonymized credit card transactions labeled as fraudulent or genuine
Machine Learning Group - ULB updated 8 months ago (Version 3)

2239 CSV 1k 36 1m

crime finance

European Soccer Database
25k+ matches, players & teams attributes for European Professional Football
Hugo Mathien updated 2 years ago (Version 10)

1483 SQLite 1k 87 529k

association... europe

↗ My shortlist datasets

- <https://bit.ly/hp-kaggle-datasets>
- I browsed public Kaggle datasets and handpicked a set of datasets relevant to business and management.
- This list will be updated after this semester reflecting your input.

GitHub Gist Search... All gists Back to GitHub

oksure / HP's Kaggle Dataset Shortlist.md Last active 1 minute ago

Code Revisions 9 Stars 1 Edit Delete Star 1 Download ZIP Raw

1. Sales

Video Game Sales with Ratings <https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings>
Video Game Sales <https://www.kaggle.com/gregorut/videogamesales>
New Car Sales in Norway <https://www.kaggle.com/dmi3kno/newcarsalesnorway>
Brooklyn Home Sales, 2003 to 2017 <https://www.kaggle.com/tianhwu/brooklynhomes2003to2017>
House Sales in King County, USA <https://www.kaggle.com/harlxoxem/housesalesprediction>
Black Friday <https://www.kaggle.com/mehdidag/black-friday>

2. Operations

Historical Sales and Active Inventory <https://www.kaggle.com/flenderson/sales-analysis>
2015 Flight Delays and Cancellations <https://www.kaggle.com/usdot/flight-delays>
Airlines Delay <https://www.kaggle.com/giovamatata/airlinedelaycauses>
Uber Pickups in New York City <https://www.kaggle.com/fivethirtyeight/uber-pickups-in-new-york-city>
Telco Customer Churn <https://www.kaggle.com/blastchar/telco-customer-churn>

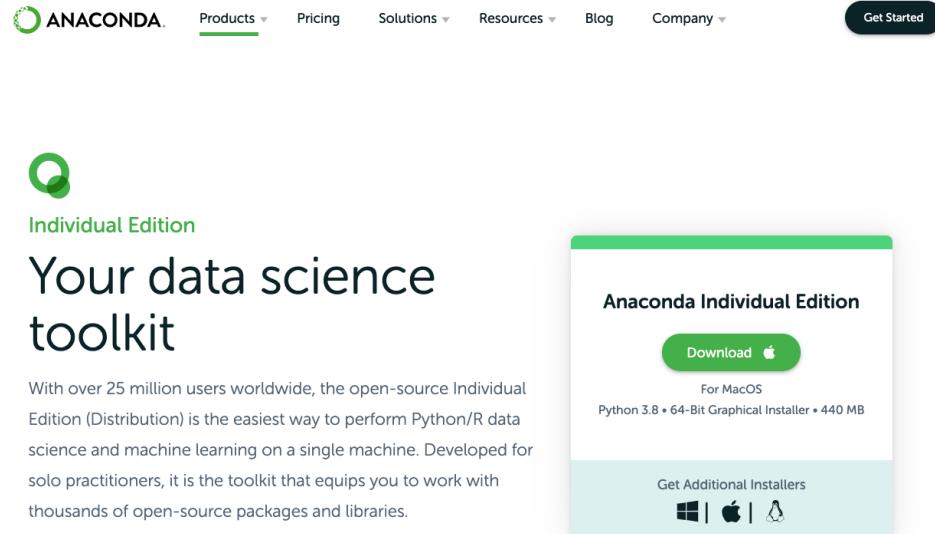
3. Retail & Commerce & Transactions

Retail Data Analytics <https://www.kaggle.com/manjeetsingh/retaildataset>

Things To Do

▶ Install Python 3 Anaconda distribution

- <https://www.anaconda.com/products/individual>
- Python 3.8 (Version 2021.05)
- Included packages:
 - https://docs.anaconda.com/anaconda/packages/py3.8_osx-64/



The screenshot shows the Anaconda Individual Edition landing page. At the top, there's a navigation bar with links for Products, Pricing, Solutions, Resources, Blog, and Company. A "Get Started" button is also visible. Below the navigation, there's a large green "Q" logo followed by the text "Individual Edition". The main headline reads "Your data science toolkit". A paragraph below it states: "With over 25 million users worldwide, the open-source Individual Edition (Distribution) is the easiest way to perform Python/R data science and machine learning on a single machine. Developed for solo practitioners, it is the toolkit that equips you to work with thousands of open-source packages and libraries." To the right, there's a download section for "Anaconda Individual Edition" specifically for Mac OS, showing a "Download" button and a file size of 440 MB. Below this, there's a link to "Get Additional Installers" with icons for Windows, macOS, and Linux.

Launch Jupyter Notebook

The screenshot shows two views of a Jupyter Notebook interface. The top view is the main dashboard at `localhost:8888/tree`, featuring a file tree with one item: "Example Networks.ipynb" (Running, 8 days ago, 441 kB). The bottom view is an open notebook at `localhost:8888/notebooks/Untitled.ipynb?kernel_name=python3`, showing a single code cell labeled "In []:".

localhost:8888/tree

jupyter

Files Running Clusters

Select items to perform actions on them.

Upload New

0 / Name Last Modified File size

Example Networks.ipynb Running 8 days ago 441 kB

localhost:8888/notebooks/Untitled.ipynb?kernel_name=python3

jupyter Untitled (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In []:

Get to know the terminal interface

```
(base) MacBook-Pro:python_code oksure$ which python
/Users/oksure/opt/anaconda3/bin/python
(base) MacBook-Pro:python_code oksure$ which jupyter
/Users/oksure/opt/anaconda3/bin/jupyter
(base) MacBook-Pro:python_code oksure$ jupyter console
Jupyter console 6.4.0

Python 3.8.8 (default, Apr 13 2021, 12:59:45)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.22.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]:
Do you really want to exit ([y]/n)?
Shutting down kernel
(base) MacBook-Pro:python_code oksure$ [IPKernelApp] WARNING | Parent appears to have exited, shutting down.

(base) MacBook-Pro:python_code oksure$ jupyter notebook
[I 2021-09-02 12:11:37.438 LabApp] JupyterLab extension loaded from /Users/oksure/opt/anaconda3/lib/python3.8/site-packages/jupyterlab
[I 2021-09-02 12:11:37.438 LabApp] JupyterLab application directory is /Users/oksure/opt/anaconda3/share/jupyter/lab
[I 12:11:37.442 NotebookApp] Serving notebooks from local directory: /Users/oksure/Dropbox/Research_SupplyNetwork/python_code
[I 12:11:37.442 NotebookApp] Jupyter Notebook 6.3.0 is running at:
[I 12:11:37.442 NotebookApp] http://localhost:8888/?token=88cbcfda9fdb3afc5d675d6f61b1a7c551e5493321c13a2d
[I 12:11:37.442 NotebookApp] or http://127.0.0.1:8888/?token=88cbcfda9fdb3afc5d675d6f61b1a7c551e5493321c13a2d
[I 12:11:37.442 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 12:11:37.461 NotebookApp]
```

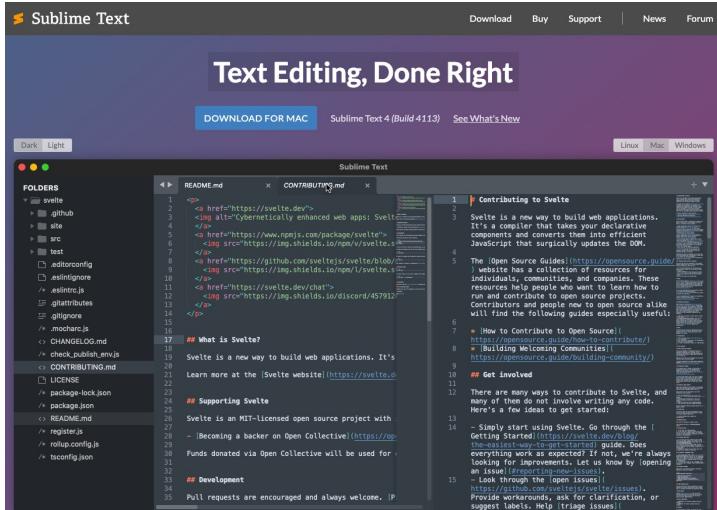
To access the notebook, open this file in a browser:
file:///Users/oksure/Library/Jupyter/runtime/nbserver-61601-open.html
Or copy and paste one of these URLs:
<http://localhost:8888/?token=88cbcfda9fdb3afc5d675d6f61b1a7c551e5493321c13a2d>
[or http://127.0.0.1:8888/?token=88cbcfda9fdb3afc5d675d6f61b1a7c551e5493321c13a2d](http://127.0.0.1:8888/?token=88cbcfda9fdb3afc5d675d6f61b1a7c551e5493321c13a2d)

[0] 1:bash 2:bash- 3:bash 4:python3.8*

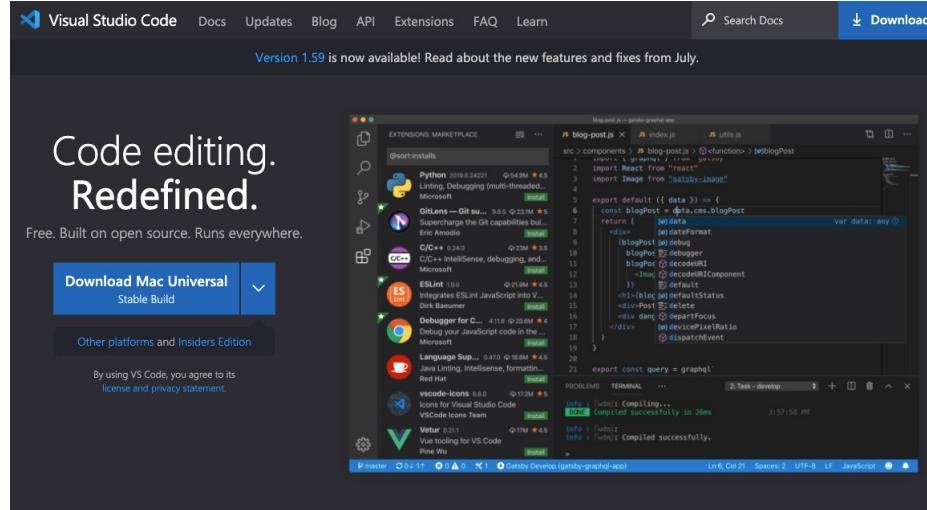
"MacBook-Pro.local" 12:11 02-Sep-21

💡 Pick your code/text editor

- Sublime Text
 - <https://www.sublimetext.com/>



- VS Code
 - <https://code.visualstudio.com/>



Things to do / Reminders

- Class Survey (5%)
 - Due on 9/13 (Tue) before class 2:00pm KST.
- Technical stuff
 - Install Stata
 - Install the Anaconda distribution of Python 3
 - Launch Jupyter Notebook
 - Get to know the terminal interface
 - Pick a text/code editor
- Review
 - Check out the following Python packages.
 - Must: requests, bs4, json
 - Optional: selenium, scrapy, xml
- eTL boards
 - Q&A Board for questions and answers
 - Discussion Board for resources or ideas to share
 - Group Project Board for forming a group and recruiting people
- See you Thursday, 9/8.
 - Data Cleaning / Storing / Merging