Project 0

USE CASE:

Create an interactive web page that connects to an external API.

REQUIREMENTS:

GIT:
- The associate squad should maintain a group GitHub repo that contains branches for their individual work.

SDLC:
- While working on this project, the associate should maintain a Scrumban workflow using daily standups to discuss and keep all squad members apprised of everyone's status.
- A Kanban board should track their tasks and progress throughout the project.
    - The Kanban board should be on the squad GitHub repo.
- Standup notes should be maintained on each member's branches and feature their notes from every standup in a markdown file.

WIREFRAMES:
- Before beginning work on the web page, the associate should create a wireframe mockup of the webpage they intend to make. You must upload your mockup to your branch on the squad repo.

HTML:
- Create a well-formed HTML page that illustrates your understanding of elements, tags, and attributes. It should feature one or more of the following:
    - Links & Images
    - Tables & Lists
    - Block Elements & Inline Elements.

CSS:
- Create custom styling for the HTML page that demonstrates an understanding of the box model, flexbox, custom properties, and CSS declarations and selectors.

JAVASCRIPT:
- Leveraging your understanding of JavaScript, develop the requisite code to call out to an external API and perform DOM manipulation based on the result.
- Utilize your understanding of events, listeners, and event propagation to react to user interaction or other events in conjunction with the callout or otherwise.
- Following good practices, ensure the code is well documented, including:
    - Header comments that detail the name of the file, author(s), created date, last modified date, an overview of the purpose of the file.
    - Comments for each function outlining the purpose of the function.
    - Self-documenting code practices that include proper variable and function names.
    - Use of camelCase naming practices.
- Ensure your code handles errors gracefully using try-catch blocks.