personal-website/major-projects/project02/script.js

```
/*
  Name: Kevin Blinn
  Class: CPSC 3750
  Date Due: July 20th, 2024
  Assignment: Project 02 - Collect App Phase 1
*/

// This function runs when the entire DOM content has been loaded
document.addEventListener('DOMContentLoaded', function() {

  // Get the search form element by its ID
  const searchForm = document.getElementById('searchForm');

  // Check if the search form exists
  if (searchForm) {
    // Add an event listener to handle the form submission
    searchForm.addEventListener('submit', function(e) {

      // Prevent the default form submission
      e.preventDefault();

      // Get the value of the search input field, trimmed of whitespace
      const query = document.getElementById('searchQuery').value.trim();

      // Call the searchGames function with the query
      searchGames(query);
    });
  }

   // Parse the URL parameters and Get the 'id' parameter
  const params = new URLSearchParams(window.location.search);
  const gameId = params.get('id');

  /// Fetch and display the details of the specified game if it exists
  if (gameId) {
    fetchGameDetails(gameId);
  } else {
    searchGames('');
  }

  // Fetch and display stats about the games collection
  fetchStats();
```

```
});

// Function to search for games based on a query
function searchGames(query) {

  // My personal API key for RAWG Video Games Database API
  const apiKey = '02fdad8f60254eea8c8509bf12e6dcde';

  // Construct the API URL with the query and page size parameters
  const url = query
    ? `https://api.rawg.io/api/games?key=${apiKey}&search=${query}&page_size=100`
    : `https://api.rawg.io/api/games?key=${apiKey}&page_size=100`;

  // Make an AJAX request to the RAWG API
  $.ajax({
    url: url,
    method: 'GET',
    success: function(response) {

      // Check if the response contains any results and display it
      if (response.results.length > 0) {
        displaySearchResults(response.results);
      } else {
        // Display a message if no results are found
        displaySearchResults([]);
      }
    },
    error: function(xhr, status, error) {
      // Log any errors that occur during the request
      console.error('Error fetching games:', error);
    }
  });
}

// Function to fetch and display details of a specific game
function fetchGameDetails(id) {

  // Same steps as before
  const apiKey = '02fdad8f60254eea8c8509bf12e6dcde';
  const url = `https://api.rawg.io/api/games/${id}?key=${apiKey}`;

  // Make an AJAX request to the RAWG API
  $.ajax({
    url: url,
    method: 'GET',
    success: function(response) {
      // Display the game details
      displayGameDetails(response);
    },
    error: function(xhr, status, error) {
      // Log any errors that occur during the request
      console.error('Error fetching game details:', error);
    }
  });
}

// Function to display the search results on the page
function displaySearchResults(games) {

  // Get the results container element by its ID
  const resultsContainer = document.getElementById('results');

  // Clear any existing content in the results container
  resultsContainer.innerHTML = '';

  // Check if there are any games to display
  if (games.length === 0) {
    resultsContainer.innerHTML = '<p>No games match this title.</p>';
  } else {
    // Loop through each game and create a card element for it
    games.forEach(game => {
      // Create a div element for the game card
      const gameElement = document.createElement('div');

      // Add a CSS class to the game card
      gameElement.classList.add('game-card');

      // Set the inner HTML of the game card with the game's information
      gameElement.innerHTML = `
        <img src="${game.background_image}" alt="${game.name}" class="game-image">
        <h3>${game.name}</h3>
        <p>${game.released || 'No release date available.'}</p>
        <a href="details.html?id=${game.id}">View Details</a>`;

      // Append the game card to the results container
      resultsContainer.appendChild(gameElement);
    });
  }
}

```

```
// Function to display the details of a specific game on the page
function displayGameDetails(game) {

  // Get the game details container element by its ID
  const detailsContainer = document.getElementById('gameDetails');

  // Set the inner HTML of the details container with the game's information
  detailsContainer.innerHTML = `
    <img src="${game.background_image}" alt="${game.name}" class="game-image">
    <h2>${game.name}</h2>
    <p>${game.description_raw || 'No description available.'}</p>
    <p>Release Date: ${game.released}</p>
    <p>Rating: ${game.rating || 'N/A'}</p>
    <p>Developers: ${game.developers.map(d => d.name).join(', ')}</p>
    <p>Genres: ${game.genres.map(g => g.name).join(', ')}</p>
    <p>Platforms: ${game.platforms.map(p => p.platform.name).join(', ')}</p>`;
}

/* FUNCTIONS FOR GETTING THE STATS */

// Function to fetch and display various stats about the games collection
function fetchStats() {

  // API key for RAWG Video Games Database API
  const apiKey = '02fdad8f60254eea8c8509bf12e6dcde';

  // Fetch total number of games
  fetchTotalGames(apiKey);

  // Fetch most popular game
  fetchMostPopularGame(apiKey);

  // Fetch latest released game
  fetchLatestReleasedGame(apiKey);

  // Fetch highest rated game
  fetchHighestRatedGame(apiKey);
}

// Function to fetch and display the total number of games in the collection
function fetchTotalGames(apiKey) {
  // Construct the API URL with the page size parameter
  const url = `https://api.rawg.io/api/games?key=${apiKey}&page_size=1`;

  // Make an AJAX request to the RAWG API
  $.ajax({
    url: url,
    method: 'GET',
    success: function(response) {
      // Display the total number of games
      $('#totalGames').text(response.count);
    },
    error: function(xhr, status, error) {
      // Log any errors that occur during the request
      console.error('Error fetching total games:', error);
      // Display an error message if the request fails
      $('#totalGames').text('Error loading');
    }
  });
}

// Function to fetch and display the most popular game in the collection
function fetchMostPopularGame(apiKey) {
  // Construct the API URL with the ordering parameter for most recently added games
  const url = `https://api.rawg.io/api/games?key=${apiKey}&ordering=-added&page_size=1`;

  // Make an AJAX request to the RAWG API
  $.ajax({
    url: url,
    method: 'GET',
    success: function(response) {
      // Check if there are any results
      if (response.results.length > 0) {
        // Display the name of the most popular game
        $('#mostPopularGame').text(response.results[0].name);
      } else {
        // Display 'N/A' if no results are found
        $('#mostPopularGame').text('N/A');
      }
    },
    error: function(xhr, status, error) {
      // Log any errors that occur during the request
      console.error('Error fetching most popular game:', error);
      // Display an error message if the request fails
      $('#mostPopularGame').text('Error loading');
    }
  });
}
```

```

// Function to fetch and display the latest released game in the collection
function fetchLatestReleasedGame(apiKey) {
  // Construct the API URL with the ordering parameter for latest released games
  const url = `https://api.rawg.io/api/games?key=${apiKey}&ordering=-released&page_size=1`;

  // Make an AJAX request to the RAWG API
  $.ajax({
    url: url,
    method: 'GET',
    success: function(response) {
      // Check if there are any results
      if (response.results.length > 0) {
        // Display the name of the latest released game
        $('#latestReleasedGame').text(response.results[0].name);
      } else {
        // Display 'N/A' if no results are found
        $('#latestReleasedGame').text('N/A');
      }
    },
    error: function(xhr, status, error) {
      // Log any errors that occur during the request
      console.error('Error fetching latest released game:', error);
      // Display an error message if the request fails
      $('#latestReleasedGame').text('Error loading');
    }
  });
}

// Function to fetch and display the highest rated game in the collection
function fetchHighestRatedGame(apiKey) {
  // Construct the API URL with the ordering parameter for highest rated games
  const url = `https://api.rawg.io/api/games?key=${apiKey}&ordering=-rating&page_size=1`;

  // Make an AJAX request to the RAWG API
  $.ajax({
    url: url,
    method: 'GET',
    success: function(response) {
      // Check if there are any results
      if (response.results.length > 0) {
        // Display the name of the highest rated game
        $('#highestRatedGame').text(response.results[0].name);
      } else {
        // Display 'N/A' if no results are found
        $('#highestRatedGame').text('N/A');
      }
    },
    error: function(xhr, status, error) {
      // Log any errors that occur during the request
      console.error('Error fetching highest rated game:', error);
      // Display an error message if the request fails
      $('#highestRatedGame').text('Error loading');
    }
  });
}

```