SECTION 21: GUIs - 45 minutes, 7 parts

4/7 GUI Widget Basics

- -> basic widgets
- -> the interact() and interactive() methods
- -> he imports ipywidgets

- **-> there are multiple options for widgets**
  - -> one of these is IntSlider()
  - -> the output of this can be called and printed in a cell

- -> we can also use the .IntSlider() method

- **-> he imports another module, then uses the display() method**
  - -> if this method is uses multiple times in the ipynb file for the same function, then the slider for all of the uses will change if one of them is
  - -> using the .close() method stops this

- -> the .value method can also be called on a cell
- -> most widgets share keys in addition to values
- -> stateful properties are connected to the state of the current display
- -> under the keys, we have a max value
- -> this can be set to 2,000 for example
- -> the stateful values can also be played around with
- -> there are also other options, depending on which widget is being played around with
- -> he creates a widget using the .FloatText() module
- -> he then creates a second variable, using the .FloatSlider() module
- -> these can be linked using the .display method
- -> he uses .jslink() and enters a string as to what we want to connect into this
  - -> these are being connected as a tuple in this example

- **-> he has**

```
In [111]: import ipywidgets as widgets

In [113]: w = widgets.IntSlider()

In [ ]: from IPython.display import display

In [115]: display(w)
```

79

```
In [116]: display(w)
```

79

```
In [119]: w = widgets.IntSlider()
          display(w)
```

92

```
In [121]: w.value
Out[121]: 92

In [ ]: w.value = 50

In [123]: w.keys
Out[123]: ['_dom_classes',
           '_model_module',
           '_model_module_version',
           '_model_name',
           '_range',
           '_view_module',
           '_view_module_version',
           '_view_name',
           'continuous_update',
           'description',
           'disabled',
           'layout',
           'max',
           'min',
           'msg_throttle',

In [124]: w.max = 2000

In [127]: a = widgets.FloatText()
          b = widgets.FloatSlider()
          display(a,b)
```

10.0

23.60

```
In [126]: display(a)

In [129]: a = widgets.FloatText()
          b = widgets.FloatSlider()


          display(a,b)

          mylink = widgets.jslink((a,'value'),(b,'max'))
```

1000

0.00

```
In [130]: mylink.unlink()
```

    - -> created two widgets
    - -> then displayed them
    - -> then linked them
    - -> two or more can be called
    - -> then .jslink is called and tuples are passed in as its arguments
    - -> the maximum of the second value can also be printed
    - -> the slider can go up to 100 here
- -> these widgets can then be unlinked using the .unlink() method

- **-> overview**
    - -> calling display() on a widget
    - -> these have keys which allow us to see their stateful properties
    - -> the property which shows up when we see the widget
        - -> the state property which we can edit or change
        - -> widgets can be linked together using .jslink()
        - -> then the widget can be passed in, along with what we want to link
        - -> this was widgets basics