

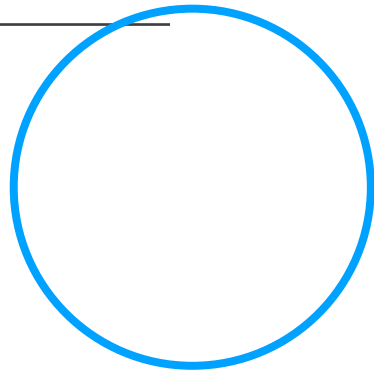
CSE 416 Section 1!

~~Zoom University - Pandemic Special Quarter 2~~

(Can't believe we made it to summer but when can we go outside?????)


JUNE 25, 2020

• HONGJUN JACK WU 😊



This material is made with color blind folks in mind.

If there is anything that is not clear or you cannot distinguish **PLEASE** let us know so we can fix it ASAP.



Goal for today!

MAIN GOAL:

INTRO TO PYTHON & NOTEBOOK & PANDAS

Materials of the Day

There are notebooks available for you to use.

We'll post the notebooks to the website sometime after we are done with all sections. (Also #24 in this presentation)

Pretty much all the notebook we are using from now on will be available to you. No worries!

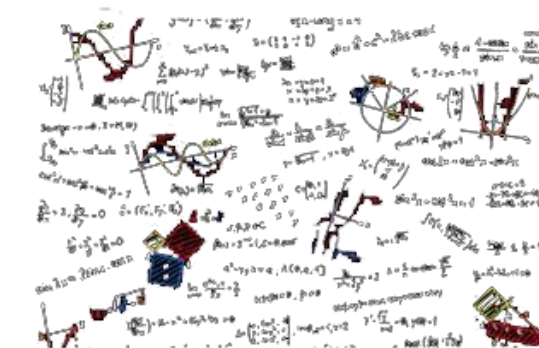
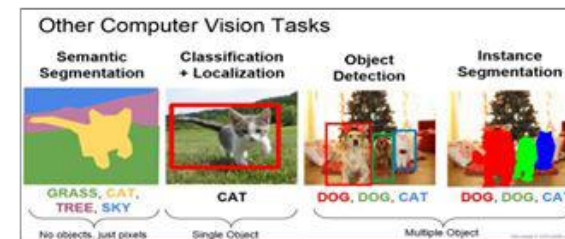
Just sit back, relax, and enjoy!



Beginning
of the
course

End
of the
course

After
becoming
an expert



**Hmmmm...
this makes sense!**



Python.

(OPTIONAL PART I)

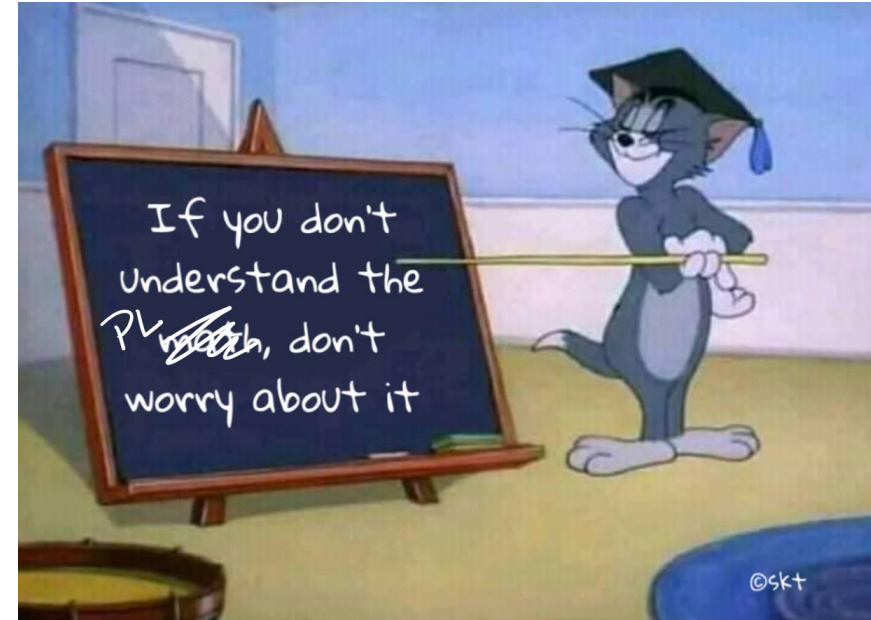
JUST FYI...

Things we are gonna talk about in this part is just extra stuff...

But I think it is necessary to talk about them when we introduce a new programming language.

No pressure! We will **NOT** test you on this.

(Well, it's just stuff I made up that is not part of the 416 curriculum...more like 341)



PROGRAMMING LANGUAGE PERSPECTIVE

When talking about a new programming language (PL):

The “**Semantics**” of the language is more important than the “**Syntax**” of the language.
(Dan Grossman, aka the PL God in CSE 🙌)

Semantics: “What primarily define a PL and its pros/cons” (Brett Wortzman)

Syntax: “How you write something.”

In other words, we try not to talk too much about the syntax differences.
(The `{}` and `;` in Java, `System.out.println();` vs. `print()`, etc)

Focus what fundamentally makes two programming languages different.

INTERPRETED VS. COMPILED: SEMANTICS

Interpreted (Python)

Python is an **"interpreted" language**. This means it uses an **interpreter**. An interpreter is very different from the compiler.

An interpreter executes the statements of code "line-by-line".

```
python HelloWorld.py
```

Compiled (Java, C++, etc)

The **compiler** executes the code entirely and lists all possible errors at a time.

You need to compile human code into machine code before you execute code.

```
javac HelloWorld.java
```

```
java HelloWorld
```

PYTHON SEMANTICS I

Python is an **interpreted, high-level, general-purpose** programming language.

Interpreted:

Uses an interpreter, not a compiler. (Compiled PL: Java, C, C++, SML).

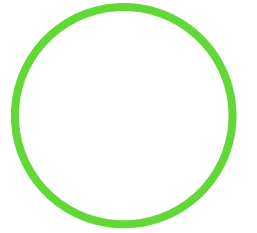
High-Level:

Abstract, user friendly, write code using human logic. (Low level PL: Assembly)

General Purpose:

Can do many things. Machine learning, web scraper, games, etc.

PYTHON SEMANTICS II



Python is **dynamically typed** and **garbage-collected**.

Dynamically Typed:

Provide mutation, no type checking restrictions, no need to declare type. (Python: `a = 1`)

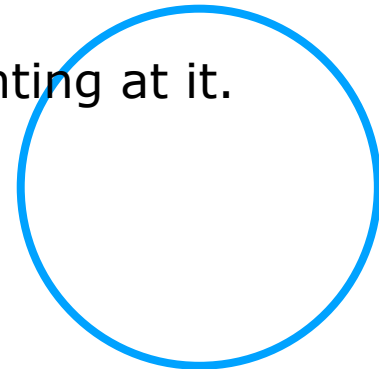
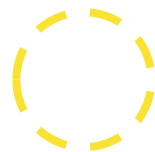
Statically Typed (In Contrary):

You need to declare the type of the variable when you declare it. (Java: `int a = 1;`)

Garbage Collected:

Provide auto garbage collection.

Like in Java, automatically get rid of a linked list node when nothing is pointing at it.



PYTHON SEMANTICS III

Python supports multiple programming paradigms, including **structured**, **object-oriented**, and **functional** programming.

Structured:

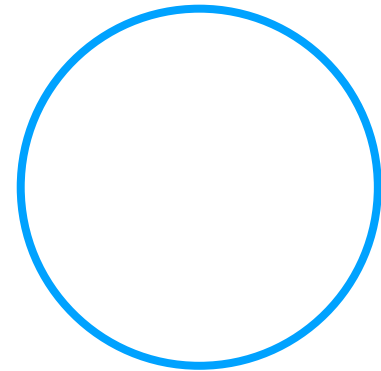
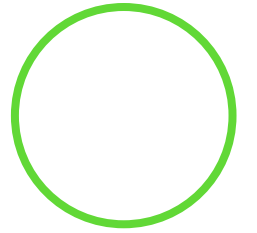
Use if/then/else, for/while, block structures (aka sub functions).

Object Oriented Programming:

Treat elements like objects, use fields, constructors, etc. (Java!)

Functional Programming:

Programs are constructed using functions, (Example: SML, Racket).





RANDOM STUFF ABOUT HW0

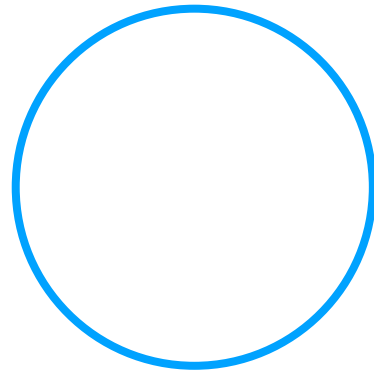
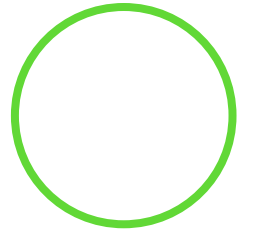
Python is very straightforward and easy to understand.

In theory even if you don't have any python experience as long as you have some Java experience and just google "python for loop", "python toString", "python list" the entire quarter you can still succeed in 416 😊.

To make your life easier we made the intro to python and intro to pandas (aka HW0) so as long as you do it it'll save you a lot of time googling.

So do it!!!! It'll help you a lot in future assignments.

Ummmm yeah! 416 is fun and chill, don't stress out and we promise you'll walk out with something useful to apply to whatever happen in the future.





CoLaboratory Notebook.

(PART II)

PYTHON AS AN INTERPRETED LANGUAGE

Big Takeaway:

An “Interpreter” runs code one by one, and no need to re-compile the entire thing!

Why is that important?

- You can test a small snippet of code without re-run the entire thing.
- That means all other variables and loaded data will still be in the memory.
- Imagine you have a HUGE dataset, takes an hour to load, and you realized made a typo in your code after you press the run button.
- In compiled languages, it’s gonna be a nightmare!!!!

Some terms:

Markdown: Neat way to format text. Looks great.

Notebook (Generally): Python code snippets + Markdown for explanation.

COLABORATORY NOTEBOOK

For the past couple quarters we've always given students options to use either a local Jupyter Notebook or [Google Colaboratory Notebook](#) (A Jupyter Notebook hosted by Google on the cloud with free GPU support).

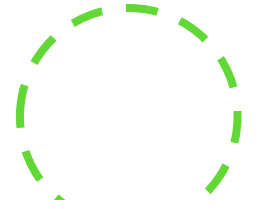
However, it's tOo mUch tROuBle for everyone to setup an environment and install all the required packages with the correct versions. **We'll use the Google Colaboratory as the official notebook for Summer 2020.**

You are welcome to setup an Python environment on your computer and run your personal projects. **We will just treat everyone the same as if they did all their homework using Google Colaboratory.**





COLAB FOR NOOBS – MAIN INTERFACE



Welcome To Colaboratory

Title (Click to change)

File Edit View Insert Runtime Tools Help

Option Bar



Cute Animals!

Share



H

Add Cell

+ Code + Text

Copy to Drive

Make a copy in your Google Drive

RAM Disk

Editing

Table of Contents

Code Snippets

Data science

Files

More Resources

Machine Learning Examples

Section

Cell Actions



Getting started

Markdown Cell (Good looking text, not code)

The document you are reading is not a static web page, but an interactive environment called a **Colab notebook** that lets you write and execute code.

For example, here is a **code cell** with a short Python script that computes a value, stores it in a variable, and prints the result:

Run Sequence

```
[2] seconds_in_a_day = 24 * 60 * 60
seconds_in_a_day
```

Cell (Python Code Snippet)

86400

Output

To execute the code in the above cell, select it with a click and then either press the play button to the left of the code, or use the keyboard shortcut "Command/Ctrl+Enter". To edit the code, just click the cell and start editing.

Variables that you define in one cell can later be used in other cells:

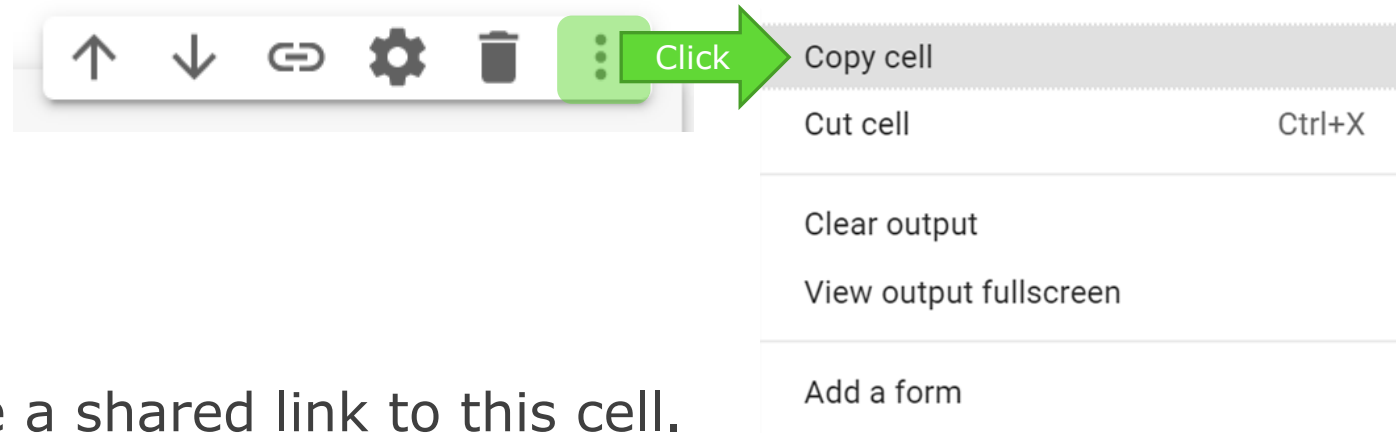
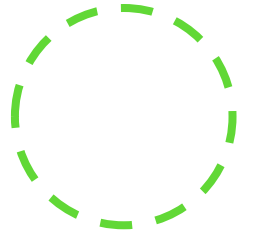
Unrun Cell, No Sequence

```
[ ] seconds_in_a_week = 7 * seconds_in_a_day
seconds_in_a_week
```

604800

Saved output from a previous run

CELL ACTIONS INTERFACE



↑ : Move Cell Up

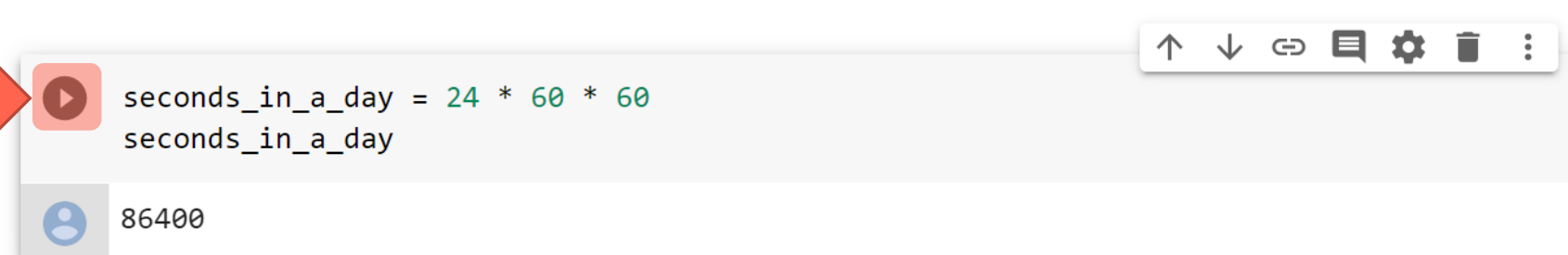
↓ : Move Cell Down

🔗 : Link to Cell - Create a shared link to this cell.

⚙️ : Colaboratory Editor Settings

🗑️ : Delete this Cell

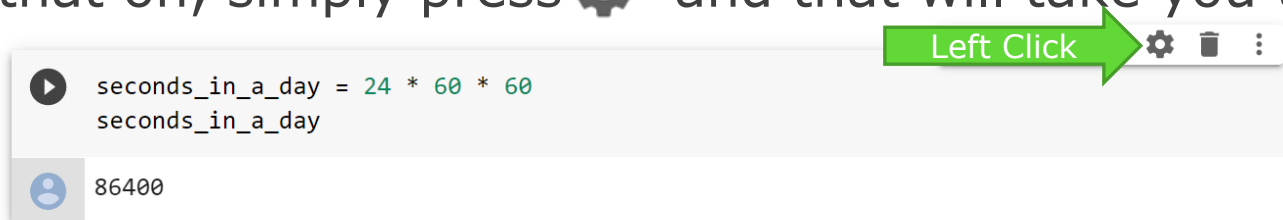
Click this button to **Run Cell**
Ctrl(Command) + Enter or Shift + Enter



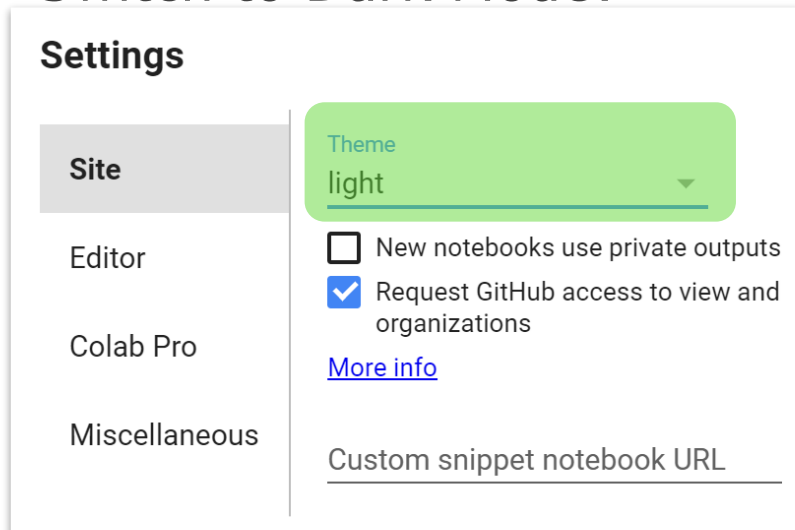
BUT I'M A CAT PERSON!!!!!!

Colab has very cute kitty mode and corgi mode, as well as dark mode.

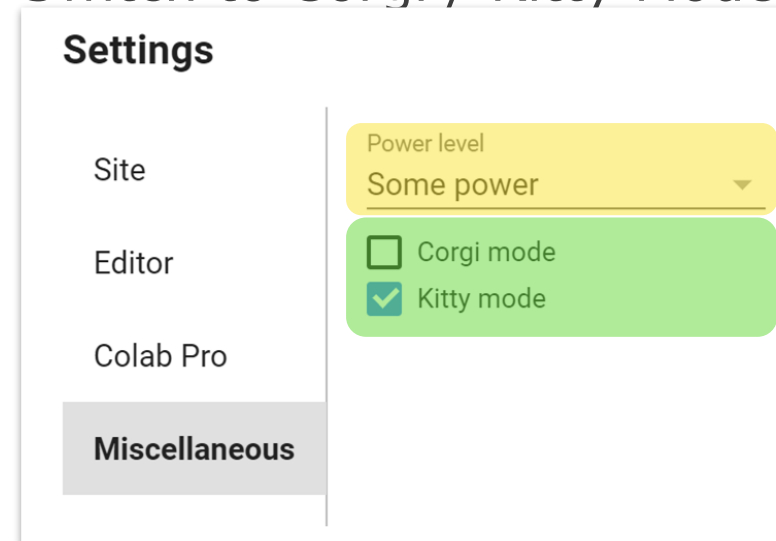
To turn that on, simply press  and that will take you to settings.



Switch to Dark Mode:



Switch to Corgi / Kitty Mode:



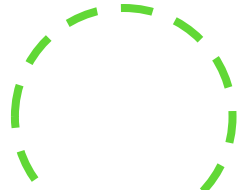
We'll leave power level a fun thing for you to explore.



MAKE A COPY TO GOOGLE DRIVE

Welcome To Colaboratory

File Edit View Insert Runtime Tools Help Cannot save changes



Share [Settings] [Profile]



+ Code + Text **Copy to Drive**

Make a copy in your Google Drive

RAM Disk [Progress] Editing [Up Arrow]

Table of contents



Copy of Welcome To Colaboratory

New Window pops out, successfully copied to your Google Drive!

File Edit View Insert Runtime Tools Help Last saved at 10:39 PM

Comment Share [Settings] [Profile]
RAM Disk [Progress] Editing [Up Arrow]

Table of contents



Drive

Search in Drive

My Drive > Colab Notebooks

File saved under /My Drive/Colab Notebooks.

Name ↑

.ipynb_checkpoints

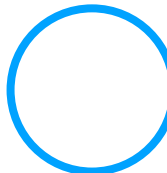
Copy of Welcome To Colaboratory

New

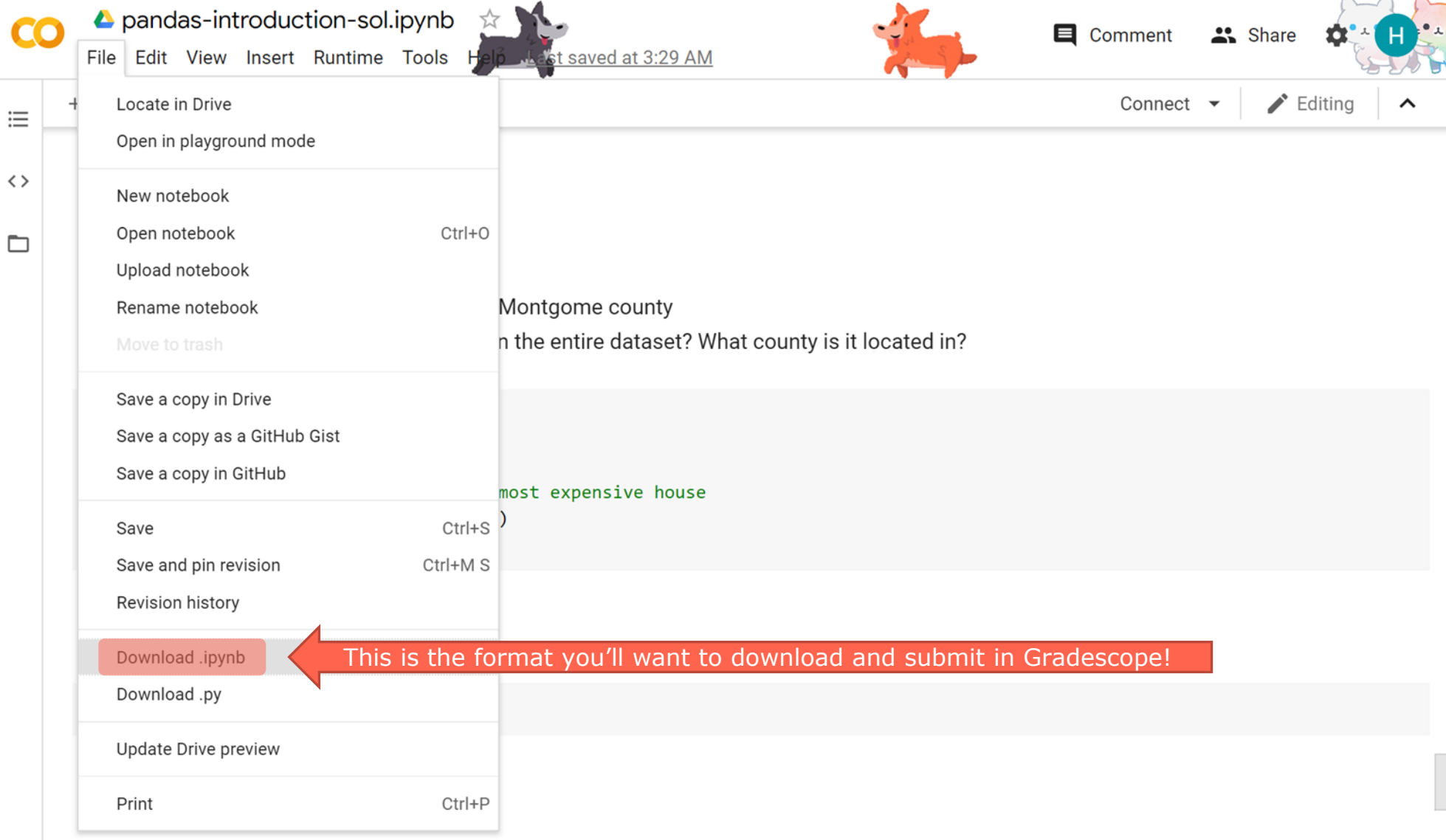
Priority

My Drive

Shared drives



DOWNLOADING FILE TO SUBMIT



The screenshot shows a Jupyter Notebook interface. At the top, the notebook title is "pandas-introduction-sol.ipynb" with a star icon and a small dog avatar. The menu bar includes File, Edit, View, Insert, Runtime, Tools, and Help. The "File" menu is open, showing options like "Locate in Drive", "Open in playground mode", "New notebook", "Open notebook", "Upload notebook", "Rename notebook", "Move to trash", "Save a copy in Drive", "Save a copy as a GitHub Gist", "Save a copy in GitHub", "Save", "Save and pin revision", "Revision history", "Download .ipynb", "Download .py", "Update Drive preview", and "Print". A red arrow points to the "Download .ipynb" option with the text "This is the format you'll want to download and submit in Gradescope!". The notebook content shows a question about Montgomery county and a code cell with the text "most expensive house".

File Edit View Insert Runtime Tools Help Not saved at 3:29 AM

Locate in Drive
Open in playground mode

New notebook
Open notebook Ctrl+O
Upload notebook
Rename notebook
Move to trash

Save a copy in Drive
Save a copy as a GitHub Gist
Save a copy in GitHub

Save Ctrl+S
Save and pin revision Ctrl+M S
Revision history

Download .ipynb
Download .py

Update Drive preview
Print Ctrl+P

Montgome county
n the entire dataset? What county is it located in?

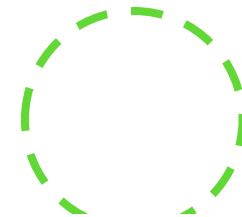
most expensive house

Comment Share

Connect Editing



MOUNT GOOGLE DRIVE IN COLAB



Copy of Welcome To Colaboratory ☆



Comment Share Settings Profile

File Edit View Insert Runtime Tools Help

RAM Disk Editing

Files

- Mount Drive
- sample_data

Hit Mount Drive

What is Colaboratory?

Colaboratory, or "Colab" for short, allows you to write and execute Python in your browser, with

Run this cell to mount your Google Drive.

DISMISS

This cell appears! Run it! Follow the prompt (log in, copy + paste code, hit enter).

Files

- drive
 - My Drive
 - Shared drives
- sample_data

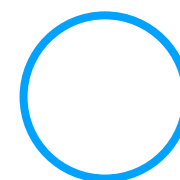
Success!!

```
from google.colab import drive
drive.mount('/content/drive')
```

Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=

Enter your authorization code:
.....

Mounted at /content/drive



OTHER HELPFUL STUFF IN COLAB



pandas-introduction-sol.ipynb ☆

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Add a Code cell or a Text (Markdown) cell below the current selected cell.

```
import requests

def save_file(url):
    r = requests.get(url)
    with open('file.txt', 'w') as f:
        f.write(r.text)

save_file('https://www.philadelphia.edu.sa/')
```

Run all

When you have too many cells and you just want to run all of them, use "Run All".

Run before

Ctrl+F8

Run the focused cell

Ctrl+Enter

Run selection

Ctrl+Shift+Enter

Run after

Ctrl+F10

Interrupt execution

When your code is stuck (ex. Infinite Loop) then use "Interrupt Execution".

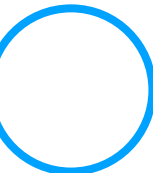
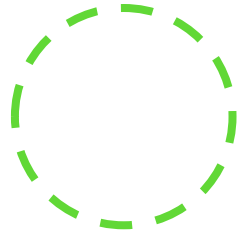
Restart runtime

Ctrl+M .

Restart and run all

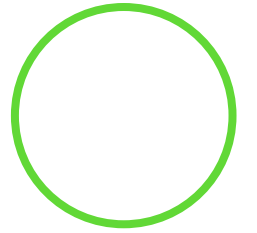
Factory reset runtime

<https://cse416/20su/files/s>





LEARN MARKDOWN IN FIVE MINUTES



Markdown:

A super easy way to make text look nicer. [Markdown Guide](#) website is very useful.

Without making stuff too complicated, the minimum amount to get you started:

Headings: Prepend “# ” in front of your heading, the **FEWER** “#” you have the bigger the heading, and you need a space between that and the actual heading to make it work.

Paragraphs: Just use a blank line to separate one or more lines of text.

Bold: ****Text you want it bold****

Italic: **Text you want it italic**

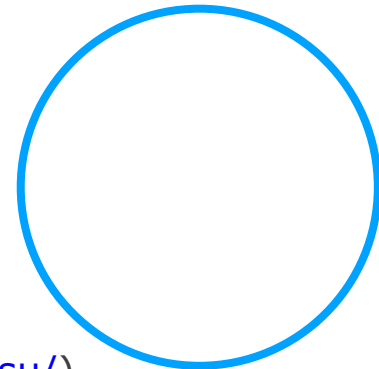
Blockquote: >Text you want to be blockquote

Code: `your code here`

Ordered Lists: Prepend the number you want, like “1. ”, “2. ”, “3. ” in front of stuff you want to list.

Unordered Lists: Just prepend “* ”. (“+ ”, “- ” will work too)

Links: [Make sure to like and subscribe to CSE416](<https://courses.cs.washington.edu/courses/cse416/20su/>)





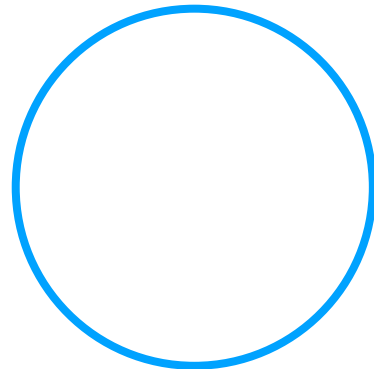
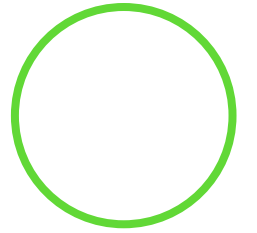
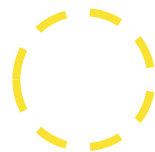
Python & Pandas.

(PART III)

 NOTEBOOKS WE WILL USE:

[Introduction to Pandas \(Blank\)](#)

[Introduction to Pandas \(Solution\)](#)





WORKING WITH PANDAS IN ONE PAGE



Import package: `import package as nickname` (too lazy to write full name)

Import CSV (aka dataset): `dataset = nickname.read_csv()`

Look at the first couple data in the set: `dataset.head()`

Specific column: `dataset['Column Name']`

◦ (add `.min()` `.max()` `.mean()` to calculate whatever you want to calculate)

Index a specific cell in dataset: `dataset['Column Name'].iloc[index]`

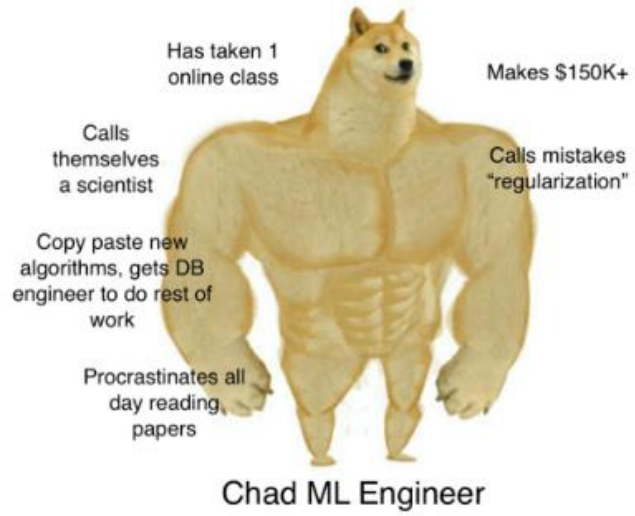
Filter: `dataset[boolean condition]`

How many rows: `len(dataset)`

How many columns: `len(dataset.columns)`

Names of columns: `dataset.columns`





Other Stuff

(PART IV)



INSTALL PYTHON LOCALLY (JUST FYI)




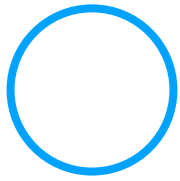
This slide exists just for your information, you don't need this.

More info check out [Spring 2020 course website](#)

[Video I recorded last year](#) on installing (Python 3.6) might be helpful if you can't figure out how to install.

How to install a package if missing:

```
conda install whatever_you_want_to_install
```

- Windows: Run **Anaconda Prompt** / macOS and Linux: **Terminal**
 - If doesn't work, try `pip install whatever_you_want_to_install`
 - Colab: I believe you can just type `pip install whatever_you_want_to_install` into one of the cells and it would install.
- 
- 

MEMES

Tbh the most fun thing (at least for me) after taking 416 is you start to understand memes about machine learning...

Here's my source of memes lol as the quarter goes you'll understand these memes more and more!

<https://www.facebook.com/groups/1638417209555402>

(AI Memes) AI & Deep Learning Memes For Back-propagated Poets

Public group · 76.2K members

when they mention AI in a movie





CREDITS

1. [Syntax and Semantics, Slide #4](#)
2. [Kaggle, Learn Python](#)

me attending
CSE416 then vs now



©skt



LICENSE

This material is originally made by [Hongjun Wu](#) for the course [CSE416: Introduction to Machine Learning](#) in the Summer 2020 quarter taught by [Vinitra Swamy](#), at University of Washington Paul G. Allen School of Computer Science and Engineering.

It was originally made for educational purpose, in a section taught by teaching assistants to help students explore material in more depth.

Any other materials used are cited in the Credits section.

This material is licensed under the [Creative Commons License](#).

Anyone, especially other educators and students, are welcomed and strongly encouraged to study and use this material.