Parsers

In the following lessons you will learn how to use the **BeautifulSoup** library to pull data out of HTML and XML files. BeautifulSoup uses a parser to transform files into a tree of Python objects that can be easily searched. So, before we start learning how to use BeautifulSoup, let's take a quick look at parsers.

In BeautifulSoup, the **parser** is a piece of software whose primary job is to build a data structure in the form of a hierarchical tree that gives a structural representation of the HTML or XML file. In other words, the parser divides these complex files into simpler parts while keeping track of how these parts are related to each other. BeautifulSoup supports a number of parsers, but throughout these lessons we will only be using the **lxml** parser. The `lxml` parser can be used to parse both HTML and XML files and has the advantage of being very fast. In order to use the `lxml` parser, you must have `lxml` installed. You can install the `lxml` parser by using the following command in your terminal:

```
$ pip install lxml
```

If you're working with perfectly formatted HTML or XML files (*i.e.* files that don't contain any missing information or mistakes) then, in the majority of cases, your choice of parser shouldn't really matter. However, if the files you are working with have missing information or mistakes, then your choice of parser will matter because each parser has different rules for dealing with missing information or mistakes. Consequently, in these cases, different parsers will create different parse trees for the same document. You can take a look at the **differences between parsers**, in the BeautifulSoup documentation, for details.