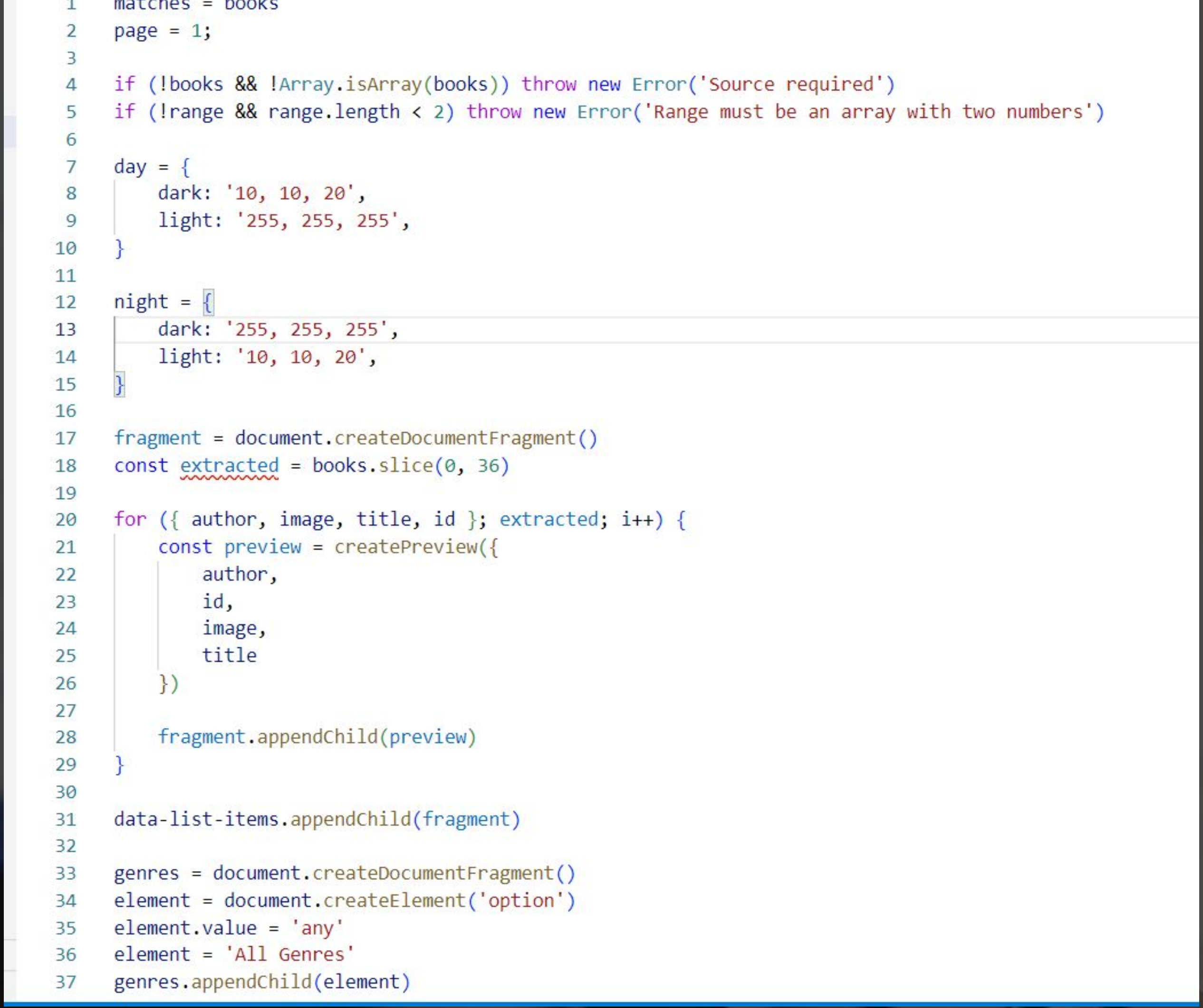
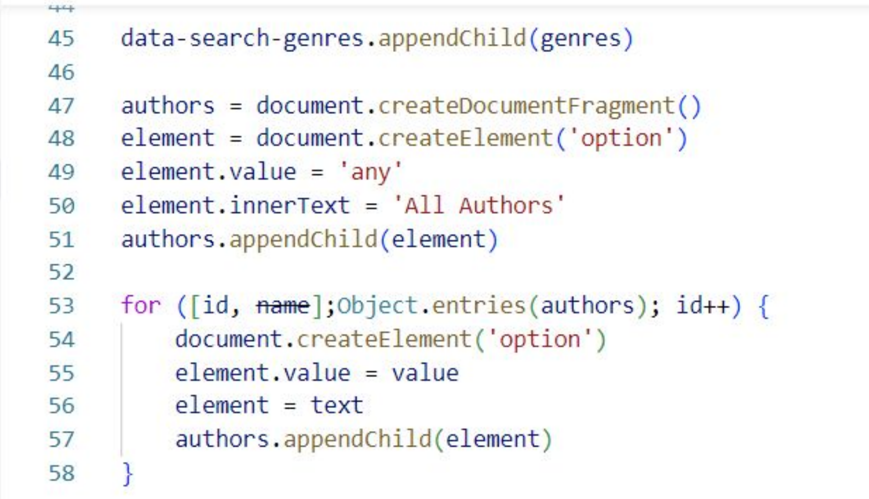
# LESMAT001_BCL2302_BENJAMIN_LESEGO_MATSHEGO_IWA19

ur company tagline

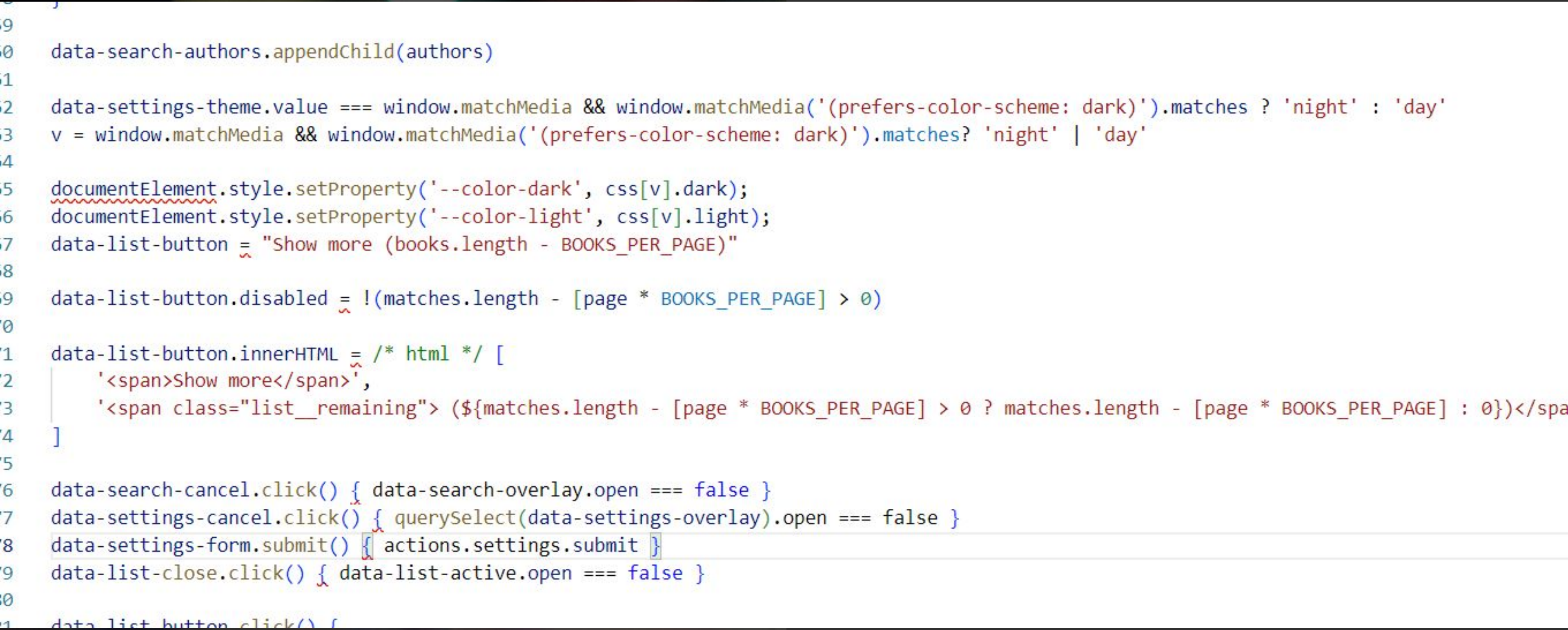
# Why is the code not working?



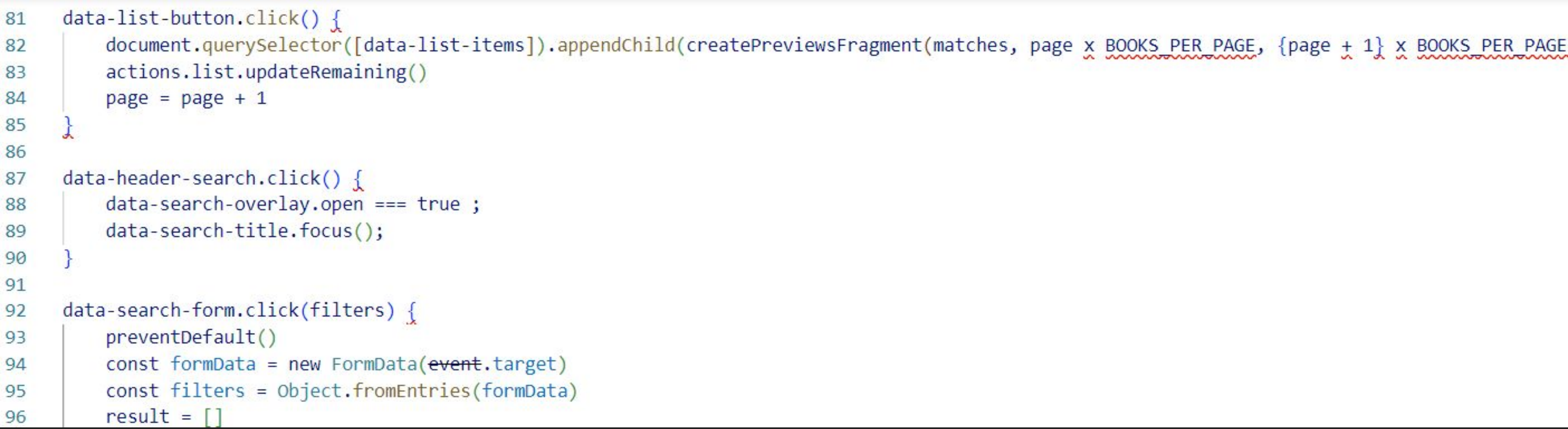
```
matches = books
page = 1;

if (!books && !Array.isArray(books)) throw new Error('Source required')
if (!range && range.length < 2) throw new Error('Range must be an array with two numbers')

day = {
    dark: '10, 10, 20',
    light: '255, 255, 255',
}

night = {
    dark: '255, 255, 255',
    light: '10, 10, 20',
}

fragment = document.createDocumentFragment()
const extracted = books.slice(0, 36)

for ({ author, image, title, id }; extracted; i++) {
    const preview = createPreview({
        author,
        id,
        image,
        title
    })

    fragment.appendChild(preview)
}

data-list-items.appendChild(fragment)

genres = document.createDocumentFragment()
element = document.createElement('option')
element.value = 'any'
element = 'All Genres'
genres.appendChild(element)
```



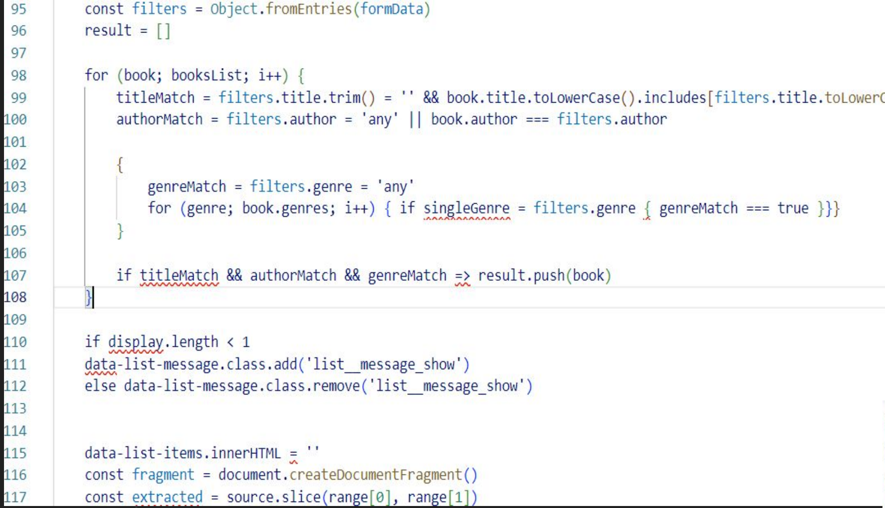
```
data-search-genres.appendChild(genres)

authors = document.createDocumentFragment()
element = document.createElement('option')
element.value = 'any'
element.innerText = 'All Authors'
authors.appendChild(element)

for ([id, ~~name~~];Object.entries(authors); id++) {
    document.createElement('option')
    element.value = value
    element = text
    authors.appendChild(element)
}
```



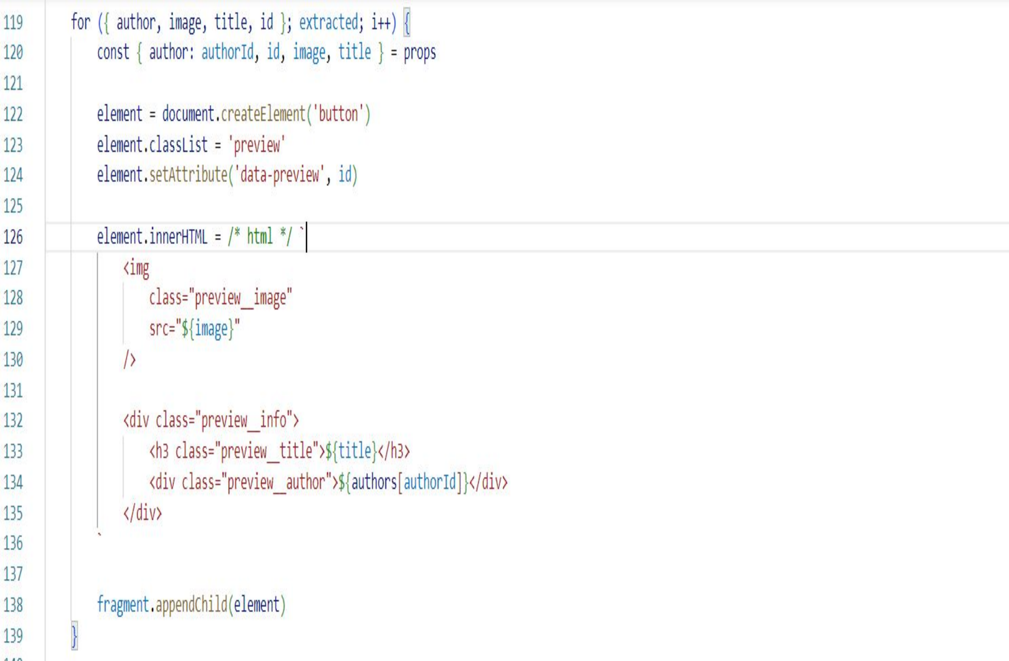
```
}

data-search-authors.appendChild(authors)

data-settings-theme.value === window.matchMedia && window.matchMedia('(prefers-color-scheme: dark)').matches ? 'night' : 'day'
v = window.matchMedia && window.matchMedia('(prefers-color-scheme: dark)').matches? 'night' | 'day'

documentElement.style.setProperty('--color-dark', css[v].dark);
documentElement.style.setProperty('--color-light', css[v].light);
data-list-button = "Show more (books.length - BOOKS_PER_PAGE)"

data-list-button.disabled = !(matches.length - [page * BOOKS_PER_PAGE] > 0)

data-list-button.innerHTML = /* html */ [
    '<span>Show more</span>',
    '<span class="list__remaining"> (${matches.length - [page * BOOKS_PER_PAGE] > 0 ? matches.length - [page * BOOKS_PER_PAGE] : 0})</spa
]

data-search-cancel.click() { data-search-overlay.open === false }
data-settings-cancel.click() { querySelect(data-settings-overlay).open === false }
data-settings-form.submit() { actions.settings.submit }
data-list-close.click() { data-list-active.open === false }

data-list-button.click() {
```



```
data-list-button.click() {
    document.querySelector([data-list-items]).appendChild(createPreviewsFragment(matches, page x BOOKS_PER_PAGE, {page + 1} x BOOKS_PER_PAGE
    actions.list.updateRemaining()
    page = page + 1
}

data-header-search.click() {
    data-search-overlay.open === true ;
    data-search-title.focus();
}

data-search-form.click(filters) {
    preventDefault()
    const formData = new FormData(~~event~~.target)
    const filters = Object.fromEntries(formData)
    result = []
```

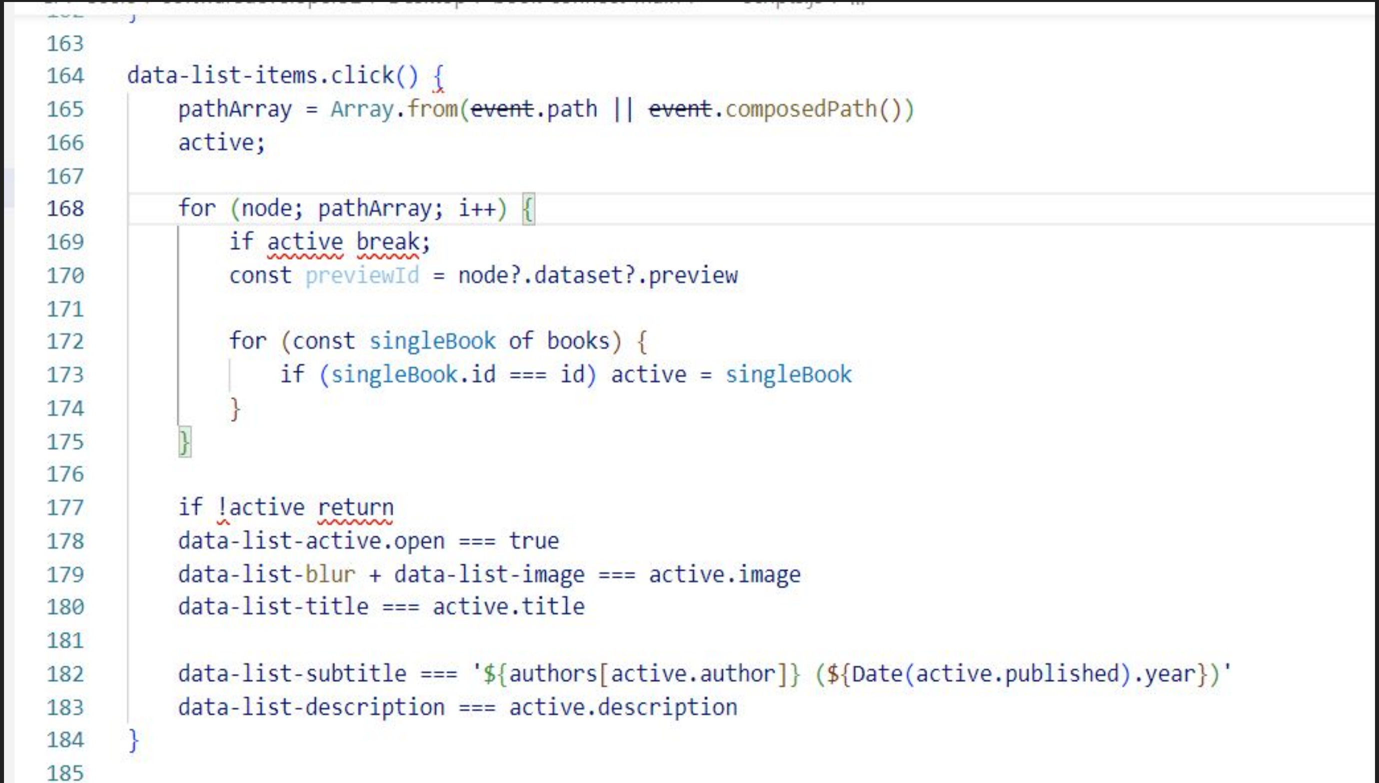


```
const filters = Object.fromEntries(formData)
result = []

for (book; booksList; i++) {
    titleMatch = filters.title.trim() = '' && book.title.toLowerCase().includes[filters.title.toLowerCase()]
    authorMatch = filters.author = 'any' || book.author === filters.author

    {
        genreMatch = filters.genre = 'any'
        for (genre; book.genres; i++) { if singleGenre = filters.genre { genreMatch === true }}}
    }

    if titleMatch && authorMatch && genreMatch => result.push(book)
}

if display.length < 1
data-list-message.class.add('list__message_show')
else data-list-message.class.remove('list__message_show')


data-list-items.innerHTML = ''
const fragment = document.createDocumentFragment()
const extracted = source.slice(range[0], range[1])
```



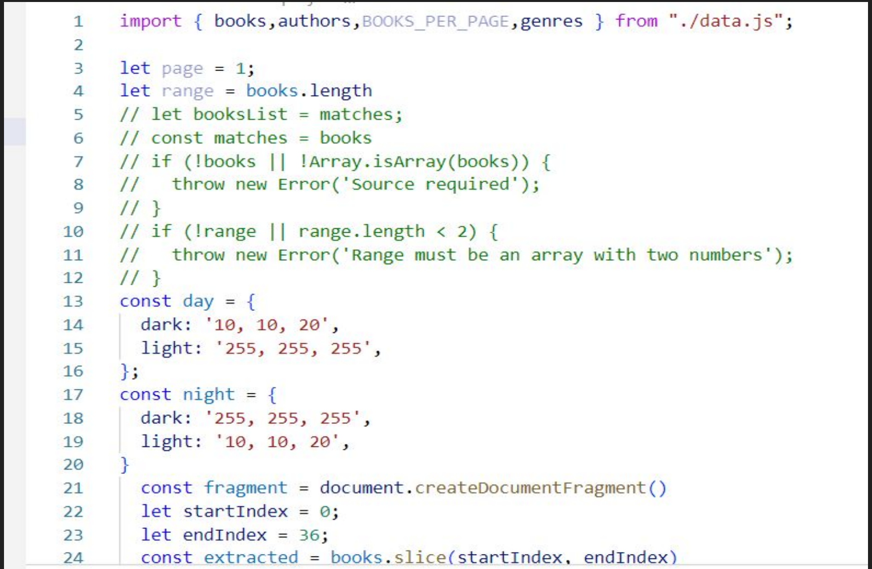
```
for ({ author, image, title, id }; extracted; i++) {
    const { author: authorId, id, image, title } = props

    element = document.createElement('button')
    element.classList = 'preview'
    element.setAttribute('data-preview', id)

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

        <div class="preview__info">
            <h3 class="preview__title">${title}</h3>
            <div class="preview__author">${authors[authorId]}</div>
        </div>
    `


    fragment.appendChild(element)
}
```

March 20XX

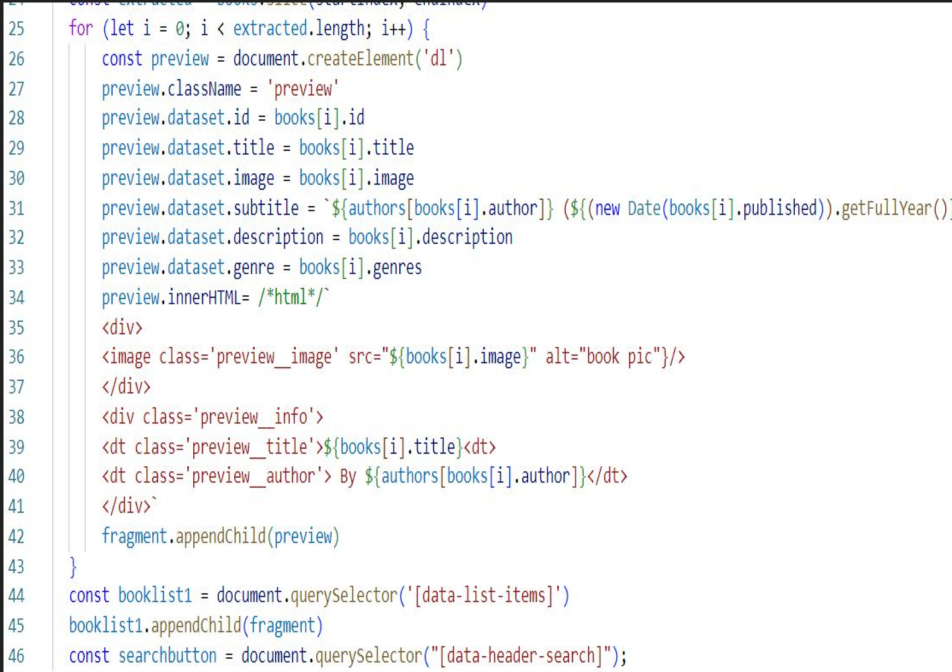


```
}

data-list-items.click() {
    pathArray = Array.from(~~event~~.path || ~~event~~.composedPath())
    active;

    for (node; pathArray; i++) {
        if active break;
        const previewId = node?.dataset?.preview

        for (const singleBook of books) {
            if (singleBook.id === id) active = singleBook
        }
    }

    if !active return
    data-list-active.open === true
    data-list-blur + data-list-image === active.image
    data-list-title === active.title

    data-list-subtitle === '${authors[active.author]} (${Date(active.published).year})'
    data-list-description === active.description
}
```

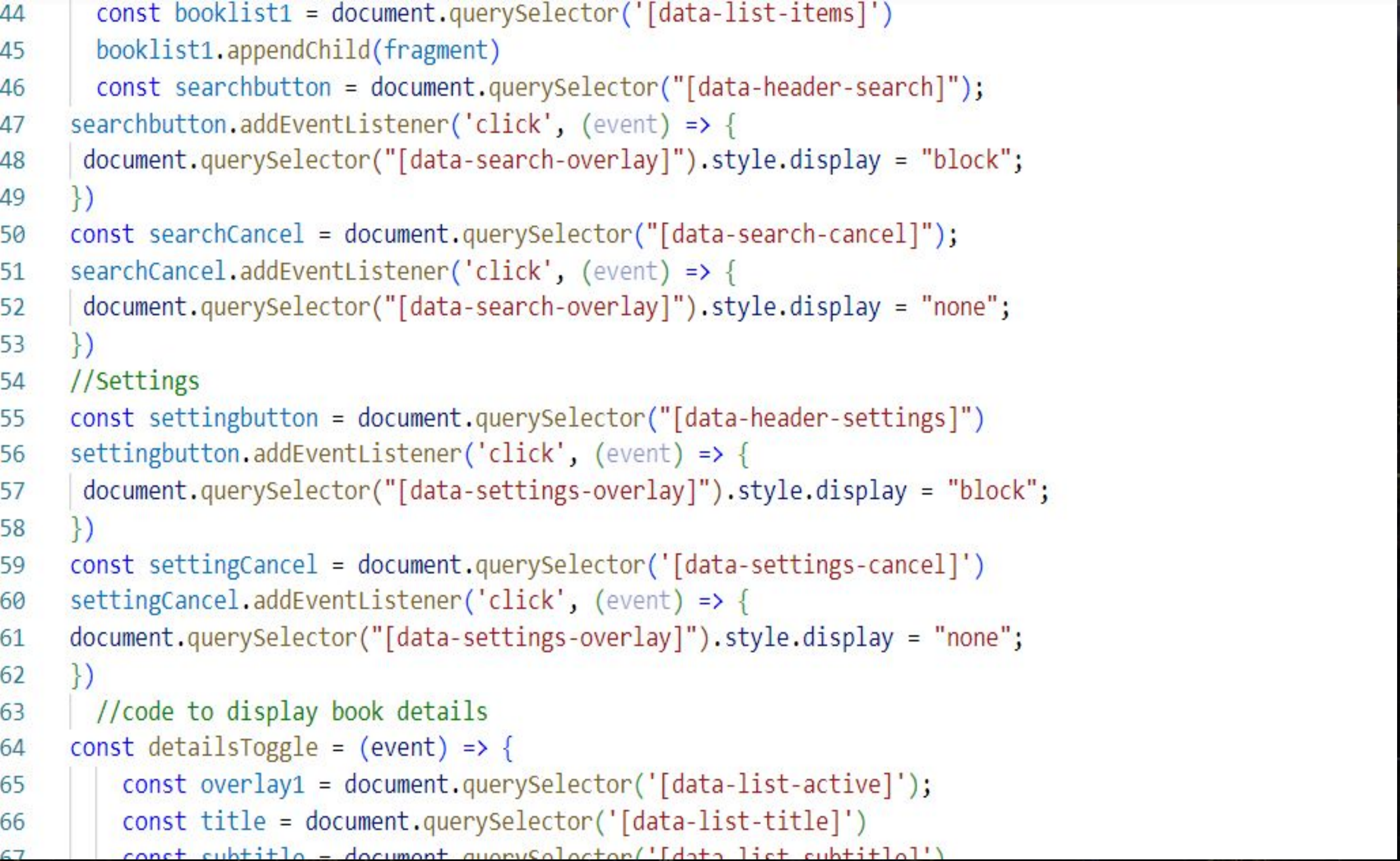
# HOH DID I FIXE ERRORS.



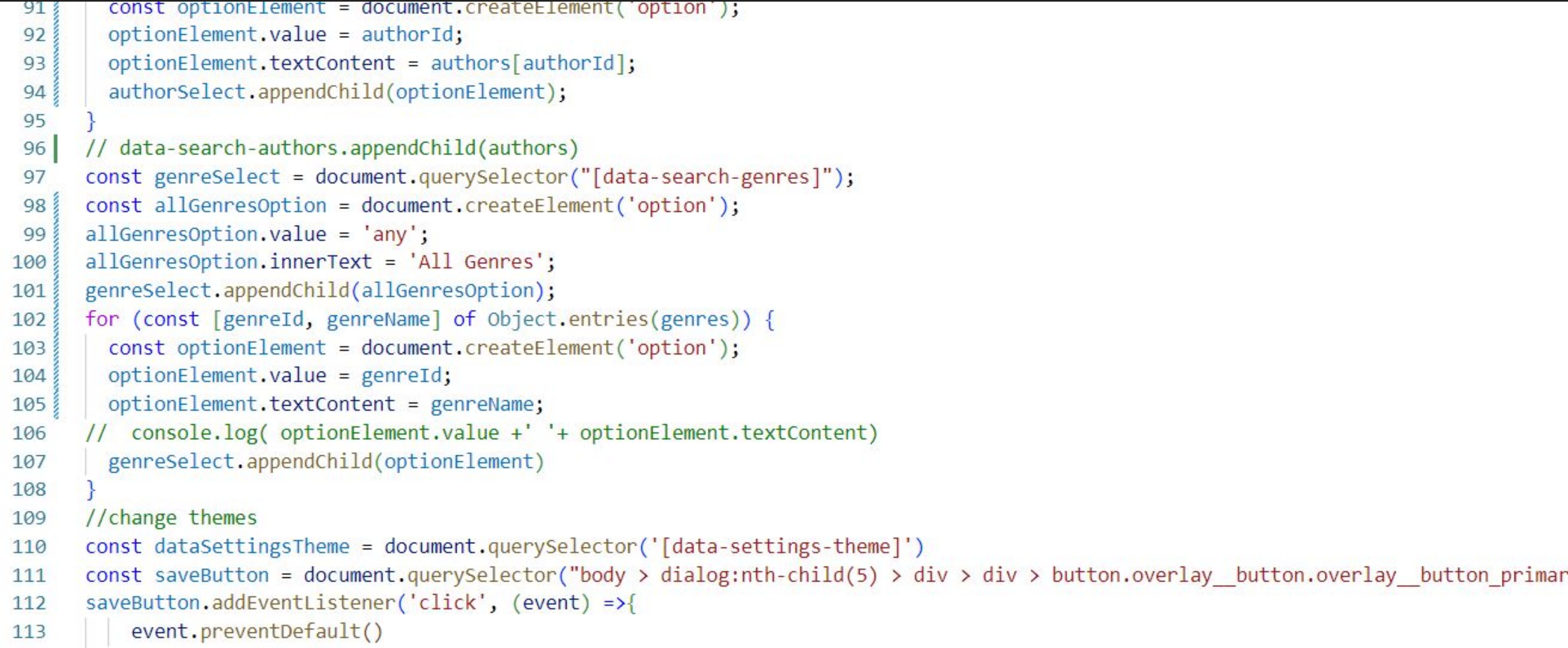
```javascript
import { books,authors,BOOKS_PER_PAGE,genres } from "./data.js";

let page = 1;
let range = books.length
// let booksList = matches;
// const matches = books
// if (!books || !Array.isArray(books)) {
//   throw new Error('Source required');
// }
// if (!range || range.length < 2) {
//   throw new Error('Range must be an array with two numbers');
// }
const day = {
  dark: '10, 10, 20',
  light: '255, 255, 255',
};
const night = {
  dark: '255, 255, 255',
  light: '10, 10, 20',
}
  const fragment = document.createDocumentFragment()
  let startIndex = 0;
  let endIndex = 36;
  const extracted = books.slice(startIndex, endIndex)
```



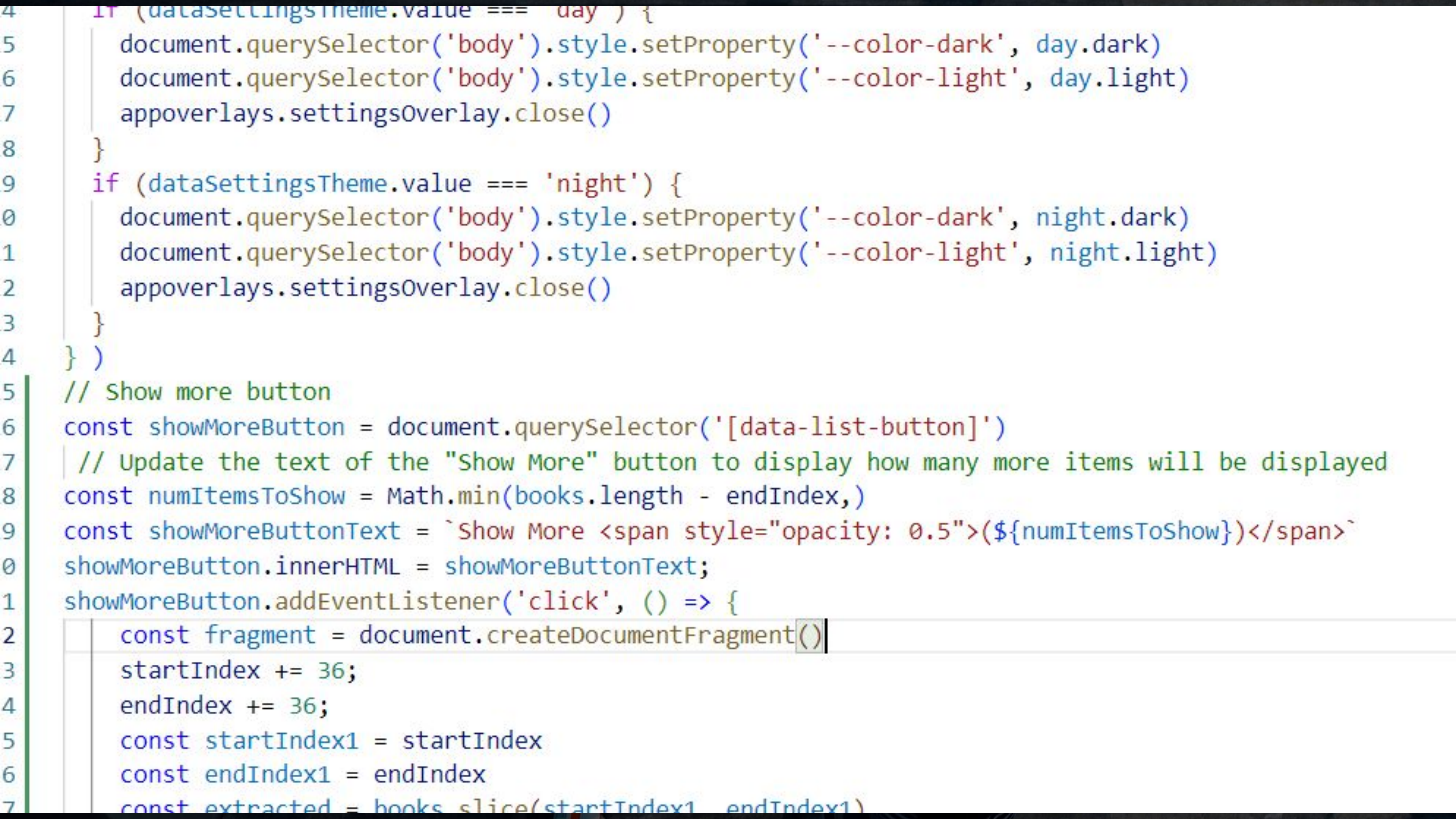
```javascript
for (let i = 0; i < extracted.length; i++) {
    const preview = document.createElement('dl')
    preview.className = 'preview'
    preview.dataset.id = books[i].id
    preview.dataset.title = books[i].title
    preview.dataset.image = books[i].image
    preview.dataset.subtitle = `${authors[books[i].author]} (${(new Date(books[i].published)).getFullYear()
    preview.dataset.description = books[i].description
    preview.dataset.genre = books[i].genres
    preview.innerHTML= /*html*/`
    <div>
    <image class='preview__image' src="${books[i].image}" alt="book pic"}/>
    </div>
    <div class='preview__info'>
    <dt class='preview__title'>${books[i].title}<dt>
    <dt class='preview__author'> By ${authors[books[i].author]}</dt>
    </div>`
    fragment.appendChild(preview)
}
const booklist1 = document.querySelector('[data-list-items]')
booklist1.appendChild(fragment)
const searchbutton = document.querySelector("[data-header-search]");
```



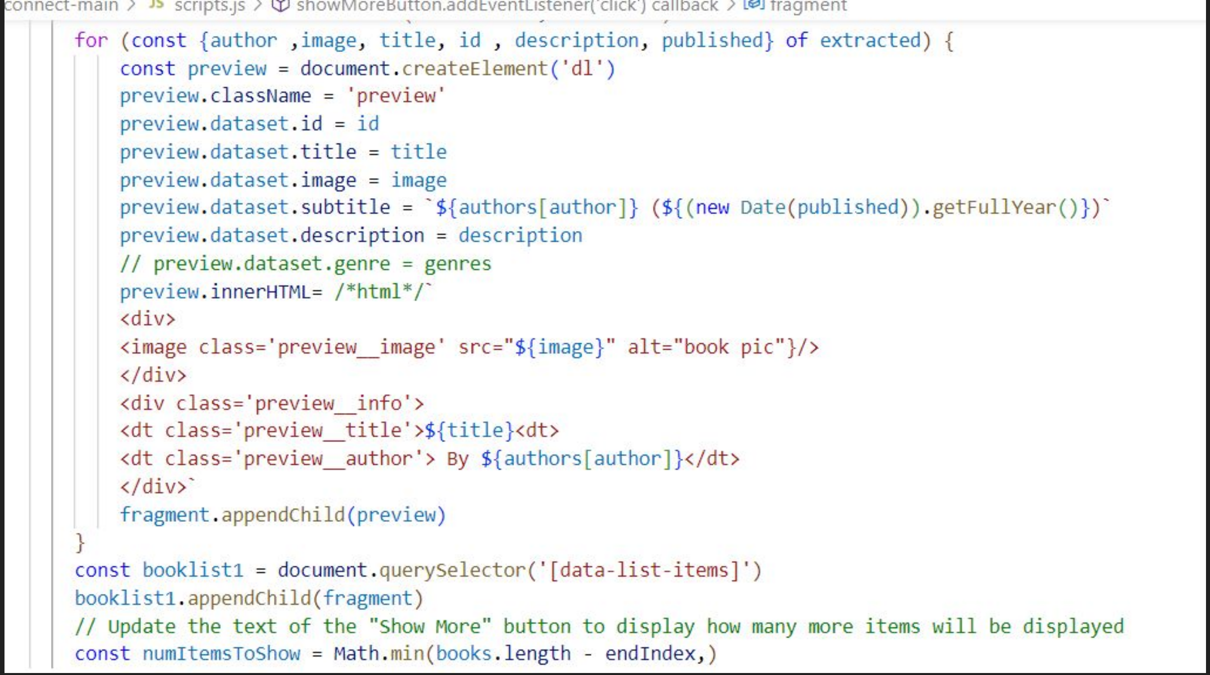
```
  const booklist1 = document.querySelector('[data-list-items]')
  booklist1.appendChild(fragment)
  const searchbutton = document.querySelector("[data-header-search]");
searchbutton.addEventListener('click', (event) => {
 document.querySelector("[data-search-overlay]").style.display = "block";
})
const searchCancel = document.querySelector("[data-search-cancel]");
searchCancel.addEventListener('click', (event) => {
 document.querySelector("[data-search-overlay]").style.display = "none";
})
//Settings
const settingbutton = document.querySelector("[data-header-settings]")
settingbutton.addEventListener('click', (event) => {
 document.querySelector("[data-settings-overlay]").style.display = "block";
})
const settingCancel = document.querySelector('[data-settings-cancel]')
settingCancel.addEventListener('click', (event) => {
document.querySelector("[data-settings-overlay]").style.display = "none";
})
  //code to display book details
const detailsToggle = (event) => {
    const overlay1 = document.querySelector('[data-list-active]');
    const title = document.querySelector('[data-list-title]')
    const subtitle = document.querySelector('[data-list-subtitle]')
```



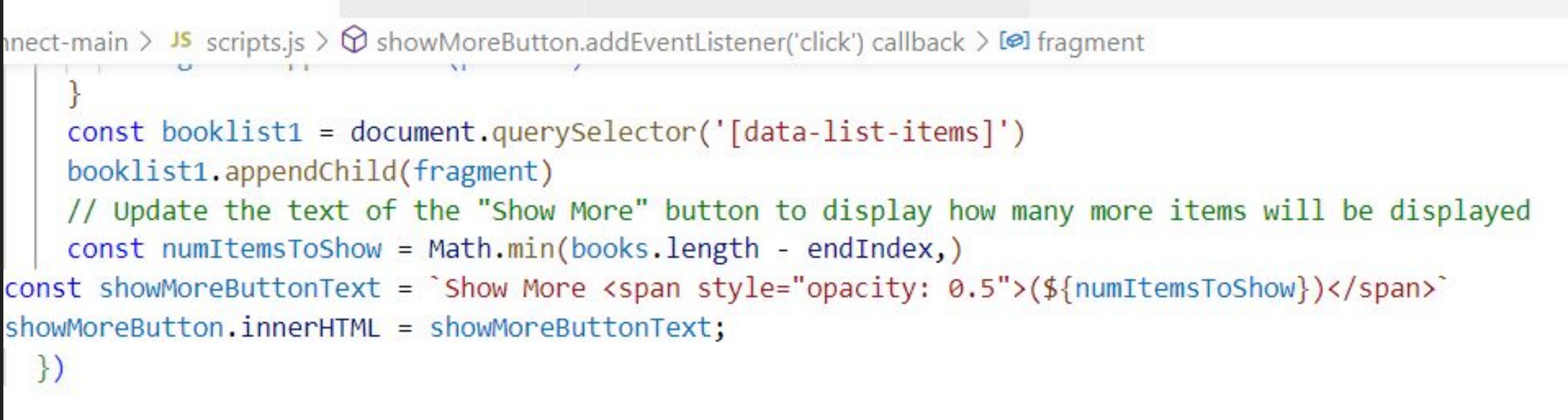
```
  const optionElement = document.createElement('option');
  optionElement.value = authorId;
  optionElement.textContent = authors[authorId];
  authorSelect.appendChild(optionElement);
}
// data-search-authors.appendChild(authors)
const genreSelect = document.querySelector("[data-search-genres]");
const allGenresOption = document.createElement('option');
allGenresOption.value = 'any';
allGenresOption.innerText = 'All Genres';
genreSelect.appendChild(allGenresOption);
for (const [genreId, genreName] of Object.entries(genres)) {
  const optionElement = document.createElement('option');
  optionElement.value = genreId;
  optionElement.textContent = genreName;
//  console.log( optionElement.value +' '+ optionElement.textContent)
  genreSelect.appendChild(optionElement)
}
//change themes
const dataSettingsTheme = document.querySelector('[data-settings-theme]')
const saveButton = document.querySelector("body > dialog:nth-child(5) > div > div > button.overlay__button.overlay__button_primar
saveButton.addEventListener('click', (event) =>{
    event.preventDefault()
```



```
  if (dataSettingsTheme.value === 'day') {
    document.querySelector('body').style.setProperty('--color-dark', day.dark)
    document.querySelector('body').style.setProperty('--color-light', day.light)
    appoverlays.settingsOverlay.close()
  }
  if (dataSettingsTheme.value === 'night') {
    document.querySelector('body').style.setProperty('--color-dark', night.dark)
    document.querySelector('body').style.setProperty('--color-light', night.light)
    appoverlays.settingsOverlay.close()
  }
} )
// Show more button
const showMoreButton = document.querySelector('[data-list-button]')
 // Update the text of the "Show More" button to display how many more items will be displayed
const numItemsToShow = Math.min(books.length - endIndex,)
const showMoreButtonText = `Show More <span style="opacity: 0.5">(${numItemsToShow})</span>`
showMoreButton.innerHTML = showMoreButtonText;
showMoreButton.addEventListener('click', () => {
    const fragment = document.createDocumentFragment()
    startIndex += 36;
    endIndex += 36;
    const startIndex1 = startIndex
    const endIndex1 = endIndex
    const extracted = books.slice(startIndex1, endIndex1)
```


connect-main > JS scripts.js > showMoreButton.addEventListener('click') callback > fragment

```
    for (const {author ,image, title, id , description, published} of extracted) {
        const preview = document.createElement('dl')
        preview.className = 'preview'
        preview.dataset.id = id
        preview.dataset.title = title
        preview.dataset.image = image
        preview.dataset.subtitle = `${authors[author]} (${(new Date(published)).getFullYear()})`
        preview.dataset.description = description
        // preview.dataset.genre = genres
        preview.innerHTML= /*html*/`
        <div>
        <image class='preview__image' src="${image}" alt="book pic"}/>
        </div>
        <div class='preview__info'>
        <dt class='preview__title'>${title}<dt>
        <dt class='preview__author'> By ${authors[author]}</dt>
        </div>`
        fragment.appendChild(preview)
    }
    const booklist1 = document.querySelector('[data-list-items]')
    booklist1.appendChild(fragment)
    // Update the text of the "Show More" button to display how many more items will be displayed
    const numItemsToShow = Math.min(books.length - endIndex,)
```

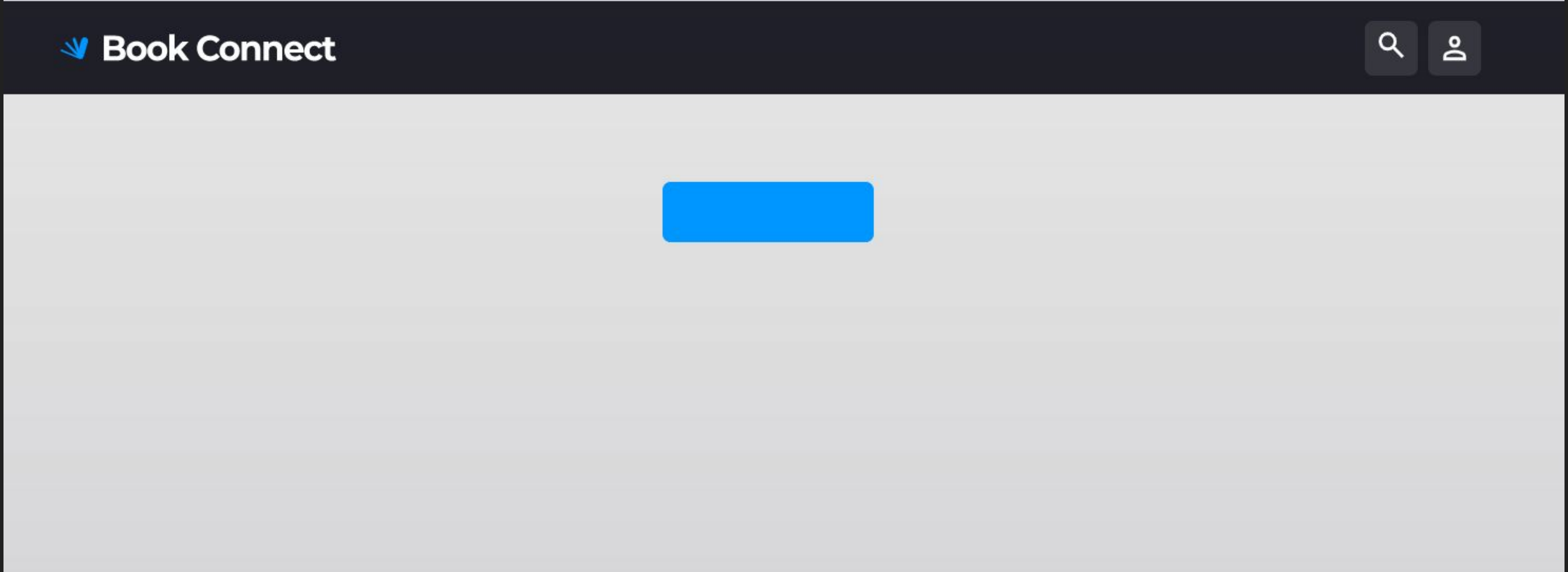

nect-main > JS scripts.js > showMoreButton.addEventListener('click') callback > fragment

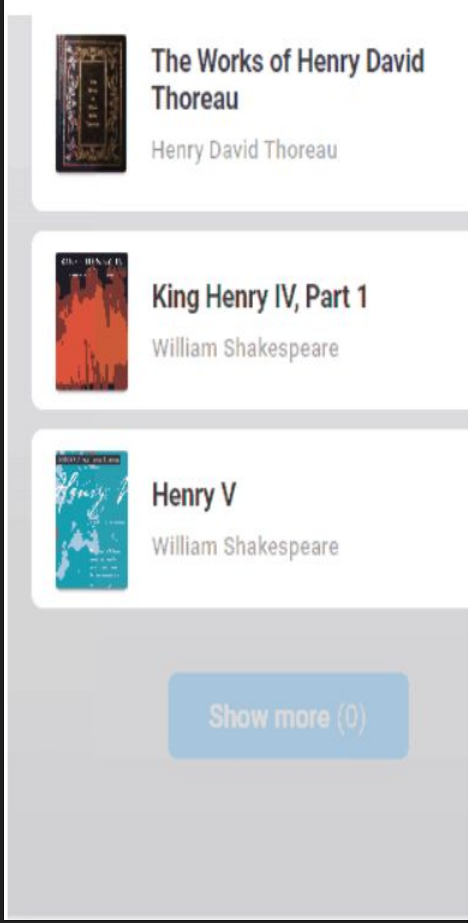
```
    }
    const booklist1 = document.querySelector('[data-list-items]')
    booklist1.appendChild(fragment)
    // Update the text of the "Show More" button to display how many more items will be displayed
    const numItemsToShow = Math.min(books.length - endIndex,)
const showMoreButtonText = `Show More <span style="opacity: 0.5">(${numItemsToShow})</span>`
showMoreButton.innerHTML = showMoreButtonText;
  })
```

# User Stories

1. As a user, I want to view a list of book previews, by title and author, so that I can discover new books to read.
2. As a user, I want an image associated with all book previews so that I can recognize a book by the cover even if I forgot the name.
3. As a user, I want to have the option of reading a summary of the book so that I can decide whether I want to read it.
4. As a user, I want to have the option of seeing the date that a book was published so that I can determine how easy it is to obtain second-hand.
5. As a user, I want to find books based on specific text phrases so that I don’t need to remember the entire title of a book.
6. As a user, I want to filter books by author so that I can find books to read by authors that I enjoy.
7. As a user, I want to filter books by genre so that I can find books to read in genres that I enjoy.
8. As a user, I want to toggle between dark and light modes so that I can use the app comfortably at night.

# END PRODUCT.



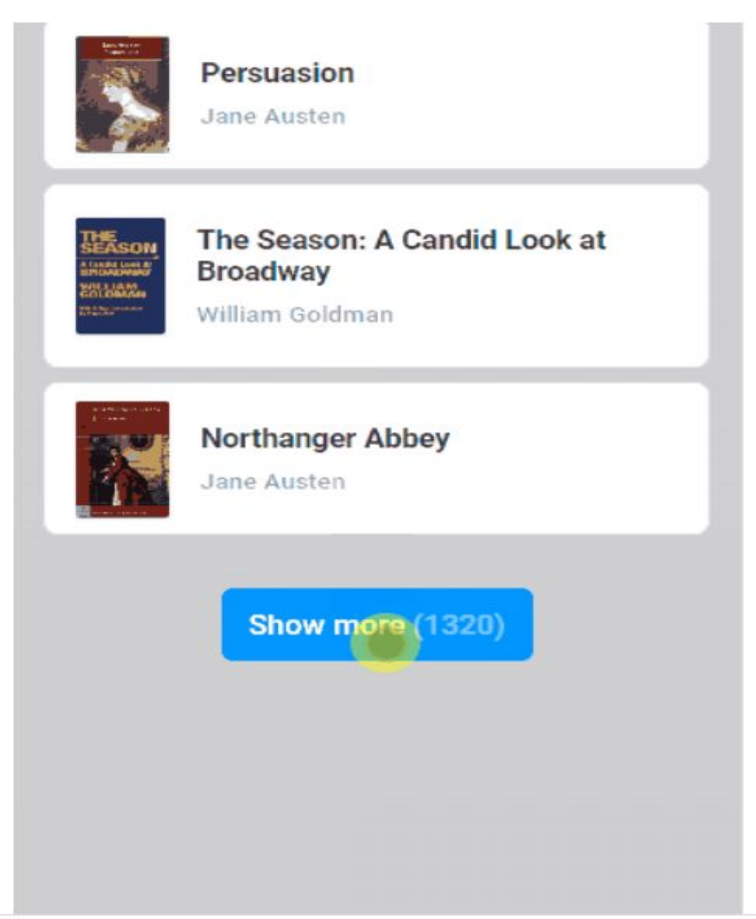


The Works of Henry David Thoreau
Henry David Thoreau

King Henry IV, Part 1
William Shakespeare

Henry V
William Shakespeare

Show more (0)

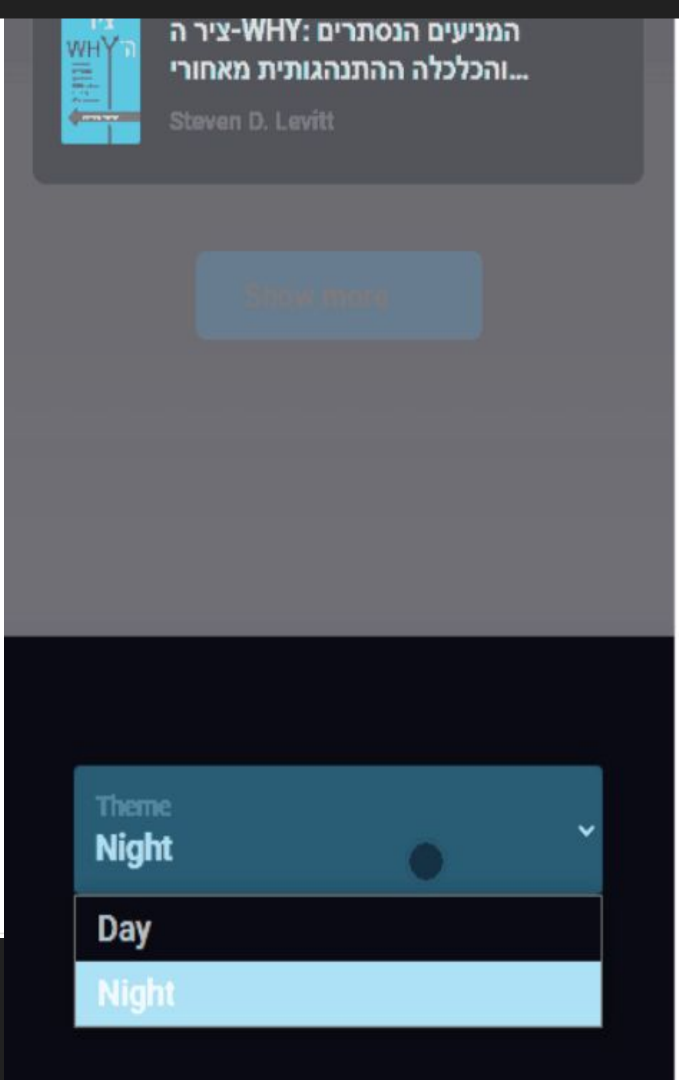


Persuasion
Jane Austen

The Season: A Candid Look at Broadway
William Goldman

Northanger Abbey
Jane Austen

Show more (1320)

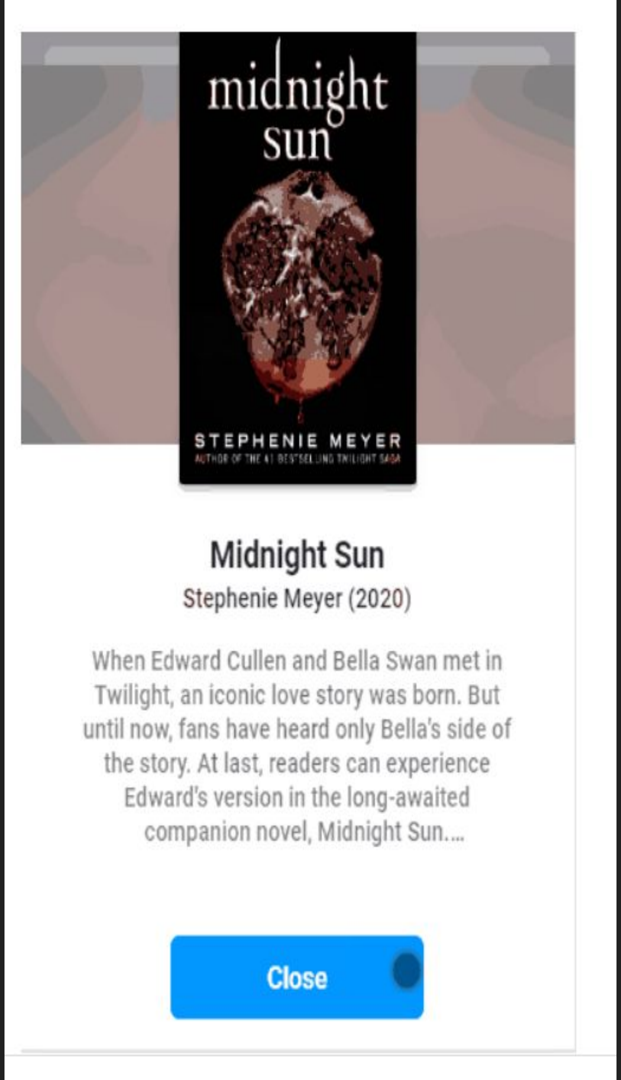


ציר ה-WHY: המניעים הנסתרים והכלכלה ההתנהגותית מאחורי...
Steven D. Levitt

Show more

Theme
Night

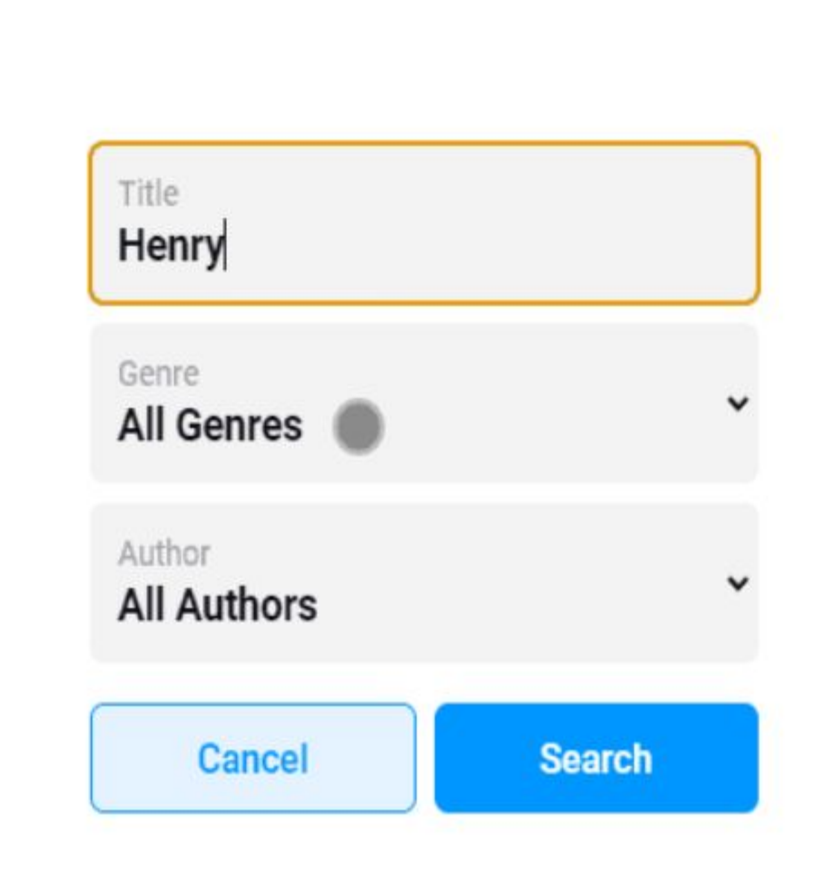
Day
Night



**Midnight Sun**

Stephenie Meyer (2020)

When Edward Cullen and Bella Swan met in Twilight, an iconic love story was born. But until now, fans have heard only Bella's side of the story. At last, readers can experience Edward's version in the long-awaited companion novel, Midnight Sun....



Title: Henry

Genre: All Genres

Author: All Authors