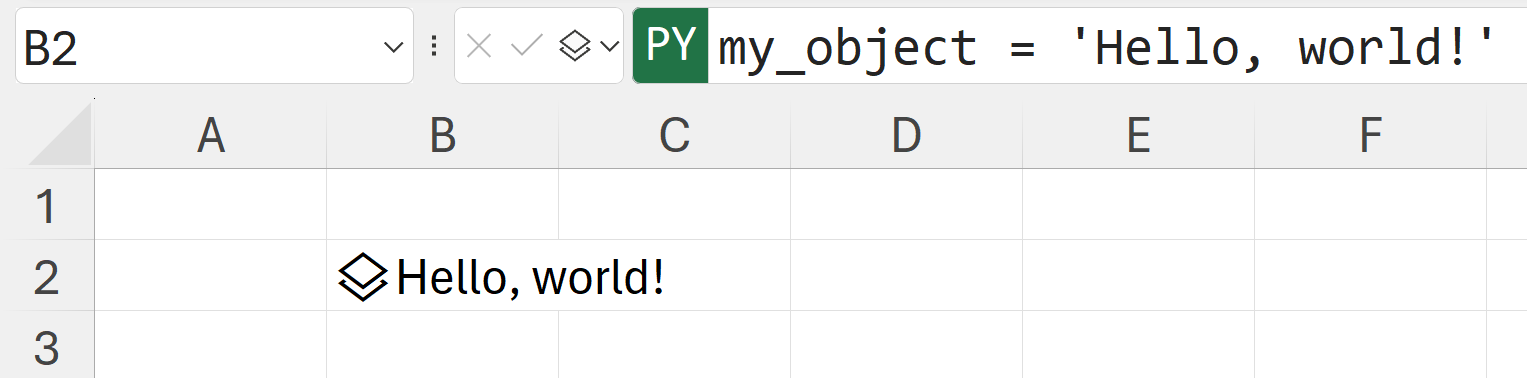
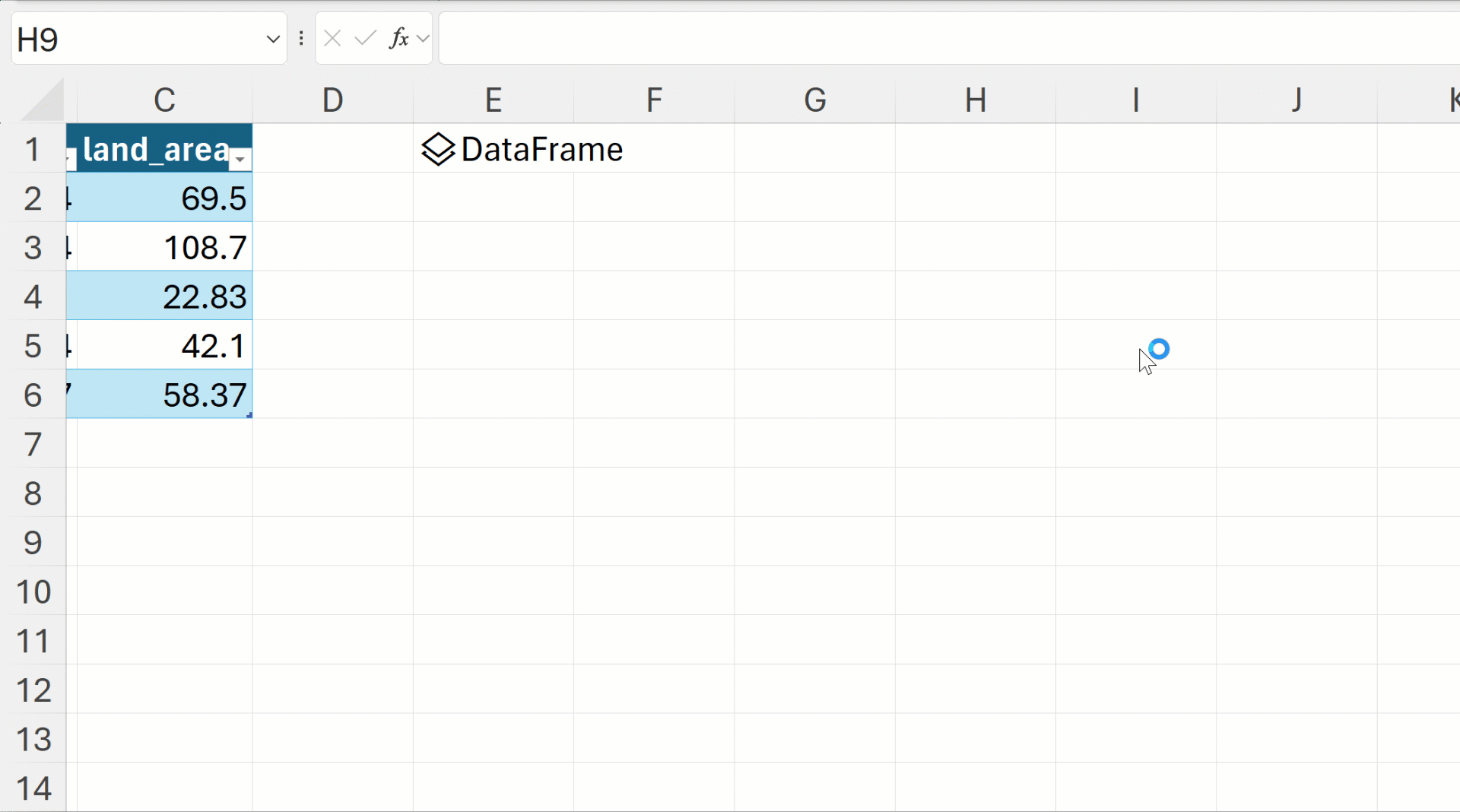
Python in Excel: First steps demo notes

* Make sure you’re enrolled in demo program! <https://support.microsoft.com/en-us/office/get-started-with-python-in-excel-a33fbcbe-065b-41d3-82cf-23d05397f53d>
* Ideally, learn a bit about Python independently of Excel
* Understand the Python in Excel initialization environment: https://stringfestanalytics.com/how-to-understand-the-initialization-environment-in-python-for-excel/
  1. See the complete set of packages: https://stringfestanalytics.com/how-to-see-all-packages-available-in-python-in-excel
* Start adding a Python cell, starting with Hello, world

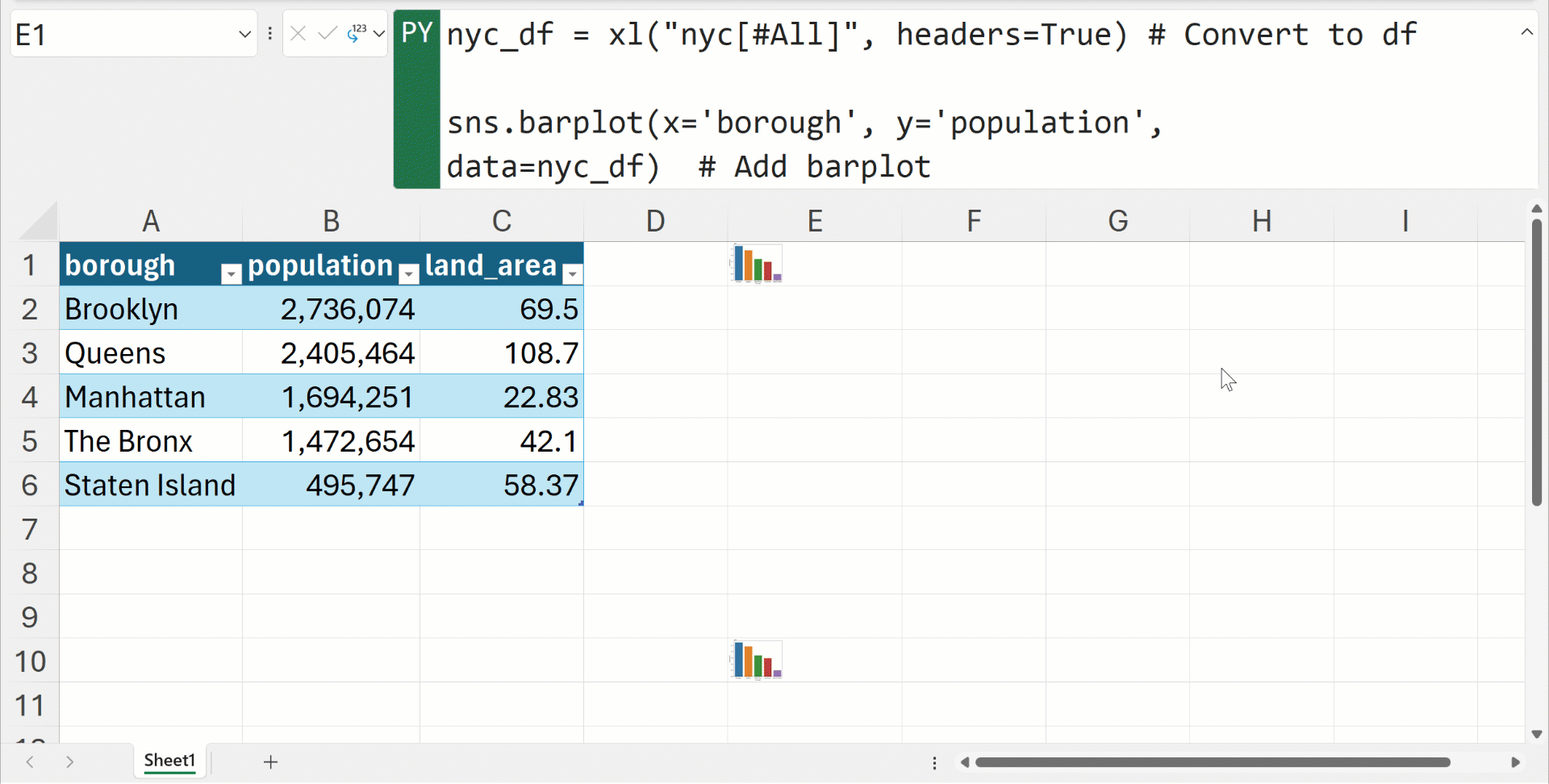


There are three ways to do this:

* Through the menu (Insert Python > Python in Excel)
* Through a formula (=PY)
* Keyboard shortcut (Ctrl Alt Shift P)
* Understanding the left-to-right, up-and-down order of operations
* Creating an Excel range into a Python object with the XL() function



* Make sure that headers=True gets turned on and works, a lot of Python’s functionality won’t work without properly named headers!
* Once the Python results have been loaded into Excel, you can build on them using the spill operator, these are dynamic arrays
* Python plots get inserted into Excel via images in cell. Try this with a simple barplot:
  + sns.barplot(x='borough', y='population', data=nyc)
  + Make sure you switch your cell from Python object to Excel output to see the plot
  + You can make this plot bigger by right-clicking and selecting either Place over Cells or Create Reference



* By simply overlaying the plot onto cells, you sever the connection to the data source. This might be acceptable if you require a one-time visualization and are not concerned with reproducibility or repeatability.
* However, if these aspects are important, then the additional effort of establishing that link becomes significant