

# CSCI 1070: Taming Big Data

Dr. Abby Stylianou

# Logistics

- Due to Blackboard issues, we will be using Moodle instead
- Assignment 0 is out today. We will hopefully work on it in class and discuss how to turn it in. It's due next Tuesday before class.
- Late assignment policy:
  - Assignments are due at 10:59am on their due date.
  - From 11:00am that day to 10:59am the following day, 10% off
  - The next day, another 10% off
  - So on and so forth

Start



VS

```
rent/web/thebuggenie $ ./tbg_cli remote:show_issue thebuggenie
tailed information about thebuggenie issue #385 on http://thebu
[EN] Add Bug Genie version to general tab of configuration page
EN
Tue, 15 Feb 2011 12:07:44 GMT (1297771684)
nd by: kanto501
Updated: Tue, 15 Feb 2011 12:07:44 GMT (1297771684)
Assigned to: zegenie
Status: In progress
Description: I think it would be nice to see what version of Bug Genie yo
sometimes the engineer upgrades the system overnight and does not tell n

Milestone: -
Category: -
Estimated time: -
Spent time: -
Percent complete: -
Priority: -

Comments:
Comment #1
Posted by: zegenie
Posted: Tue, 15 Feb 2011 12:26:25 GMT (1297772785)
Comment: The issue was updated with the following change(s):
* The status has been updated, from "'New'" to "'In progress'".
* The assignee has been changed, from "'Not assigned'" to "'zegenie'".
* Information about the user working on this issue has been changed, from
-----

Comment #2
Posted by:
Comment:
-----
version number
```

Graphical User Interfaces vs. Command Line Interfaces

# Writing and Running Code

- Old-school pipeline:
  1. Write a text file with python code in it
  2. Save it to `program_name.py`
  3. Go to your command line interface and run:

```
python program_name.py
```

# Writing and Running Code

- Old-school pipeline:
  1. Write a text file with python code in it
  2. Save it to program\_name.py
  3. Go to your command line interface and run:

```
python program_name.py
```

*Let's do that!*

# Let's Create a Folder for Our Class

- Name the folder cs1070
- I don't care where you put it, but I'd make it somewhere easy to get to (such as in your Documents folder)
- Get the "full path" for the folder
  - I'll walk around and help with this if you can't figure it out but it should be something like:
    - /Users/abby/Documentrs/cs1070/ (mac)
    - /home/abby/Documents/cs1070/ (linux)
    - C:\Users\abby\Documents (windows)

# Write our first program:

1. Open a text editor
2. Type the following code:

```
print('Hello world')
```

3. Save the file as `first_program.py` in that folder you just created

# Open the Command Line Interface

- On Mac or Linux machines, open the application called “Terminal”
- On Windows machines, open the application called “Command Prompt”



# Navigate to the Right Folder

- Remember the “full path” for the folder you created?
  - /Users/abby/Documents/cs1070/ (mac)
  - /home/abby/Documents/cs1070/ (linux)
  - C:\Users\abby\Documents (windows)
- In the command line, type in:

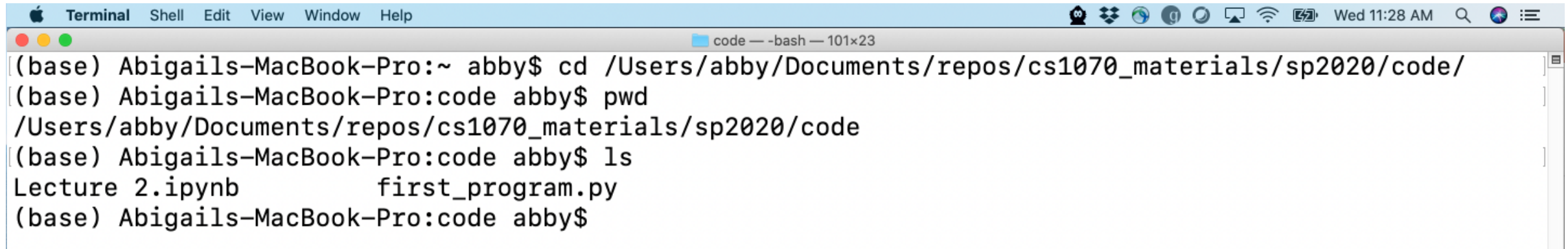
*cd /path/to/your/folder*

replacing /path/to/your/folder with .... the path to your folder

(this command should work for Linux, Mac and Windows)

# A Couple Useful Commands

- `cd` : stands for “change directory”
- list the contents of the current directory
  - Linux/Mac: `ls`
  - Windows: `dir`
- get the full path of the present working directory
  - Linux/Mac: `pwd`
  - Windows: `cd` (*without any directory passed in*)

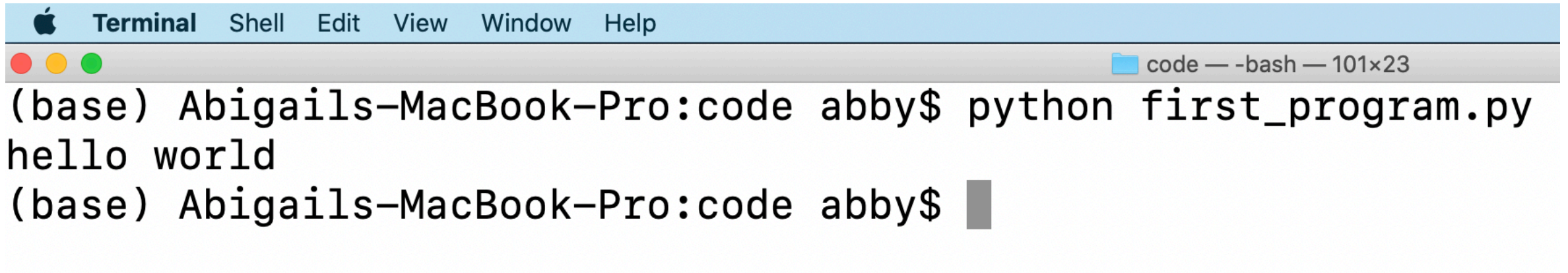


```
Terminal  Shell  Edit  View  Window  Help
code — -bash — 101x23
(base) Abigails-MacBook-Pro:~ abby$ cd /Users/abby/Documents/repos/cs1070_materials/sp2020/code/
(base) Abigails-MacBook-Pro:code abby$ pwd
/Users/abby/Documents/repos/cs1070_materials/sp2020/code
(base) Abigails-MacBook-Pro:code abby$ ls
Lecture 2.ipynb          first_program.py
(base) Abigails-MacBook-Pro:code abby$
```

# Running Your Program

- Now that we've opened the command line and navigated to the correct folder, we're ready to run our program.
- In the command line interface, type:

```
python first_program.py
```

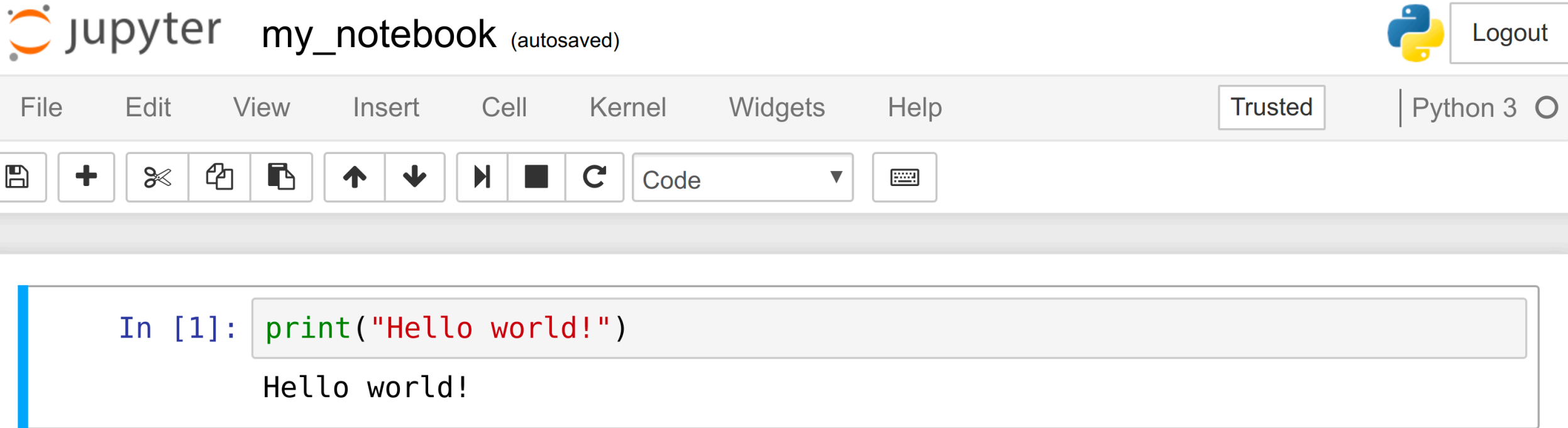


The image shows a macOS Terminal window. The title bar at the top is light blue and contains the Apple logo, the word "Terminal", and menu items: "Shell", "Edit", "View", "Window", and "Help". Below the title bar is a grey bar with three colored window control buttons (red, yellow, green) on the left and a folder icon followed by the text "code — -bash — 101x23" on the right. The main area of the terminal is white and contains the following text: a prompt "(base) Abigails-MacBook-Pro:code abby\$" followed by the command "python first\_program.py", the output "hello world", and a second prompt "(base) Abigails-MacBook-Pro:code abby\$" with a grey cursor block.

```
(base) Abigails-MacBook-Pro:code abby$ python first_program.py
hello world
(base) Abigails-MacBook-Pro:code abby$
```

# Jupyter Notebooks

- Write and run Python code in the browser



The screenshot displays the Jupyter Notebook web interface. At the top left, the Jupyter logo is followed by the text 'my\_notebook (autosaved)'. On the top right, there is a Python logo and a 'Logout' button. Below these elements is a horizontal menu bar with the following items: 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', 'Help', 'Trusted', and 'Python 3'. Underneath the menu bar is a toolbar containing icons for saving, creating a new file, cutting, copying, pasting, undo, redo, and a dropdown menu currently set to 'Code'. The main area of the notebook shows a single code cell. The prompt 'In [1]:' is followed by the Python code `print("Hello world!")` on one line, and the output 'Hello world!' is displayed on the line below. A blue vertical bar is visible on the left side of the code cell.

jupyter my\_notebook (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Code

```
In [1]: print("Hello world!")
Hello world!
```

# Jupyter Notebooks

- Installed on the lab machines already.
- Open Terminal/Command Prompt and run:

`pip install jupyter`

*(if that fails, you might need to run `pip install --user jupyter`)*

- The lab machines already have it installed
- If your computer doesn't have pip (Windows machines especially may not, use a lab machine today and come to office hours to get set up)

# Jupyter Notebooks

- To start a new notebook, first navigate to the folder you create earlier in Terminal (what was the command?)
- Then, run:

```
jupyter notebook
```

```
code — jupyter-notebook — 101x24
(base) Abigails-MacBook-Pro:~ abby$ cd /Users/abby/Documents/repos/cs1070_materials/sp2020/code/
(base) Abigails-MacBook-Pro:code abby$ jupyter notebook
[I 13:20:10.436 NotebookApp] The port 8888 is already in use, trying another port.
[I 13:20:10.487 NotebookApp] JupyterLab extension loaded from /Users/abby/opt/anaconda3/lib/python3.7/site-packages/jupyterlab
[I 13:20:10.487 NotebookApp] JupyterLab application directory is /Users/abby/opt/anaconda3/share/jupyterlab
[I 13:20:10.489 NotebookApp] Serving notebooks from local directory: /Users/abby/Documents/repos/cs1070_materials/sp2020/code
[I 13:20:10.489 NotebookApp] The Jupyter Notebook is running at:
[I 13:20:10.489 NotebookApp] http://localhost:8889/?token=8fdeb7f96ec60777694508b60ca7d2402e3a16b2e66d977b
[I 13:20:10.489 NotebookApp] or http://127.0.0.1:8889/?token=8fdeb7f96ec60777694508b60ca7d2402e3a16b2e66d977b
[I 13:20:10.489 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 13:20:10.496 NotebookApp]
```

To access the notebook, open this file in a browser:

file:///Users/abby/Library/Jupyter/runtime/nbserver-78531-open.html

Or copy and paste one of these URLs:

http://localhost:8889/?token=8fdeb7f96ec60777694508b60ca7d2402e3a16b2e66d977b

or http://127.0.0.1:8889/?token=8fdeb7f96ec60777694508b60ca7d2402e3a16b2e66d977b



Select items to perform actions on them.

Upload

New ▾

<input type="checkbox"/>	0 ▾	/	Name ▾	
<input type="checkbox"/>		Lecture 2.ipynb		1 B
<input type="checkbox"/>		first_program.py		1 B

Notebook:

Python 2

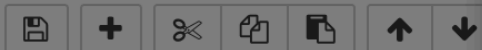
Python 3

Other:

Text File

Folder

Terminal



In [ ]:

## Rename Notebook



Enter a new notebook name:

Cancel

Rename



```
In [ ]: # this line is a comment
        # comments explain what's happening in code
        # you should always comment your code sufficiently so that
        # you can come to back it later and remember what the code does

        # print out 'hello world'
        print('hello world')

        # to run this, hit Ctrl-Enter
```



```
In [1]: # this line is a comment
# comments explain what's happening in code
# you should always comment your code sufficiently so that
# you can come to back it later and remember what the code does

# print out 'hello world'
print('hello world')

# to run this, hit Ctrl-Enter
```

```
hello world
```

# Assignment 0