PROG 2700 ASSIGNMENT 2C – Retrieving API Data

**Client-Side Programming**

# Prerequisites

### Strongly Recommended: Assignment 2A & 2B

# Summary

Using the publicly available Deck of Cards API, you will create a small JavaScript application that uses the API to provide data so that you can do the following.

* Retrieve a Deck of shuffled cards from the API.
* Initially pull 5 cards from the deck and display them in a web page.
* Write/research a function that takes the cards and shows the highest poker hand that can be calculated based on the 5 cards.

# Code Submission

**You must commit and push all of your code to your provided GitHub repository.** It is also required that you commit and push often so that you build a history of commits and pushes that you can show to the instructor.

# Requirements (40 points)

1. **Retrieve and persist a Deck of Cards from the API (10 pts)**

Using the Deck of Cards API (<https://deckofcardsapi.com/>), use a demonstrated method of AJAX data retrieval to retrieve a deck of cards that can be used by the application. Be sure to store the returned data in an appropriate way.

1. **Request Five Cards from the Deck (10 pts)**

Using the deck that was retrieved in REQ-001, ask the API for a hand of five cards from the deck. Store the given cards in an appropriate manner in your code so that you can evaluate its contents.

1. **Display the Hand in a Web Page (10 pts)**

Display the cards in some manner that can be seen in the browser. This can be done by either

|  |
| --- |
| 1. Displaying the cards names in some manner using *document.write* to the web page 2. Displaying the images of the cards on the web page by modifying the DOM. (ie Manipulate img tags defined on the page) |

1. **Write a Function that will determine the highest Poker Hand for the displayed cards (10 pts)**

Write a function that will determine and output the highest poker hand based on the given five cards:

Hand rankings can be found here: <https://www.cardplayer.com/rules-of-poker/hand-rankings>

or here: <https://www.unibet.com/poker/guides/poker-hand-rankings-with-cheat-sheet-1.784253>

1. **Wrap the entire application in an Independently Invoked Function Expression (IIFE) (5 pts)**

In order for the entire application to be contained within its own scope and to not pollute the global scope, wrap the entire contents of the file in an Independently Invoked Function Expression (IIFE) and be prepared to demonstrate how your script’s data is contained within its own local scope and not within the browser’s global scope (window).

# Instructions

1. **Don’t forget that a live in-person demonstration of your code is part of this assignment. You will need to show your code to the instructor in class on the due date while going through an evaluation of your code’s functionality. Part of the assessment will include your ability to speak about the code you wrote, even if it doesn’t completely work or do what you expect.**
2. **Late submissions will be subject to the late penalties laid out in the course outline.**

# Academic Integrity and Plagiarism

**Code sharing by any means is considered plagiarism and is strictly forbidden under the NSCC Academic Integrity policy.**

[NSCC ACADEMIC INTEGRITY GUIDELINES](https://www.nscc.ca/docs/about-nscc/policies-procedures/policy-academicintegrity.pdf)

[NSCC ACADEMIC INTEGRITY REPORTING POLICY](https://www.nscc.ca/docs/about-nscc/policies-procedures/procedures-academicintegritystudent.pdf)