

# Core Lab Interaction

## Homework 5

### Your assignment is to apply CSS selectors to an IRL bookcase

[https://www.w3schools.com/cssref/css\\_selectors.asp](https://www.w3schools.com/cssref/css_selectors.asp)

Your final product will be two parts:

1. a map of your bookcase
2. a collection of (5) selectors (your choice), each with the **query** and **result**

*this is a creative assignment, the only code syntax that matters is the CSS selector syntax*

## 1. map of your bookcase

- no need to map your entire bookcase, only the items that get used
- for consistency, where applicable, let's use titles (book titles) as ids
- an abbreviated example looks like this (notice I divided the bookcase into shelves):

```
<bookcase>
  <shelf>
    <book id="The Unbearable Lightness of Being"></book>
    <book id="Between the World and Me"></book>
  </shelf>
  <shelf>
    <book id="Bluets"></book>
    <book id="Eros, the Bittersweet"></book>
  </shelf>
</bookcase>
```

## 2. selectors

*example (this selects all books where "children" appears somewhere in the title attribute):*

**query:** [title~="children"]

**result:** Midnight's Children, Children of Men

## elements and attributes

- Quick review: an **element** is “body”, “div”, “a”, and an **attribute** is “background-color”, “padding”. In this exercise **elements** become “bookcase”, “shelf”, “book” and **attributes** become “cover-color”, “book-thickness”, “hardcover/softcover”, “language”, etc..
- elements and attributes are here for you to invent, including how attributes work, like “book-thickness” can be about page count: “820pages”, or it can be more abstract: “thick”, “medium”
- For the map I’m only requiring that you list elements, **for brevity, leave out attributes**.
  - Pretend there is an imaginary stylesheet elsewhere that has an entry for each book:

```
#The Unbearable Lightness of Being{
  language: english;
  font-style: serif;
  my-rating: 5stars;
  friend-who-recommended-it-to-me: Freddie;
  .....
}
```

## helpful things

- a successful assignment will demonstrate a **variety** of types of selectors, with query results giving multiple entries (not just 1 book each time)
- if you don’t own books, no worries, google image search “bookcase”. Or, if you own things that are equally sortable in containers this should be no problem as well. Email me in this case.
- notice that the map establishes a top-down order (it’s up to you to decide right-to-left or left-to-right) because many of the selectors depend on order (the first **<book>** under a **<shelf>**)

*if you give your shelves ids, new selectors become available:*

```
<bookcase>
  <shelf id="top-shelf">
    <book></book>
```

*elements other than <book> are great! Just make sure they can share attributes*

```
<folder></folder>      <diary></diary>      <dvd></dvd>
```

*this is fun: imagine books contain <p> themselves, it’s possible to go INSIDE of a book*

```
#AChristmasCarol::first-line  (“MARLEY was dead: to begin with.”)
```

*the concept of “bookcase” can be very abstract*

```
<floor>
  <pile id="big-stack-in-the-corner">
    <book></book>
    <book></book>
  </pile>
  <pile id="bedside-pile">
    <book></book>
  </pile>
</floor>
```