DWA_03.4 Knowledge Check_DWA3.1

---

1. Please show how you applied a Markdown File to a piece of your code.

See:
https://github.com/Arc4d3-G/DEWBRE300_FTO2308_Group2A_DewaldBreed_IWACapstone/blob/main/README.md

---

2. Please show how you applied JSDoc Comments to a piece of your code.

```javascript
/**
 * @property {Function} generatePreviewHtml - Function that creates and returns
 *   HTML for book previews to be appended to the list element.
 *
 * @param {Object} targetObject - Accepts any object literal that
 * contains book information, in this case objects {@link book} from
 * data.js, or once a search form has been submitted, {@link searchResult}
 * from scripts.js .
 *
 * @param {number} rangeStart - Defines where slicing of the targetObject
 * should occur, which is calculated by taking the current {@link page} value
 * and multiplying it by {@link BOOKS_PER_PAGE}
 *
 * @param {number} rangeEnd - Similar to rangeStart but defines where slicing
 * should end.
 *
 * @returns {DocumentFragment} listFragment - Returns the newly created document
 * fragment which can then be appended to the list element.
 */
export const generatePreviewHtml = (targetObject, rangeStart, rangeEnd) => {
    const listFragment = document.createDocumentFragment()
    const extractedBooks = targetObject.slice(rangeStart, rangeEnd)
    for (let { author, image, title, id } of extractedBooks) {
        author = authors[author] // to display the author as a name and not an id

        const preview = document.createElement('button')
        preview.className = 'preview'
        preview.dataset.id = id

        preview.innerHTML = /* html */ `
        <img
                class="preview__image"
                src="${image}"
        />

        <div class="preview__info">
            <h3 class="preview__title">${title}</h3>
            <div class="preview__author">${author}</div>
        </div>
        `

        listFragment.appendChild(preview)

    }
    return listFragment
}
```

---

3. Please show how you applied the @ts-check annotation to a piece of your code.

In this snippet, the second and third parameters of "generatePreviewHtml" should be numbers. Using @ts-check, it shows an error because the second param is a string, and should be a number.

```
// @ts-check
import { generatePreviewHtml, html } from "./generators";

                            Argument of type 'string' is not assignable to parameter of type 'number'. ts(2345)

                            View Problem (Alt+F8)    Quick Fix... (Ctrl+.)
generatePreviewHtml(html, "6", 7)
```

_____

4. As a BONUS, please show how you applied any other concept covered in the 'Documentation' module.

Added a Table of Contents to my README.md using markdown all in one extension:

```
<!-- omit in toc -->
# Book-Connect

Table of Contents (up to date)
- [About](#about)
- [User Stories](#user-stories)
- [Features](#features)
- [Requirements](#requirements)
- [How to use:](#how-to-use)
```

_____