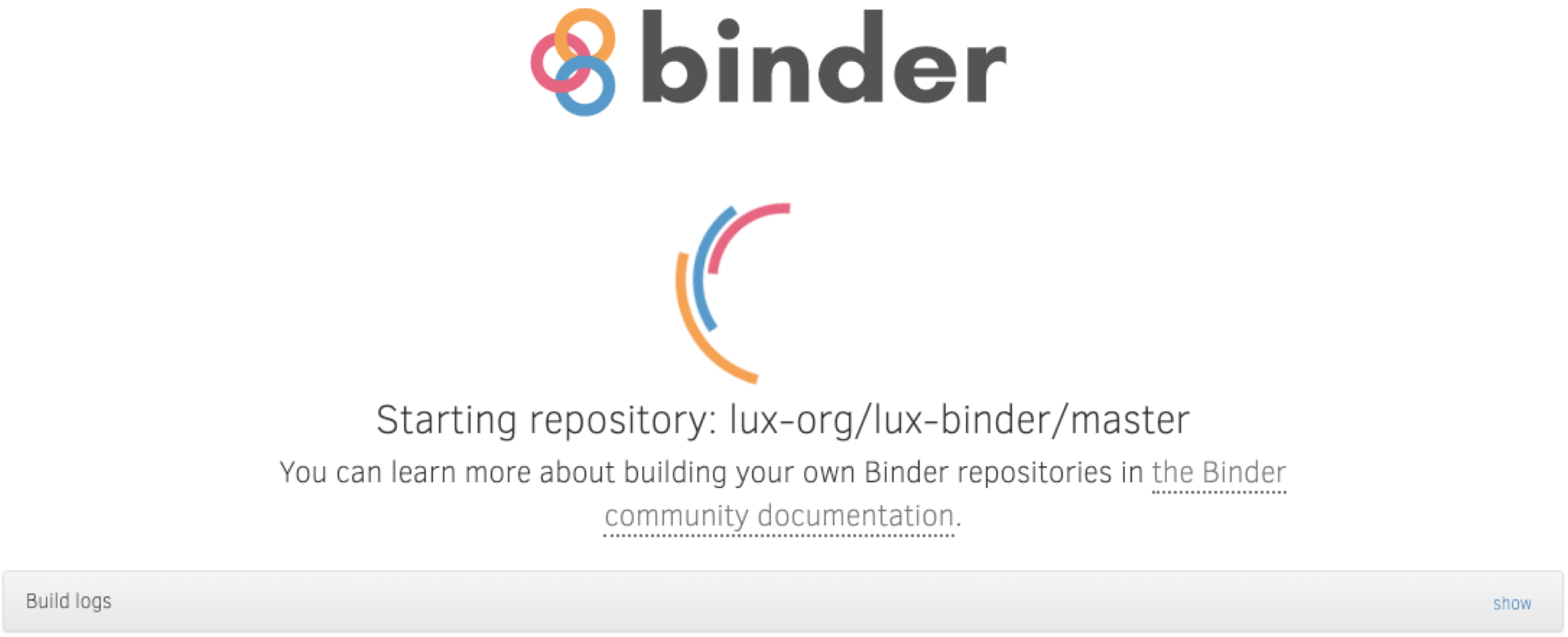
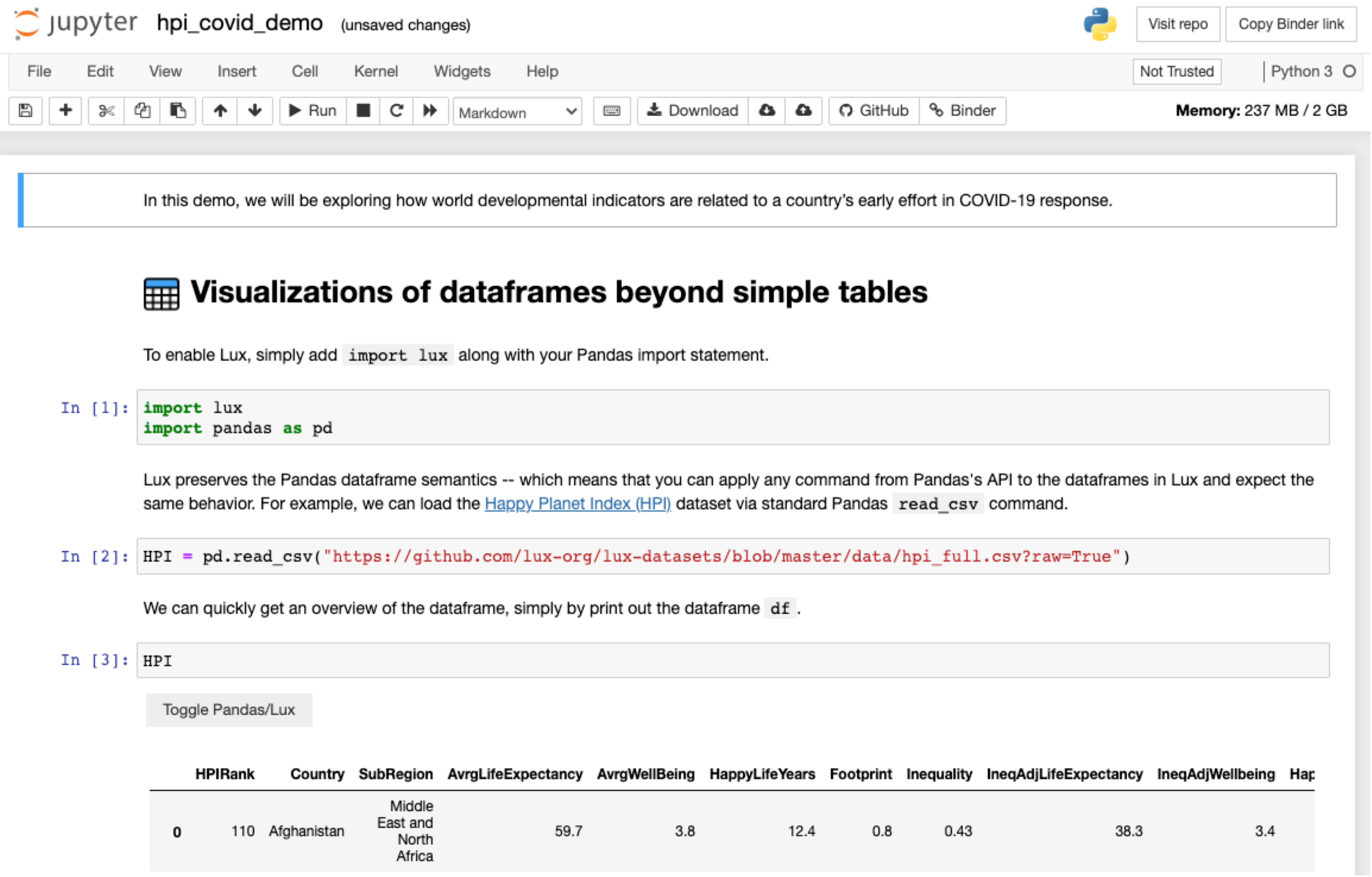
Discussion: Lux

Hands-on Example Demo



Starting repository: lux-org/lux-binder/master

You can learn more about building your own Binder repositories in the Binder community documentation.



In this demo, we will be exploring how world developmental indicators are related to a country's early effort in COVID-19 response.

Visualizations of dataframes beyond simple tables

To enable Lux, simply add `import lux` along with your Pandas import statement.

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

Lux preserves the Pandas dataframe semantics -- which means that you can apply any command from Pandas's API to the dataframes in Lux and expect the same behavior. For example, we can load the Happy Planet Index (HPI) dataset via standard Pandas `read_csv` command.

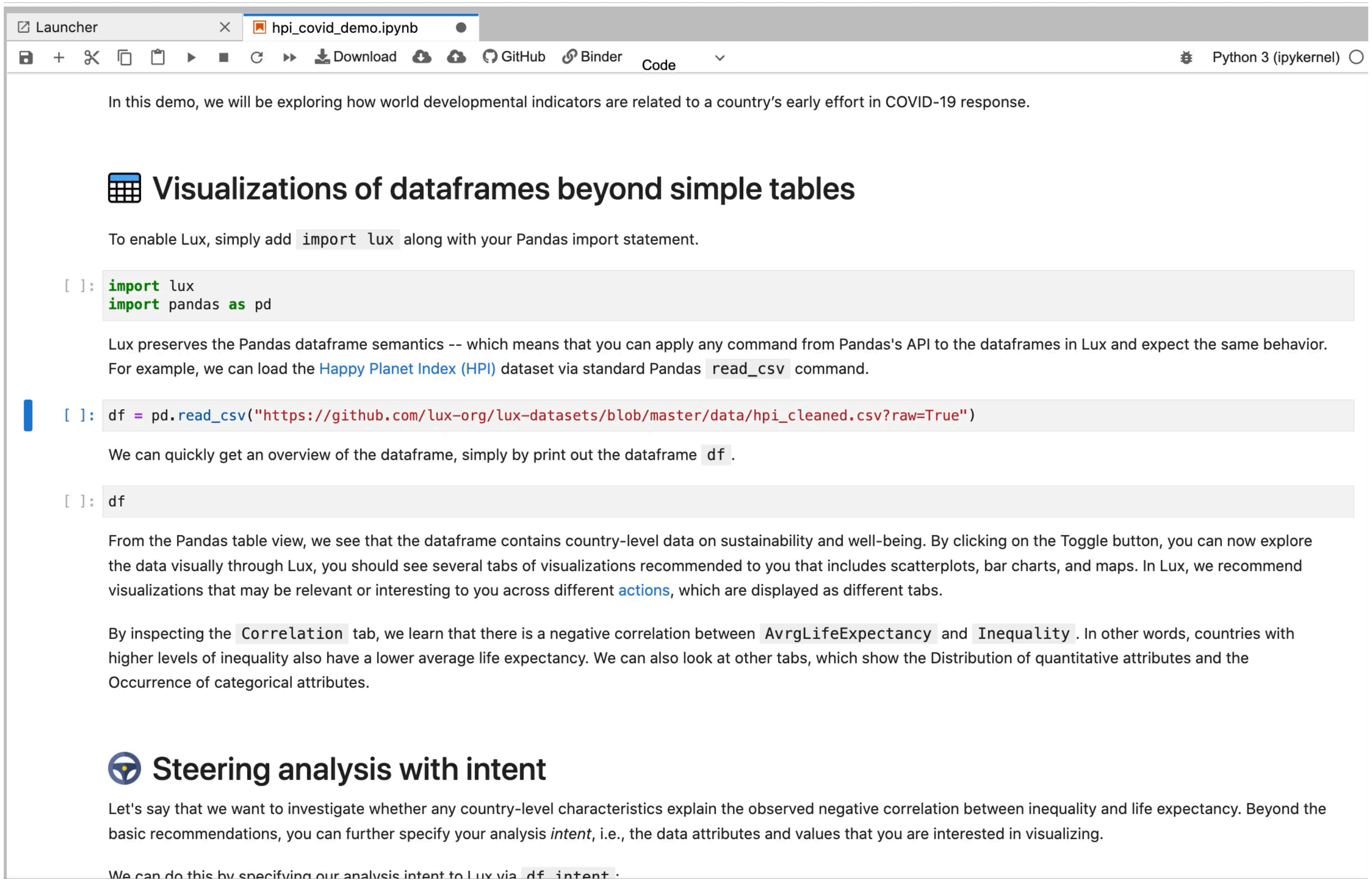
```
In [2]: HPI = pd.read_csv("https://github.com/lux-org/lux-datasets/blob/master/data/hpi_full.csv?raw=True")
```

We can quickly get an overview of the dataframe, simply by print out the dataframe `df` .

```
In [3]: HPI
```

	HPIRank	Country	SubRegion	AvrgLifeExpectancy	AvrgWellBeing	HappyLifeYears	Footprint	Inequality	IneqAdjLifeExpectancy	IneqAdjWellbeing	Hap
0	110	Afghanistan	Middle East and North Africa	59.7	3.8	12.4	0.8	0.43	38.3	3.4	

tinyurl.com/demo-lux



In this demo, we will be exploring how world developmental indicators are related to a country's early effort in COVID-19 response.

Visualizations of dataframes beyond simple tables

To enable Lux, simply add `import lux` along with your Pandas import statement.

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

Lux preserves the Pandas dataframe semantics -- which means that you can apply any command from Pandas's API to the dataframes in Lux and expect the same behavior. For example, we can load the Happy Planet Index (HPI) dataset via standard Pandas `read_csv` command.

```
[ ]: df = pd.read_csv("https://github.com/lux-org/lux-datasets/blob/master/data/hpi_cleaned.csv?raw=True")
```

We can quickly get an overview of the dataframe, simply by print out the dataframe `df`.

```
[ ]: df
```

From the Pandas table view, we see that the dataframe contains country-level data on sustainability and well-being. By clicking on the Toggle button, you can now explore the data visually through Lux, you should see several tabs of visualizations recommended to you that includes scatterplots, bar charts, and maps. In Lux, we recommend visualizations that may be relevant or interesting to you across different actions, which are displayed as different tabs.

By inspecting the `Correlation` tab, we learn that there is a negative correlation between `AvrgLifeExpectancy` and `Inequality`. In other words, countries with higher levels of inequality also have a lower average life expectancy. We can also look at other tabs, which show the Distribution of quantitative attributes and the Occurrence of categorical attributes.

Steering analysis with intent

Let's say that we want to investigate whether any country-level characteristics explain the observed negative correlation between inequality and life expectancy. Beyond the basic recommendations, you can further specify your analysis *intent*, i.e., the data attributes and values that you are interested in visualizing.

We can do this by specifying our analysis intent to Lux via `df.intent`:

So…

- What did you think of the paper?
  - Writing
  - Interface/Approach
  - Evaluation

Use Cases

- Where in typical workflows would it be actually useful? Where would it not be useful?

Interface

- What are other possible interfaces Lux could have used to bring vis rec to notebooks?

What other…

- … scalability ideas can you think of? Beyond lazy computation, prioritizing “cheaper” visualization, caching and reuse…
- Clearly, speed is a concern for Lux

Other Drawbacks

- Lux interactions need to be explicitly saved: no history
- Language for visualization construction is simple but a bit clunky
- Bespoke visualizations impossible