Working with APIs

What is an API?

An Application Programming Interface (API) is a method for one application to interact with another application that protects both applications. Imagine if you wanted to build an application that showed where the next bus leaving from Humber College was on a map. You simply would not be able to build that data source on your own. However, another company has built a data application that does have access to that information, and they have formatted the data so that it can be used by your application without giving you access to their application, and without accessing your application. This interaction (interfacing) between the two applications is made possible by a set of commands created by the data application – the API. Your Humber bus application uses the API commands to retrieve data from the data application, then displays the desired data in your application.

How to Work With One: - Read the Documentation!

Once you find a data source that provides the data that you want to use, you will need to read the API documentation. The documentation will specify the commands or URLs that you will need, along with the parameters or variables that you can set, and the type of data that it will return. APIs might return an XML file, a text file, some HTML, a JSON string, or a CSV file. In some cases where there are multiple options, you will be able to specify which data format you would like. Many API documents also provide example code and/or example result sets so that you can see what you are working with.

Some APIs have fees or limits for usage, or require you to sign up (or purchase) a "key" to gain access to their API data. A key is a unique identifier string that you will add to your code so that the API knows that you are a registered user, and that you haven't exceeded your usage allowance.

Getting Started with an API

1. Find an API that has the data that you wish to include in your application.
2. Learn about the API from its website and determine:
    a. Do I need to register to use this API? What do I need to register?
    b. Are there costs to use this API? What are the costs?
    c. Are there limits to the use of this API? What are the conditions and limits?
    d. What data types/formats does this API provide?
3. **Read the API documentation** to learn the commands or URLs ("end points") that will return the data sets, and to find examples of the commands and data results.
4. Create a test page that retrieves data from the API and explore what is contained in the data result.