Building a REST API with Python and Flask

REST (REpresentational State Transfer) is an architectural style used for building APIs using the World Wide Web's Hypertext Transfer Protocol. REST is probably the most widely used method of building APIs designed for use over the Internet, and a wide range of software tools and frameworks provide support for creating and using REST APIs.

In this workshop, we'll use the **Flask** framework and the **Python** scripting language to create a simple REST API for creating, listing, reading, updating and deleting resources over the Internet. Flask (http://flask.pocoo.org/) is a very lightweight Python framework (or micro framework) that can be used for creating Web applications of all types. Its simplicity, robustness and extensibility make it a great choice for creating REST APIs.

Participants who wish to follow along interactively should bring a laptop running a fairly up-to-date version of OS X, Windows or Linux. We will go through the steps of installing Flask and related packages during the workshop. It would be very helpful, however, if participants come to the workshop with the following software pre-loaded on their systems:

sqlite. Lightweight relational database management system, which we will use as the data store for the REST API. Sqlite is preloaded on OS X and many Linux distributions. Windows install instructions available at http://www.sqlitetutorial.net/download-install-sqlite/. NOTE: Windows users should download *both* the command-line shell program and the appropriate DLL (64-bit or 32-bit depending on your CPU_.

python. Interpreter for the Python scripting language. Python is preloaded on OS X and virtually all Linux distributions. Windows install instructions available at https://www.howtogeek.com/197947/how-to-install-python-on-windows/. NOTE: either Python 2 or Python 3 will work for purposes of this workshop.

pip. Package management system for Python. If you have a recent version of Python, pip may be installed with Python. To find out if pip is installed, open a terminal window or command prompt and type `pip --version`. If pip is not found, install instructions are available at https://dev.to/el_joft/installing-pip-on-windows (for Windows) or https://ahmadawais.com/install-pip-macos-os-x-python/ (for OS X).

virtualenv. Tool for creating different environments for different Python projects. May be installed using **pip**, as described here: https://virtualenv.pypa.io/en/stable/installation/.

curl (optional). Command-line tool for making HTTP requests. May be used to test the API from outside a Python/Flask environment. Download instructions for many operating systems at: https://curl.haxx.se/download.html.