Data Science App Development

DIANE WOODBRIDGE, PH.D

Contents

Class Intro

Collaboration Tools
- Code Convention
- Virtual Environment
- Automated Documentation

Backend
- Data Acquisition and Storage

Contents

Class Intro

Collaboration Tools

- Code Convention
- Virtual Environment
- Automated Documentation

Backend

- Data Acquisition and Storage

Course Objectives

Entrepreneur class for designing and developing a data science web application.

The course will cover overall techniques and collaboration tools.

- Brush up on Python - Coding conventions, Documentation, Testing, etc.
- Learn basic front and back-end development.
- Apply web analytics tools for tracking website traffic and user activities.
- Design and develop a web application and deploy on AWS.

Students are encouraged to apply technical techniques and business strategies that they learned from previous courses.

This class also requires to present a business plan and progress regularly.

- This should meet the standards from Big Data Business Strategies (MSAN 603).

Tentative Course Schedule

March 23 – Intro, Collaboration, Data Acquisition and St
March 30 – No class
April 6 – Group Presentation
April 13 – Backend, Quiz 1
April 20 – Frontend
April 27 - Web Analytics
May 4 – Group Presentation, Quiz 2
May 16 – Final (Presentation to VC)

Reference Materials

Kenneth Reitz. *The Hitchhiker's Guide to Python*. http://docs.python-guide.org/en/latest/

Grinberg, Miguel. *Flask web development: developing web applications with python*. " O'Reilly Media, Inc.", 2018.

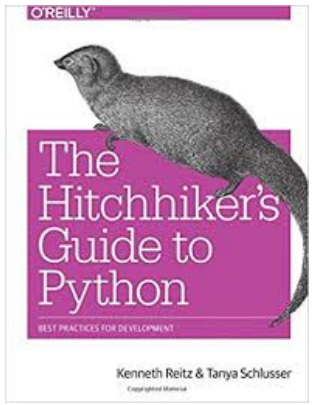
HTML5 Tutorial. https://www.w3schools.com/html/

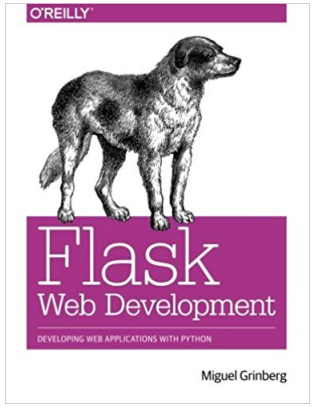
CSS Tutorial. https://www.w3schools.com/css/

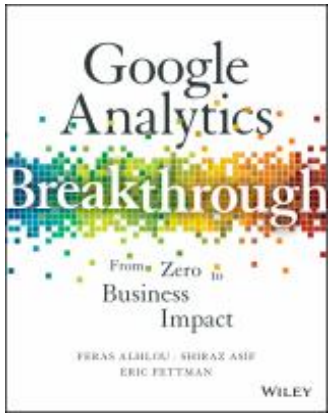
Duckett, Jon. *HTML and CSS: design and build websites*. John Wiley & Sons, 2011.

Google Analytics Academy (Beginner, Advanced). https://analytics.google.com/analytics/academy/

Alhlou, Feras, Shiraz Asif, and Eric Fettman. *Google Analytics Breakthrough: From Zero to Business Impact*. John Wiley & Sons, 2016.

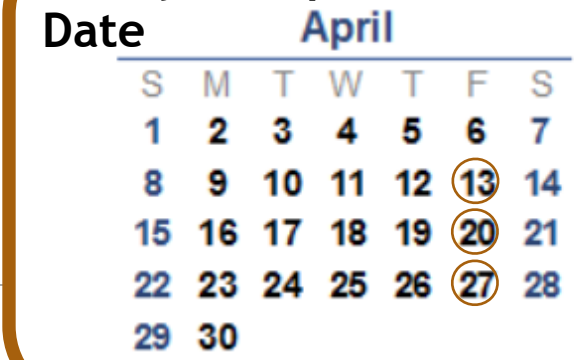






Course Evaluation

Friday Group Presentation Date



Attendance (13%)
- Class : 6%
- Group Presentation Attendance (Friday 2:00 - 4:00 starting from April 13)
  - Need to attend at least one day : 3%
  - Room 453
- Final Presentation on May 16th 10:00 – 12:00 : 4%
- No cellphones, No social media, No Slack!

Homework (16 %) – Individual Assignment
- No late submissions/resubmissions allowed!! (We are going to run the grading code only once.)

Group Project (55%) – Backend, Frontend and Web Analytics
- Code
- Documentation
- Presentation/Demo

Quiz (Multiple choices/Short answers/Fill in the blank/Text) (16%)
- Date : 9 – 10 am. April 20th and May 4th.

Office Hour

Tuesday 12:00– 1:00 PM (101 Howard #522)

Others…

Example Data

- https://github.com/dianewoodbridge/2018-msan698-example.git

Poll

- https://pollev.com/dianewoodbri311

Last Year's Finalists

Acute Commute

- Our application considers all forms of travel to and from the transit stops to provide additional options and give the shortest possible travel time for the trip.

Match.job

- We focus on a skill-based job search that will allow the user to focus on finding jobs that match their skillset instead of a specific job title or position or industry. We propose to allow our clients to perform these types of queries.

MedHub

- We aim to build a single app for all of a person's health related needs. It will be a personalized app which will let a user input their symptoms/disease, recommend a doctor, and schedule an appointment. The user could alternately input the name of a drug, and the app would provide a list of doctors who prescribe that drug. It will also store a history of all past diseases/symptoms, doctors visited, medications prescribed, etc. Additionally, the user would be able to set reminders for taking pills, refills, and follow-up appointments in the app.

Last Year's Finalists

Move Anywhere

- Not only will we aggregate and provide raw data from several sources and for several cities, but we also will provide analytics that answer important questions. We will use user-input to find what the user needs and is looking for. For example instead of providing a map with all of the apartments for rent in a city, we will use the expected income, the family size or relationship status of the user, whether the user has a pet, and how important "walkability" of a neighborhood is for the user to suggest 4 or 5 apartments or homes for rent or for sale. Our analytics will fill in the gaps. (i.e if you know what career you pursue, how big you want your house to be, and a walkable neighborhood, we suggest a city.)

Prefer

- We are going to create a simple, image-centric, swipe-left/swipe-right app. On the backend it will use collaborative filter embeddings along with user/product metadata to rank items by predict user preference. We will use an explore/exploit approach to shows users things they like while sometimes showing them something new and learning more about their preferences.

Last Year's Finalists

SFPal
- There are many factors to consider for an individual moving to a big city: rent price, commute time, safety, nightlife, restaurants, outdoor activities, noise, etc. Although you might be able to discover that information from a friend (or a friend's friend) we merge that data altogether onto one platform to make decisions easier for you. We leverage data to help you decide on an area to live in that caters most to your needs.

TOBBL-Tower of Babel
- We are trying to create a platform where people can discuss their ideas and come to a point of resolution. The data generated will also allow us to connect users based on similar ideologies and opinions. We will implement a graph-network conversation with relationship types such as 'support,' 'oppose,' and others which will propagate upvotes to end-nodes which act as answers to specific questions.

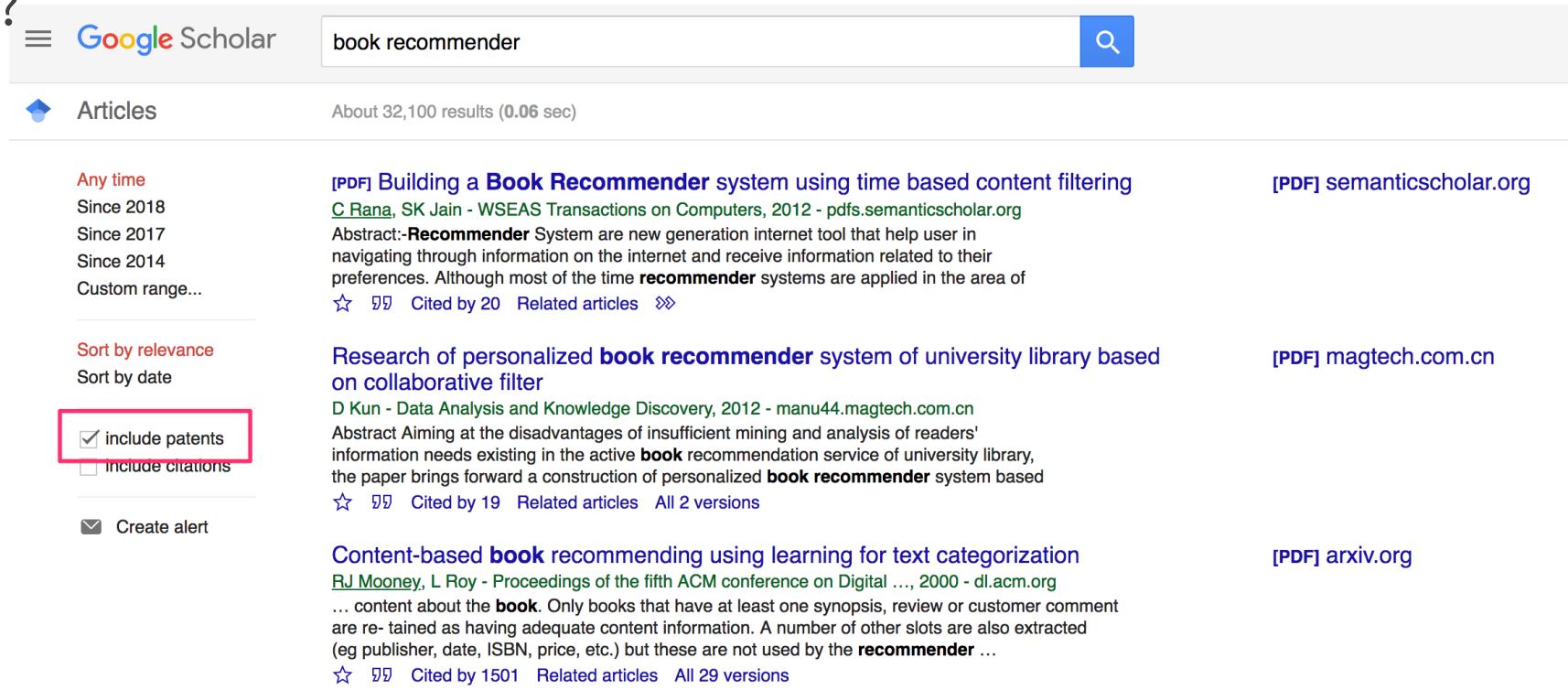
YelpInterest
- Our target is customers searching Yelp for a particular food or dish. Instead of simply listing restaurants as Yelp does now, our aim is to return search results in the form of pictures of the dish as served at the restaurants.

Project Ideas

Project Ideas

Is there an already existing application?

- What is the difference?
- Why yours is better?
  - Validation?



Possible Outcomes

Possible Outcomes

Possible Outcomes

Forecasting Smart Meter Energy Usage using Distributed Systems and Machine Learning

Chris Dong*, Lingzhi Du*, Feiran Ji*, Zizhen Song*, Yuedi Zheng*,
Paul Intrevado, Diane Myung-kyung Woodbridge
{cadong,ldu4,fji3,zsong11,yzheng41,pintrevado,dwoodbridge}@usfca.edu
Data Science Program
University of San Francisco

Distributed Data Analytics Framework for Smart Transportation

Alexander J. Howard*, Tim Lee*, Sara Mahar*, Paul Intrevado, Diane Myung-kyung Woodbridge
{ajhoward7,semahar2,tdlee,pintrevado,dwoodbridge}@usfca.edu
Data Science Program
University of San Francisco

Project Requirements

Languages/Tools

- Python 3.6
- Flask
- Database – SQL / NoSQL
- HTML, CSS, JavaScript, etc.
- Amazon Web Services

➢ Must specify what you used in your documentation.

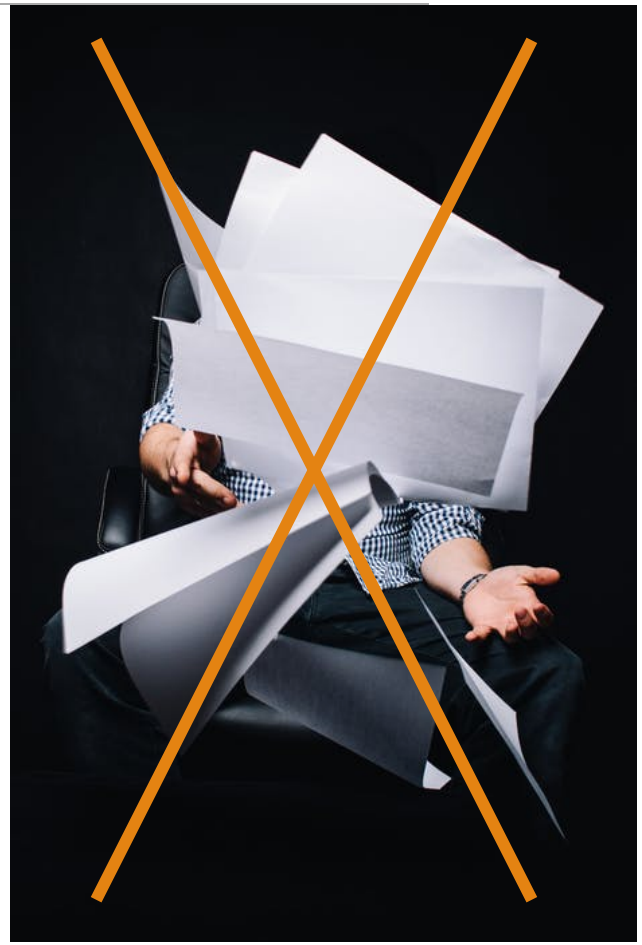
If you are using something else..

- Understand what the tool is about. (No copy and paste and this shouldn't be your bottleneck.)
- Make sure you have automated documentation tools supporting it.

Project Requirements

Project presentation

- One to three team members present their progress on April 13, 20 and 27.
  - A presentation should be 5 minutes maximum.
- You need to present at least ONCE.
  - If you don’t present during the module, your presentation credit is 0.
  - If you present more than once, the maximum of two/three will be your presentation credit.
- You need to come and peer-review at least one day.



Why Python 3?

Python 2.7 will retire in 2020.

Python 3

- The most drastic improvement is the better Unicode support and mostly impact low-level networking developers.
- Improved standard library modules, security and bug fixes.
- More : What's new in Python.

However, make sure the 3rd party packages that you are planning to use is compatible with Python 3.

- Possibly you can port Python 2 code to Python 3.

➢ **In this class, we are going to use Python 3.6**

https://wiki.python.org/moin/Python2orPython3

Contents

Class Intro

Collaboration Tools

- Code Convention
- Virtual Environment
- Automated Documentation

Backend

- Data Acquisition and Storage

Collaboration

This class requires collaborations.

- Things to consider..
  - Code Standard
  - Environment Setup
  - Documentation
  - And many more!

Contents

Class Intro

Collaboration Tools
- **Code Convention**
- Virtual Environment
- Automated Documentation

Backend
- Data Acquisition and Storage

Code Style

For collaborative environment, using a consistent style provides better code management, readability, understandability and maintenance.

Follow commonly used or internally agreed code conventions.

- Code Conventions : Commonly used and recommended choices as a more readable option.
- PEP (Python Enhancement Proposals)
  - PEP 8 : Style Guide for Python Code
  - PEP 20 : The Zen of Python

PEP 8 - Style Guide for Python Code

PEP 8 covers code layout, whitespace, naming conventions and other style related topics.

- Code layout
- Whitespaces
- Comments
- Naming Conventions

https://www.python.org/dev/peps/pep-0008

PEP 8 - Style Guide for Python Code

PEP 8 Covers code layout, whitespace, naming conventions and other style related topics.

- Code layout
  - Max line length: 79 characters.
  - Indentation
    - Use a tab (4 spaces) per indentation level.
    - Continued lines - long lines should be broken over multiple lines.
  - Blank lines
    - Surround top-level function and class definitions with two blank lines.
    - Method definitions inside a class are surrounded by a single blank line.
  - Importing libraries/packages
    - Imports are always put at the top of the file.
    - Imports should usually be on separate lines.
    - Imports should be grouped in the following order and you should place a line between each group.
      - standard library imports
      - related third party imports
      - local application/library specific imports

https://www.python.org/dev/peps/pep-0008

Homework

```
def function_with_many_many_argunemts(variable_1, variable_2, variable_3, variable_4):
    print("wow")

def another_function():
    print("wow")

import sys, os
```

PEP 8 - Style Guide for Python Code

PEP 8 Covers code layout, whitespace, naming conventions and other style related topics.

- White spaces
  - No whitespace
    - Immediately inside parentheses, brackets or braces.
    - Between a trailing comma and a following close parenthesis.
    - Immediately before a comma, semicolon, or colon.
      - However, in a slice the colon acts like a binary operator, and should have equal amounts on either side
    - Immediately before the open parenthesis that starts the argument list of a function call.
    - Immediately before the open parenthesis that starts an indexing.
  - Surround the following binary operators with a single space on either side.
    - assignment (=), augmented assignment (+=, -= etc.), comparisons (==, <, >, !=, <>, <=, >=, in, not in, is, is not),
      Booleans (and, or, not)
  - If operators with different priorities are used, consider adding whitespace around the operators with the lowest priority(ies).
    - Ex. x = x*2 – 1
  - Don't use spaces around the = sign when it is used to assign a keyword argument or a default parameter value.
  - Multiple statements on the same line are discouraged.

https://www.python.org/dev/peps/pep-0008

Homework

```
def insert_function( data ):
    preprocessed_data= [1 , 2]

def insert_function(data = None):
    return True
```

PEP 8 - Style Guide for Python Code

PEP 8 Covers code layout, whitespace, naming conventions and other style related topics.

- Comments
  - Comments should be complete sentences written in English.
  - Block comments: Block comments generally consist of one or more paragraphs built out of complete sentences, with each sentence ending in a period.
    - Each line of a block comment starts with a # and a single space.
    - block comments are indented to the same level as that code.
  - Inline comments: An inline comment is a comment on the same line as a statement.
    - Inline comments should be separated by at least two spaces from the statement.
    - They should start with a # and a single space.
    - Use inline comments sparingly.
  - Docstring :
    - Write docstrings for all public modules, functions, classes, and methods.
    - For one line docstrings, keep the closing """ on the same line.
    - The """ that ends a multiline docstring should be on a line by itself.

https://www.python.org/dev/peps/pep-0008

Docstring

Docstring

- Describe the operation of the function or class.
  - All modules should normally have docstrings, and all functions and classes exported by a module should also have docstrings.
- Use triple double quotes (""") around docstrings.
- Docstrings will be shown in an interactive Python session when you type help(module.function).

```
>>> import hw1
>>> help(hw1.drop)
```

```
Help on function drop in module hw1:

drop(db, table)
    Designed specifically for MonogoDB.
    Todo: extend it for other DBs.
(END)
```

Homework

```
def drop(db, table):
    """
    Designed specifically for MonogoDB.
    Todo: extend it for other DBs.
    """
    find(db).drop(table)
     #An error will occur, if db or table doesn't exist.
```

PEP 8 - Style Guide for Python Code

PEP 8 Covers code layout, whitespace, naming conventions and other style related topics.

- Naming Conventions
  - Naming styles:
    - b (single lowercase letter) – Never use 'l' (lowercase letter el), 'O' (uppercase letter oh), or 'I' (uppercase letter eye).
    - B (single uppercase letter)
    - lowercase, lower_case_with_underscores
    - UPPERCASE, UPPER_CASE_WITH_UNDERSCORES
    - CapitalizedWords (or CapWords, or CamelCase)
    - mixedCase
    - _single_leading_underscore: weak "internal use" indicator. (won't be imported.)
    - Packages and modules
      - Packages: Packages should have short, all-lowercase names, although the use of underscores is discouraged.
      - Modules: Modules should have short, all-lowercase names. Underscores can be used in the name.
    - Function and variables
      - Function names: should be lowercase, with words separated by underscores.
      - Variable names: follow the same convention as function names..

https://www.python.org/dev/peps/pep-0008

Homework

```
def DELETEFROMTABLE(data):  # Fix this: Readability?
    PROCESSED_Data = preprocess(data)
    return False
```

Too many to remember?

PEP 8 is explicit enough and can be checked programmatically.

```
$ pip install pycodestyle
$ pycodestyle your_code.py
```

```
ML-ITS-603436:python_bestpractice dwoodbridge$ pycodestyle ex1.py
ex1.py:27:15: E261 at least two spaces before inline comment
ex1.py:27:16: E262 inline comment should start with '# '
ex1.py:27:80: E501 line too long (96 > 79 characters)
ex1.py:30:1: E265 block comment should start with '# '
ex1.py:31:80: E501 line too long (89 > 79 characters)
```

PEP 20 – Principle for Python's Design

The Zen of Python

```
Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although never is often better than *right* now.
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
Namespaces are one honking great idea -- let's do more of those!
```

https://www.python.org/dev/peps/pep-0020/

Contents

Class Intro

Collaboration Tools
- Code Convention
- **Virtual Environment**
- Automated Documentation

Backend
- Data Acquisition and Storage

Environment

A project team member might work on several projects requiring different versions of Python or other packages.

Virtual environment

- Keep dependencies required by different project in separate places.
- Easily switch to a environment that requires different dependencies.

Environment

```
$ pip install virtualenv
```

1. Go to your project directory and create a virtual environment.

```
$ virtualenv --python python3 python3_env
```

2. This will generate python3_env directory and you need to activate the virtual environment.

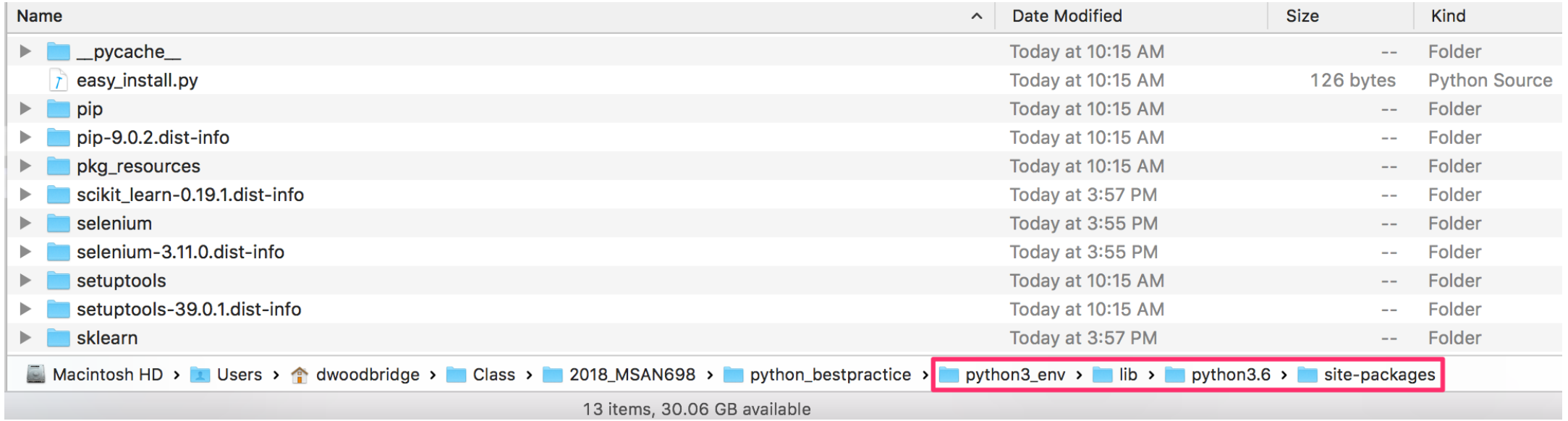
```
$ source python3_env/bin/activate
```

`Output :` (python3_env) ML-ITS-603436:python_bestpractice dwoodbridge$

Environment

3. Add libraries to the virtual environment (Installed in python3_env/lib/python3.6/site-packages).

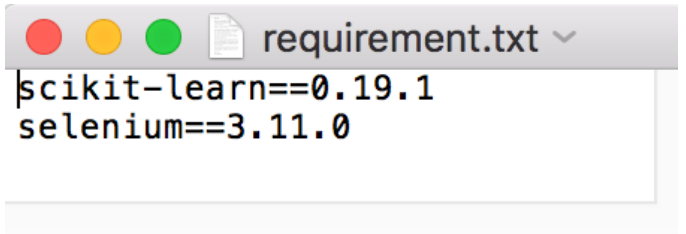
```
$ pip install selenium
$ pip install scikit-learn
```



Environment

4. To build your own projects for others, freeze will write all the installed packages to the file requirement.txt.

```
$ pip freeze > requirement.txt
```



5. Using requirement.txt, collaborators can install all of the dependencies in their virtual environment.

```
$ pip install -r requirement.txt
```

```
(another-env) ML-ITS-603436:python_bestpractice dwoodbridge$ pip install -r requirement.txt
Collecting scikit-learn==0.19.1 (from -r requirement.txt (line 1))
  Using cached scikit_learn-0.19.1-cp36-cp36m-macosx_10_6_intel.macosx_10_9_intel.macosx_10_9_x86_64.macosx_10_10_intel.macosx_10_10_x86_64.whl
Collecting selenium==3.11.0 (from -r requirement.txt (line 2))
  Using cached selenium-3.11.0-py2.py3-none-any.whl
Installing collected packages: scikit-learn, selenium
Successfully installed scikit-learn-0.19.1 selenium-3.11.0
```

Environment

5. To return to normal system settings,

```
$ deactivate
```

Contents

Class Intro

Collaboration Tools
- Code Convention
- Virtual Environment
- **Automated Documentation**

Backend
- Data Acquisition and Storage

Documentation

README

- General information written in plain text to users and programmers of a project.
- Contents
  - A few lines explain the purpose of the project or library.
  - Contributors (Authors).
  - The URL of the main source.
  - Change logs.
  - Copyrights.

https://en.wikipedia.org/wiki/README

Documentation

Project Publication

- Your project publication might include more detailed information.
  - Introduction and overview of the project
  - Contributors (Authors) and their roles.
  - Tutorial - detailed step-by-step instructions to setup and run the code.
  - API reference – API information (written in docstring) and actual code.
  - Developer guidance – code conventions and design strategy for other contributors.

Documentation

Sphinx

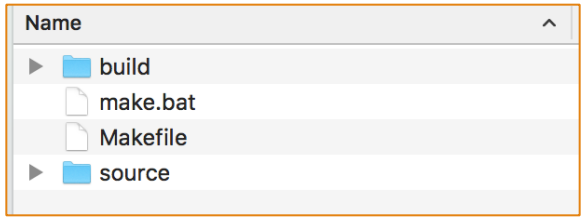
- Most popular Python documentation generation tool.
- Generate html, LaTex, txt, etc. Sphinx takes docstrings in your code and texts in reStructuredText (.rst) to create a code document.
- Installation

```
$ pip install sphinx
```

- Set up a document source directory

```
$ sphinx-quickstart
```

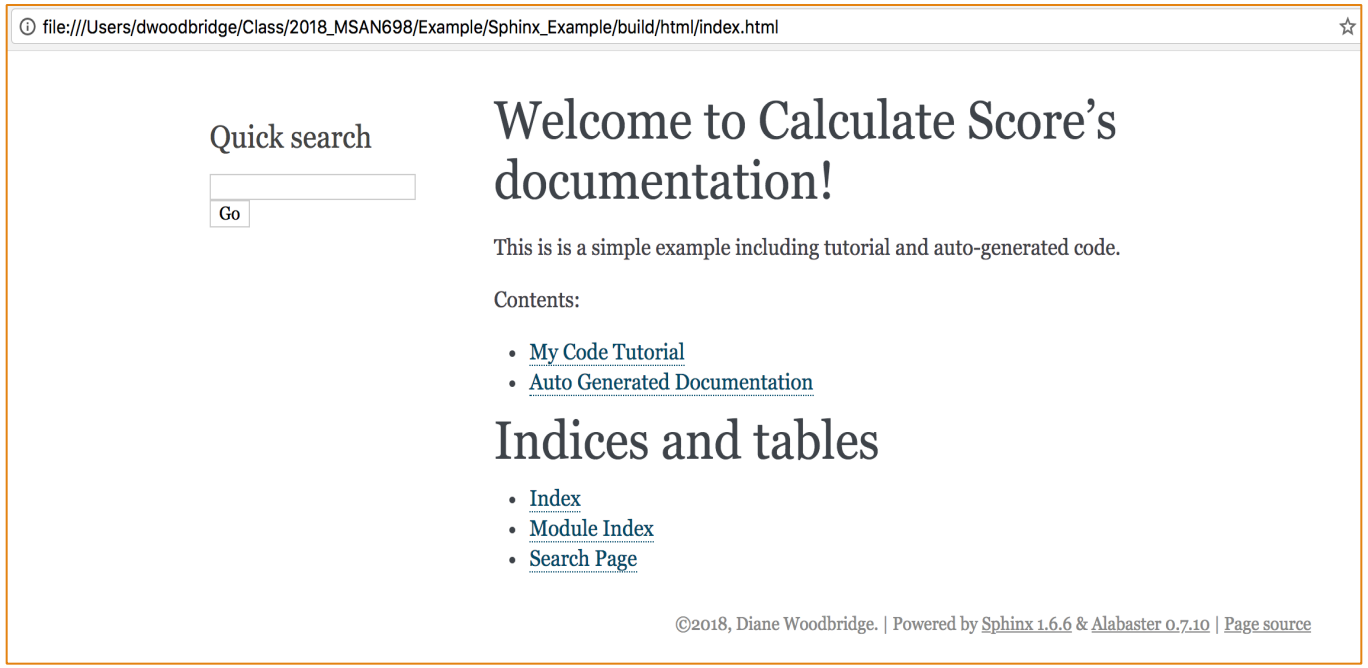
  - Creates a document source directory and conf.py with configuration options that you choose.



Documentation

Sphinx

- Generate and Publish automated document on Git.
  - Follow this tutorial.



Homework

.rst are in the github repo (Week1/Sphinx).

Fix hw1.py and publish your documents.

- Make sure to…
  - Change the author name.
  - Update tutorial (code name, etc.).
  - Updated hw1.py should be added.

Contents

Class Intro

Collaboration Tools
- Code Convention
- Virtual Environment
- Automated Documentation

Backend
- Data Acquisition and Storage

Backend Development

Definition

- The backend of an application is responsible business logic, data acquisition, database interaction, data processing, etc.
  - Backend code is run on the server.

Contents

Class Intro

Collaboration Tools
- Code Convention
- Virtual Environment
- Automated Documentation

Backend
- **Data Acquisition and Storage**

Data Acquisition and Storage

Brainstorm what you need to collect for your application.

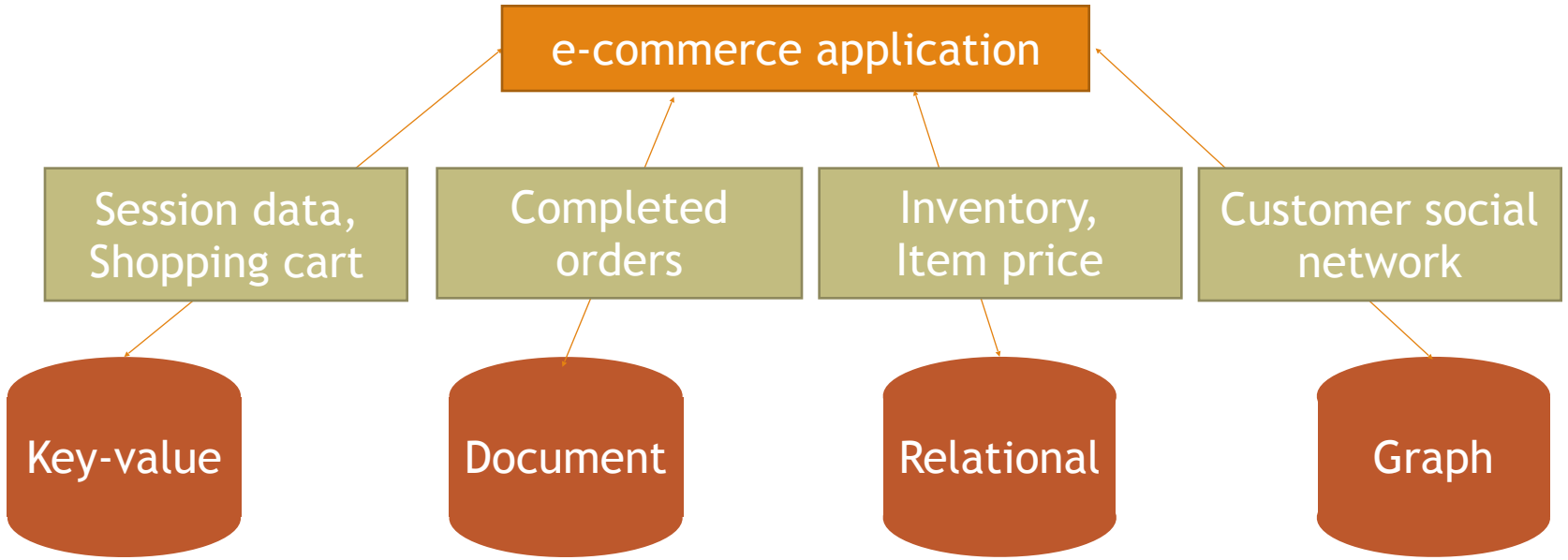
- Things to consider.
  - What are the main data that you need to collect?
  - Are the data sets that you need even available? (Or you can collect on time?)
    - Farmers real-time crop data in Petaluma?
    - HIPPA regulated patient health record?
    - IRB approval required data?
- Techniques
  - Data acquisition (MSAN692 Data Acquisition)
    - Open data set, API calls, Crawling, Selenium, etc.
  - Data storage
    - SQL (MSAN 691 Relational Databases)
    - NoSQL (MSAN 697 Distributed Data Systems)

Note :
Data 1) acquisition and 2) storage deploy code (written in Python 3.6) should be ready to be presented on April 13th.

Choice of DBMS

Polyglot Persistence

- Using multiple data storage technologies, chosen based on the way data is being used by individual applications.
- **NoSQL data stores do not replace relational databases.**



Example 1

Spam and Ham Classification

- Dataset
  - Spam
  - Ham

Training Data

Data

- SMS Spam Collection Data Set, Machine Learning Repository, University of California, Irvine.
  - Source : https://archive.ics.uci.edu/ml/datasets/sms+spam+collection
  - Size : 5574 SMS messages (747 spam and 4827 ham messages.)
  - Format : TSV

```
1  ham  Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got
2  ham  Ok lar... Joking wif u oni...
3  spam Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ent
4  ham  U dun say so early hor... U c already then say...
5  ham  Nah I don't think he goes to usf, he lives around here though
6  spam FreeMsg Hey there darling it's been 3 week's now and no word back! I'd like some fun you up for it
7  ham  Even my brother is not like to speak with me. They treat me like aids patent.
8  ham  As per your request 'Melle Melle (Oru Minnaminunginte Nurungu Vettam)' has been set as your callert
9  spam WINNER!! As a valued network customer you have been selected to receivea £900 prize reward! To clai
10 spam Had your mobile 11 months or more? U R entitled to Update to the latest colour mobiles with camera
11 ham  I'm gonna be home soon and i don't want to talk about this stuff anymore tonight, k? I've cried eno
12 spam SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6da
13 spam URGENT! You have won a 1 week FREE membership in our £100,000 Prize Jackpot! Txt the word: CLAIM to
14 ham  I've been searching for the right words to thank you for this breather. I promise i wont take your
15 ham  I HAVE A DATE ON SUNDAY WITH WILL!!
16 spam XXXMobileMovieClub: To use your credit, click the WAP link in the next txt message or click here>>
17 ham  Oh k...i'm watching here:)
18 ham  Eh u remember how 2 spell his name... Yes i did. He v naughty make until i v wet.
19 ham  Fine if that□s the way u feel. That□s the way its gota b
20 spam England v Macedonia - dont miss the goals/team news. Txt ur national team to 87077 eg ENGLAND to 87
21 ham  Is that seriously how you spell his name?
```

Data

Database : MongoDB
- Collection : msan698
- Database : sms

```
$ mongod
On a new terminal…
$ mongo
>  use msan698
```

```
> db.sms.find()
{ "_id" : ObjectId("5aa9896e5c50bf0be613d330"), "Label" : "ham", "Message" : "Go until jurong point, crazy.. Available only in bugis n g
reat world la e buffet... Cine there got amore wat..." }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d331"), "Label" : "spam", "Message" : "WINNER!! As a valued network customer you have been selec
ted to receivea £900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hours only." }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d332"), "Label" : "spam", "Message" : "Had your mobile 11 months or more? U R entitled to Update
 to the latest colour mobiles with camera for Free! Call The Mobile Update Co FREE on 08002986030" }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d333"), "Label" : "ham", "Message" : "I'm gonna be home soon and i don't want to talk about this
 stuff anymore tonight, k? I've cried enough today." }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d334"), "Label" : "ham", "Message" : "As per your request 'Melle Melle (Oru Minnaminunginte Nuru
ngu Vettam)' has been set as your callertune for all Callers. Press *9 to copy your friends Callertune" }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d335"), "Label" : "spam", "Message" : "SIX chances to win CASH! From 100 to 20,000 pounds txt> C
SH11 and send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply Reply HL 4 info" }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d336"), "Label" : "ham", "Message" : "I've been searching for the right words to thank you for t
his breather. I promise i wont take your help for granted and will fulfil my promise. You have been wonderful and a blessing at all time
s." }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d337"), "Label" : "ham", "Message" : "I HAVE A DATE ON SUNDAY WITH WILL!!" }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d338"), "Label" : "spam", "Message" : "XXXMobileMovieClub: To use your credit, click the WAP lin
k in the next txt message or click here>> http://wap. xxxmobilemovieclub.com?n=QJKGIGHJJGCBL" }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d339"), "Label" : "ham", "Message" : "Oh k...i'm watching here:)" }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d33a"), "Label" : "spam", "Message" : "URGENT! You have won a 1 week FREE membership in our £100
,000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C www.dbuk.net LCCLTD POBOX 4403LDNW1A7RW18" }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d33b"), "Label" : "ham", "Message" : "U dun say so early hor... U c already then say..." }
{ "_id" : ObjectId("5aa9896e5c50bf0be613d33c"), "Label" : "ham", "Message" : "Fine if thats the way u feel. Thats the way its gota b"
}
```

See Week1/DataStorage in the github.

Insert Training Data (Data Storage)

```
def insert_data_file(input, database, collection, format, fields):
    """
    Read input data as TSV and insert into the collection.
    """

    mongoimport_query = "mongoimport" + " --db " + database \
                        + " --collection " + collection + " --file " + input \
                        + " --type " + format + " --fields " + fields

    subprocess.call(mongoimport_query, shell=True)
```

Output :

```
> db.sms.count()
5574
```

Testing Data

Data

- Assumption
  - A text message response is sent to a server using services like twilio.
  - Format : Free text.

See Week1/DataStorage in the github.

Insert Data with a Label (Data Storage)

```
def insert_data_entry(input, fields, delimiter, database, collection):
    """
    Read a single line input and insert into collection.
    todo: Raise exception
    """
    db_conn = create_connection(database)

    if(fields.count(delimiter) != input.count(delimiter)):
        print("Error : The number of delimeters in the input \
               and fields are different.")
        return

    splitted_fields = fields.split(delimiter)
    splitted_input = input.split(delimiter)

    data = {}
    for count in range(0, fields.count(delimiter)+1):
        data[splitted_fields[count]] = splitted_input[count]

    db_conn[collection].insert_one(data)
```

Output :

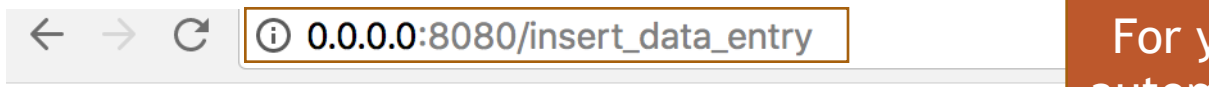
```
> db.sms.count()
5575
```

```
> db.sms.find()
{ "_id" : ObjectId("5aa9a40a58a8860bc68b4722"), "Label" : "ham", "Message" : "How are you?" }
```

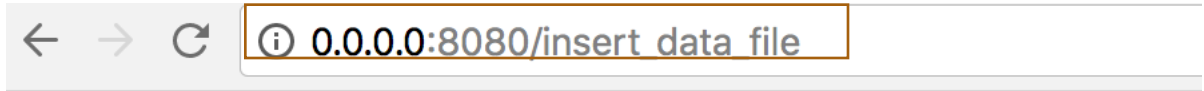
Launch as a flask application

See the slides from MSAN 603 technical sessions.

```
ML-ITS-603436:smsspamcollection dwoodbridge$ python application.py
 * Running on http://0.0.0.0:8080/ (Press CTRL+C to quit)
 * Restarting with stat
 * Debugger is active!
```



Data Entry Inserted - Total 1



Data File Inserted - Total 5575

See Week1/DataStorage in the github. For your project, make sure have an automated data acquisition code, too.

Make it faster!

Indexing
- Optimize query performance.
- B-Tree is default in MongoDB.
  - Default index : _id (Cannot be deleted)
- getIndexes()
  - Check which indexes exist on a collection.
- createIndex({“field” : direction})
  - Direction : 1 (Ascending), -1 (Descending)
- dropIndex({“field” : direction})

```
$ mongod
On a new terminal…
$ mongo
>  use msan698
```

```
> db.sms.createIndex({"Label":1})
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 1,
        "numIndexesAfter" : 2,
        "ok" : 1
}
```

* This will make it fast when you query by “Label”

https://docs.mongodb.com/v3.0/tutorial/modify-an-index/
https://docs.mongodb.com/v3.2/tutorial/analyze-query-

Reference

Kenneth Reitz. *The Hitchhiker's Guide to Python*. http://docs.python-guide.org/en/latest/

Grinberg, Miguel. *Flask web development: developing web applications with python*. " O'Reilly Media, Inc.", 2018.