

Basic Programming in Python: Project examples

Sebastian Höffner Aline Vilks

Wed, 14 June 2017

In this document we present a few project ideas to get you started. They are not in any specific order. The projects are by no means fully fleshed out and should just give you some ideas.

Note that we added some libraries here and there, those are just libraries which came to our minds first when thinking about the problems. There might be better options, they might be too big for your problems, they are sometimes a little bit difficult to use or even understand. While they might not solve your final projects, you might find this list come in handy in the future, when you tackle real world projects.

Vocabulary trainer

Create a program to learn words of a new language by the card index box pattern. Asks for a translation, progresses if correct, returns if wrong.

Function visualizer

Allow to visualize functions by adjusting parameters, e.g. $f(x) = x^p$ where p can be changed. Plots the resulting function over a range of numbers to provide some intuition about how functions look like.

Cool packages and modules: `matplotlib`, `numpy`

Calculator

Implement a simple calculator, either with some graphical user interface (e.g. `tkinter`) or a command line interface. Allow at least for base operations like `+`, `-`, `*`, `/`. Stretch goal: Add parentheses.

TODO list

Store a list of tasks and allow to cross them off, add to them, modify them. Alternatively one can think of shopping lists or similar.

Cool packages and modules: `sqlite3`

Travel planer

Plan you next vacation by using public APIs to get information about a goal location: weather, distance, hotels, currency exchange rates, ...

Cool packages and modules: `requests`, `json`

(Esoteric) Language interpreter

Write a language interpreter for an esoteric programming language, like the popular brainfuck, the delicious chef or the poetic Shakespeare.

Cool packages and modules: `parse`

Raspberry Pi

If you happen to own a Raspberry Pi mini computer, you can use the `RPi.GPIO` to play around with it and build some light shows or similar things.

Cool packages and modules: `RPi.GPIO`

Image processing

Create a simple image processing tool which allows e.g. cropping of images, filtering (blurs, sepia, vintage, ...), separates the background, etc.

Cool packages and modules: `pillow`, `matplotlib`, `scipy`, `OpenCV`

Audio processing

Generate a simple text to speech processor, build a music player which streams music, or sort your MP3s by their genres.

Cool packages and modules: `eyeD3`, <https://wiki.python.org/moin/Audio/>

Games

Implement a classic game like Battleship, tic tac toe, or Connect four. If you are ambitious checkout `tkinter`, `PyGame` or similar libraries to create graphical games.

Cool packages and modules: e.g. `tkinter`, `PyGame`

Webservice

Build a small web server which can serve websites or allow blogging. Handle a login procedure, store values in a database.

Cool packages and modules: `flask`, `django`, `http.server`, `sqlite3`

Data analysis

Grab a data set from the UCI Machine Learning repository, kaggle, your own data, ... and analyse it. Plot it, calculate means, fit linear models, filter the data, ...

Cool packages and modules: `numpy`, `scipy`, `matplotlib`, `scikit-learn`, `h5py` (if using matlab data)

Web crawler

Download your favorite news websites' latest headlines. Or scrape wikipedia for company logos. Generate a calendar from your facebook events. You choose! Take a look at e.g. `BeautifulSoup` to handle html easily.

Cool packages and modules: `requests`, `BeautifulSoup`

Machine learning

Pick some dataset from the UCI Machine Learning repository, kaggle, etc. and try to use a machine learning model to learn some structure in it.

Cool packages and modules: `scikit-learn`, `tensorflow`