Advanced Web Development and Web Scraping
Fall 2018
Assignment #9 – JSON and Web APIs

1. The PokéApi (https://pokeapi.co) can be used to retrieve information about a Pokémon by its name. Example: https://pokeapi.co/api/v2/pokemon/pikachu/
   Create a web page that includes a textbox and a submit button where the user can enter the name of a Pokémon.

   a. When the user submits a name, a request is made to the PokéApi and the following information is displayed (examples for Pikachu are given):
      - The Pokémon's name (pikachu)
      - The Pokémon's image, (the *front_default* sprite should be displayed).
      - The Pokémon's height (divided by 10 to display in meters) and weight (divided by 10 and displayed in kilograms)
      - The Pokémon's abilities (lightning-rod, static)
      - The Pokémon's types (electric)

   b. If the user enters an invalid name, then after the request is made *xhttp.status* will be equal to 404 or 0; if this is the case, display an Error message that the Pokémon does not exist or could not be retrieved at this time. Note: a status of 0 indicates that the request was blocked by the API server, which may happen for invalid URLs.

   c. Use local storage to remember the information retrieved from the last valid request. This information saved in local storage should take the form of a single Javascript object, then converted to JSON, and stored using local storage. When the page is loaded, this information is read from local storage and the most recent successful search is displayed, if available.

   The following are Pokémon that can be used for testing: pikachu, bulbasaur, incineroar

2. Develop a Python script that uses the Open Movie Database API (https://www.omdbapi.com/). Note that you will need to sign up for a free API key (www.omdbapi.com/apikey.aspx). Your script should work as follows:
   a. Prompt the user to enter a movie (or keywords) to search for.
   b. Submit a request to the API to search by movie title (by setting the *s* parameter)
   c. If only 1 match is found, then submit an API request to get information from the movie (using the *i* parameter to set the IMDb ID), and display the results as described below.
   d. If multiple matches are found, display a list of matches and ask the user to select the number for the movie they want to retrieve. These results are in the following format (both title and year are displayed):
      1. Star Wars: Episode IV - A New Hope (1977)
      2. Star Wars: Episode V - The Empire Strikes Back (1980)
      3. (rest are omitted)

e. When the user selects the movie (e.g., they select *1* for Star Wars: Episode IV), a call is made to the API to retrieve information for the movie (looking up the movie by its IMDb id using the *i* parameter). The following information is then displayed:

Star Wars: Episode IV – A New Hope (1977)
Rated: PG
Release Date: 25 May 1977
Ratings:

Internet Movie Database:	8.6/10
Rotten Tomatoes:	93%
Metacritic:	90/100