Postal Codes

UK Postcodes

UK postcodes, although common, are not all publicly-accessible data, so OpenStreetMap lacks information about the majority of them. Because of this, I was unable to implement my postcode-search feature by using it alone. I found that there were several tools on the internet that were able to convert postcodes into geo-coordinates, and thought that, if I could find a version of these that could be operated via API calls, then I could use these alongside data from OSM - as I had just implemented a feature for creating a map out of a given set of geo-coordinates, so this would not be difficult to do. I ended up coming across a tool named “postcodes.io” for this purpose.

Postcodes.io[1] is a free, open-source tool with an MIT licence[2], meaning that one can use it as much as one wants, so long as one distributes the licence when doing so. It is updated regularly, and is very easy-to-use. I was able to easily integrate this tool into my application (after creating a test program interacting with it) and get the postcode-search up and running. However, I did not know how to validate a UK postcode, so it lacks any kind of input validation. I will be looking into the best way to do this in the section below.

Validation

UK postcodes contain 5-7 alphanumeric characters. They are always in uppercase, and are split into two parts, which are separated by a space. The first part - the “outward code” - consists of 2-4 characters, and the second part - the “inward code” - always consists of 3 characters[3].

The inward code follows the format NUMBER-LETTER-LETTER in all instances. The outward code, however, has a far larger variety of valid formats: It can start with either one or two letters, and then be followed by either one number, two numbers, or a number and a letter. This makes the outward code difficult to validate.

The postcode-to-coordinates tool that I am using in my project - postcodes.io - does not care about a lack of spaces or uppercase. It also comes with a tool for checking whether a postcode is valid, but it is preferable to check this on the frontend instead of via an API (so not to further delay users with a slower internet connection).

With all this in mind, it may just be best to check for the presence of an input and a length between 5-7 characters (with the space excluded) when validating these inputs, and let the application return an error when the postcode is found to be invalid by the postcodes.io API.

Global Postal Codes

My application was never meant to serve the UK alone, so only having the ability to search for UK postcodes is not sufficient. Ideally, I would include support for every type of postal code currently in existence, but there is no guarantee there would be equivalent technologies to postcodes.io for processing these. That, however, is what I aim to find out in this section.

There are 160 countries that have some sort of postal code[4]. Different countries have different laws about the kinds of data stored and made publicly available, especially when it comes to something as personal as an address, so it's unlikely that data will be able to be retrieved from all of them.

In my search, I have found a tool which can do exactly what I need, (i.e. postal codes-to-coordinates conversion, otherwise known as "geocoding",) and can do so for 83 countries. This tool is called "pgeocode"[5] and is a Python module that sources its data from the GeoNames database[6].

Data from the GeoName database is licensed under a Creative Commons Attribution Licence[7], meaning that it is free-to-use as long as you give credit as to where the data was sourced from. "pgeocode" is licensed under a 3-Clause BSD Licence[8], which means that it can be used and modified in any way, provided that the licence is included. In summary, I am freely able to use both of these resources in my application as long as I make the source of my data clear and include all the necessary licences.

The postal code look-up for "pgeocode" is more strict than that of postcodes.io - as it requires the space to be present. If the space (between the outward code and the inward code) is not present, then the postal code will be seen as invalid. It is, however, still case-insensitive. Because pgeocode has data for UK addresses and can perform the same function as postcodes.io, then it could be used as a replacement for it to simplify the code used in the postcode-search. I would, however, still need to keep the licence for postcodes.io in the repository - as I have no intentions of deleting the test program that utilises it.

Due to the vast differences in the formatting of the postal codes of 83 different countries, it would appear to be an impossible task to implement validation for all of them within a time-limit that is reasonable for an intermediary version. Thus, I will simply notify the user that they must use the correct spacing for that country's postal codes, and only check for the presence of an input before attempting to process it.

Summary

OpenStreetMap alone cannot accommodate for postal code look-ups due to its lack of data on postal codes (due to them being not readily-available). Thus, in order to implement this feature, another tool must be used to convert a postal code into something that can be used to query the OSM database. In this circumstance, conversion to geo-coordinates would be best.

I can use the open-source Python module "pgeocode" to convert postal codes from 83 different countries into geo-coordinates, and use those coordinates in queries to the OpenStreetMap database (via the Overpass API) to generate a map of the surrounding area. To do so legally, I must include credit to the GeoName database in the postal-code-search section of my application, and the licence for pgeocode in the distribution of my application. I will also be including the licence for postcodes.io in my repository - as I experimented with it during the research/development process and still have a test program that makes use of it within the "Test Programs" folder of my repository.

References

1. http://postcodes.io/
2. https://opensource.org/licenses/MIT
3. https://ideal-postcodes.co.uk/guides/uk-postcode-format
4. https://www.upu.int/UPU/media/upu/documents/PostCode/General-Addressing-Issues.pdf
5. https://pypi.org/project/pgeocode/
6. http://www.geonames.org/about.html
7. https://creativecommons.org/licenses/by/3.0/
8. https://opensource.org/licenses/BSD-3-Clause