



Introduction to Python Notebooks

Data Programming

*If you were a fruit, which
would you be?*

Icebreaker



Questions?



Announcements

- **Written Check-in #8 due Friday Aug 14!**
- **Programming Practice #7 due Wed, Aug 19!**
- **HW5 due Mon Aug 17 (**No Resubmissions**)**
 - Submitting on Gradescope
 - Wait for autograder!
- **2 more resubmission cycles! (*see ed for details*)**
- **Final Exam on Friday, August 21 at 9:40 am**
 - Past final exams posted on [final exam page](#) (includes only in-scope topics)

Today's Roadmap

1. Python in the Wild
2. Notebooks
3. Working with Data



*data programmer
in the wild*

Python in the Wild

```
1 #!/usr/bin/env python3.9
2
3 import sys
4 import json
5 import textwrap
6
7 prefix = " "
8 preferredWidth = 150
9
10 if len(sys.argv) != 2:
11     print("Please specify a single file as an argument")
12     sys.exit()
13
14 with open(sys.argv[1]) as json_file:
15     results = json.load(json_file)
16     tests = results['tests']
17
18     wrapper = textwrap.TextWrapper(initial_indent=prefix,
19                                   replace_whitespace=False,
20                                   subsequent_indent=' ' * len(prefix))
21
22     print('Total Score:', results['score'], '\n')
23     for test in tests:
24         print(test['name'], '-', str(test['score']) + '/' +
25               str(test['max_score']))
26         if 'output' in test:
27             for line in test['output'].splitlines():
28                 print(wrapper.fill(line))
29     print()
```

.py file

*What if we need to
work with a lot of data?*



notebooks.ipynb

Introduction to Notebooks

CSE 160

August 14, 2026

```
[1]: import pandas as pd
```

```
[3]: data = pd.read_csv("lego_sets.csv")
```

data

```
[3]:
```

	set_id	name	year	theme	subtheme	themeGroup	category	pieces	n
0	1-8	Small house set	1970	Minitalia	NaN	Vintage	Normal	67.0	
1	2-8	Medium house set	1970	Minitalia	NaN	Vintage	Normal	109.0	
2	3-6	Medium house set	1970	Minitalia	NaN	Vintage	Normal	158.0	
3	4-4	Large house set	1970	Minitalia	NaN	Vintage	Normal	233.0	
4	4-6	Mini House and Vehicles	1970	Samsonite	Model Maker	Vintage	Normal	NaN	
...	...	...	...	...	...	...	...	...	
18452	SHUTTLE-1	Space Shuttle	2022	Promotional	LEGO brand stencils	Miscellaneous	Other	NaN	

.ipynb notebook

Python Notebooks

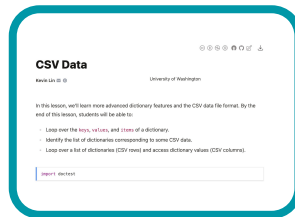
A Python notebook a “web-based application suitable for **capturing the whole computation process**: developing, documenting, and executing code, as well as communicating the results” ([IPython Docs](#))



Code Editor



Text Document



Webpage

Out: `.py` 🤔 → In: `.ipynb` 😊



Jupyter Notebook

Python Notebook Tips

Code Cells

Contain Python code

- Run a code cell (SHIFT + ENTER or ►) to execute just the code in the cell

`[ ]:` `x = "batman"` → `[54]:` `x = "batman"`
cell not run yet cell run (54th cell to be run)

- Variables, functions are **shared** across entire notebook
⇒ *Order in which cells are run matter*
- Expression on last line is evaluated and outputted below cell:

```
[55]: 1 + 2
      "Lego " + x
[55]: 'Lego batman'
```

- Some Shortcuts:**
 - a** insert a code cell above
 - b** insert a code cell below



Markdown Cells

Contain text, images, markdown

- Convert a code cell (default) to markdown by pressing M or selecting "Markdown" from the dropdown menu
- When rendered (SHIFT + ENTER or ►), markdown cells display formatted text:

```
## This is a markdown cell
```

This is a markdown cell

- Use the [markdown language](#) (or HTML tags) to format text

Kernel

Runs the code cells, manages variables

- Need an active kernel to run code cells: Python 3 (ipykernel) ○
- If kernel crashes (or becomes inactive), restart kernel by clicking "Kernel" > "Restart Kernel..."



Questions?



Live and on Sli.do

Sli.do

#cse160

Our Dataset

lego_sets.csv

Lego Set

Each **incident** (row) has:

- **set_id**: Official LEGO item number
- **name**: Name of the LEGO set
- **year**: Release year
- **theme**: LEGO theme the set belongs to
- **subtheme**: Subtheme within the theme
- **themeGroup**: Overall group the theme belongs to
- **category**: Type of set
- **pieces**: Number of pieces in the set
- **minifigs**: Number of mini figures included in the set
- **agerange_min**: Minimum age recommended
- **US_retailPrice**: US retail price at launch
- **bricksetURL**: URL for the set on brickset.com
- **thumbnailURL**: Small image of the set
- **imageURL**: Full size image of the set

	set_id	name	year	theme	subtheme	themeGroup	category	pieces	minifigs	agerange_min	US_retailPrice	bricksetURL	thumbnailURL
16131	71708-1	Gamer's Market	2020	Ninjago	Prime Empire	ion/Adventure	Normal	218	9	7	29.99	sets/71708-1	all/71708-1.jpg
16132	71709-1	It's Velocity Racers	2020	Ninjago	Prime Empire	ion/Adventure	Normal	322	3	7	29.99	sets/71709-1	all/71709-1.jpg
16133	71710-1	Ninja Tuner Car	2020	Ninjago	Prime Empire	ion/Adventure	Normal	419	4	8	39.99	sets/71710-1	all/71710-1.jpg
16134	71711-1	It's Cyber Dragon	2020	Ninjago	Prime Empire	ion/Adventure	Normal	518	5	8	49.99	sets/71711-1	all/71711-1.jpg
16135	71712-1	Temple of Madness	2020	Ninjago	Prime Empire	ion/Adventure	Normal	810	6	9	79.99	sets/71712-1	all/71712-1.jpg
16136	71713-1	Empire Dragon	2020	Ninjago	Prime Empire	ion/Adventure	Normal	286	3	8	29.99	sets/71713-1	all/71713-1.jpg
16137	71714-1	It's Arcade Pod	2020	Ninjago	Arcade Pod	ion/Adventure	Normal	49	2	7	9.99	sets/71714-1	all/71714-1.jpg
16138	71715-1	It's Arcade Pod	2020	Ninjago	Arcade Pod	ion/Adventure	Normal	47	2	7	9.99	sets/71715-1	all/71715-1.jpg
16139	71716-1	It's Arcade Pod	2020	Ninjago	Arcade Pod	ion/Adventure	Normal	48	2	7	9.99	sets/71716-1	all/71716-1.jpg
16140	71717-1	It's Skull Dungeons	2020	Ninjago	It's the Mountain	ion/Adventure	Normal	401	4	7	29.99	sets/71717-1	all/71717-1.jpg
16141	71718-1	It's Battle Dragon	2020	Ninjago	It's the Mountain	ion/Adventure	Normal	321	2	7	19.99	sets/71718-1	all/71718-1.jpg
16142	71719-1	It's Mino Creature	2020	Ninjago	It's the Mountain	ion/Adventure	Normal	616	4	8	49.99	sets/71719-1	all/71719-1.jpg
16143	71720-1	It's Fire Stone Mech	2020	Ninjago	It's the Mountain	ion/Adventure	Normal	968	5	9	69.99	sets/71720-1	all/71720-1.jpg
16144	71721-1	Sorcerer's Dragon	2020	Ninjago	It's the Mountain	ion/Adventure	Normal	1016	7	9	79.99	sets/71721-1	all/71721-1.jpg
16145	71722-1	Sorcerer's Dungeons	2020	Ninjago	It's the Mountain	ion/Adventure	Normal	1171	8	9	99.99	sets/71722-1	all/71722-1.jpg
16146	75263-1	It's Wing Microfighter	2020	Star Wars	Microfighters	Licensed	Normal	86	1	6	9.99	sets/75263-1	all/75263-1.jpg
16147	75264-1	It's Hutt Microfighter	2020	Star Wars	Microfighters	Licensed	Normal	72	1	6	9.99	sets/75264-1	all/75264-1.jpg
16148	75265-1	It's Microfighters	2020	Star Wars	Microfighters	Licensed	Normal	198	2	6	19.99	sets/75265-1	all/75265-1.jpg
16149	75266-1	It's Battle Pack	2020	Star Wars	It's of Skywalker	Licensed	Normal	105	4	6	14.99	sets/75266-1	all/75266-1.jpg
16150	75267-1	It's Mandalorian Battle Pack	2020	Star Wars	It's Mandalorian	Licensed	Normal	102	4	6	14.99	sets/75267-1	all/75267-1.jpg
16151	75268-1	It's Snowspeeder	2020	Star Wars	Episode V	Licensed	Normal	91	2	4	19.99	sets/75268-1	all/75268-1.jpg
16152	75269-1	Duel on Mustafar	2020	Star Wars	Episode III	Licensed	Normal	208	2	7	19.99	sets/75269-1	all/75269-1.jpg
16153	75270-1	Obi-Wan's Hut	2020	Star Wars	Episode IV	Licensed	Normal	200	4	7	29.99	sets/75270-1	all/75270-1.jpg
16154	75271-1	It's Landspeeder	2020	Star Wars	Episode IV	Licensed	Normal	236	3	7	29.99	sets/75271-1	all/75271-1.jpg
16155	75272-1	Sith TIE Fighter	2020	Star Wars	It's of Skywalker	Licensed	Normal	470	3	9	79.99	sets/75272-1	all/75272-1.jpg

Think Pair Share

What are some questions we might want to answer with this data?



name	year	theme	pieces	minifigs	agerange_ min	US_retailP rice
'Snowspeeder'	'2020'	'Star Wars'	'91'	'2'	'4'	'19.99'



Think Pair Share

Write a function called **get_sets_by_year** that takes the list of dictionaries from **DictReader** and returns:

- A single **dictionary** where each **year is a key** and the corresponding value is the **number of sets produced that year** in the data

```
[{'\uffeffset_id': '1-8',
  'name': 'Small house set',
  'year': '1970',
  'theme': 'Minitalia',
  'subtheme': '',
  'themeGroup': 'Vintage',
  'category': 'Normal',
  'pieces': '67',
  'minifigs': '',
  'agerange_min': '',
  'US_retailPrice': '',
  'bricksetURL': 'https://brickset.com/sets/1-8',
  'thumbnailURL': 'https://images.brickset.com/sets/small/1-8.jpg',
  'imageURL': 'https://images.brickset.com/sets/images/1-8.jpg'},
 {'\uffeffset_id': '2-8',
  'name': 'Medium house set',
  'year': '1970',
  'theme': 'Minitalia',
  'subtheme': '',
  'themeGroup': 'Vintage',
  'category': 'Normal',
  'pieces': '109',
  'minifigs': '',
  'agerange_min': '',
  'US_retailPrice': '',
  'bricksetURL': 'https://brickset.com/sets/2-8',
  'thumbnailURL': 'https://images.brickset.com/sets/small/2-8.jpg',
  'imageURL': 'https://images.brickset.com/sets/images/2-8.jpg'},
 ...]
```

Notebook Philosophy

or, *How to make the most of your notebook*

Take Small Steps

Avoid large code cells.

- Defining a function? *Give it a cell!*
- Loading a dataset? *Give it a cell!*
- Debugging? *Use a new cell!*

Document!

A good Python notebook combines code and text to **explain each data processing step** and make reproducing results easy!



Questions?

Live and on Sli.do

Sli.do

#cse160