Python Data Toolkit

If you cannot manipulate data, everything else in this course is moot. Familiarizing yourself with the tools listed below (I highly recommend going in the order presented) will prepare you to work with almost any data at any level in the Python environment. The most essential learning goals are listed along with each tool.

1. list: You will be able to work with lists of arbitrary items and use indexing/slicing to manipulate a subset of items from the list.
2. dict: You will be able to work with (key, value) pairs.
3. numpy: You will be able to work with N-dimensional arrays (e.g., initialization, indexing/slicing, views, math, aggregation, logical masks, matrix multiplication). You will appreciate the power of N-D arrays for many types of data, and the speed and clarity of numpy vs. pure python.
4. pandas: You will be able to work with dataframes (i.e., tables; indexing/slicing, to/from numpy, groupby, plot). You will appreciate the power of pandas dataframes for exploratory data analysis and see that pandas far exceeds the ability of programs like Excel to manage large tabular datasets.

Although not as fundamental as the above, the following tools may certainly be of interest.

5. xarray: You will appreciate the usefulness of N-D arrays with labeled dimensions. This is numpy + pandas to the next level.
6. zarr: You will be able to store data in a self-describing hierarchical format with chunked arrays either locally or in the cloud.
7. dask: You will be able to perform computations on datasets too large to fit into your local computer memory.

Other resources of interest:

- pydata: Community leveraging python to work with many types of data.
- parquet: A modern and vastly superior format for storing large tabular datasets as compared to comma separated values (.csv).