

CIS 3515 Assignment 7

Instructions: Create a new application called **BookShelf**, that will take shape over the next few labs.

Your application will incorporate a dynamic (responsive) UI, that will display a **single fragment** at a time if the application is being run **on a small screen (cell phone) in portrait mode**, but must display **two fragments simultaneously** if being run **on a larger device (tablet)** or if being run **in landscape mode, regardless of screen size** (You can create a virtual tablet device to test this configuration).

1. In your activity:
 1. Use a **HashMap** to represent a book. Your hashmap should have two values: **title**, and **author**
 2. Create a set of arbitrary book objects and store them in an **ArrayList<HashMap>**. Pick some random book titles, real or made up – you should have at least 10 books. You can fetch those books from a string resource, or simply create a helper method or helper class to generate the books for you – all that matters is that in the end you have an ArrayList object, that contains 10 HashMap objects, where each hashmap has a title and author value.
2. Create 3 layout files for your activity for the following scenarios:
 1. portrait: this will contain a single display container (this could be the default)
 2. landscape: this will contain 2 display containers
 3. Tablet: this will contain 2 display containers
3. Create two fragments: BookListFragment, and BookDetailsFragment.
 1. BookListFragment will contain a ListView that, when provided with an arraylist of books, will display the book title and book author for each book in its ListView. The title and author should each be displayed in their own TextView – that is, each item in the ListView will contain 2 TextViews, one showing the book title, and the other showing the book author. When the user clicks one of the books in the ListView, the fragment should invoke a method in its parent with the **index** of the book that was clicked. The fragment should have a factory method (**newInstance(ArrayList<HashMap> books)**) that creates a fragment using a list of books to set up the fragment's initial state
 2. BookDetailsFragment will have the ability to display a book title and author when provided with a book object. It will have a single public API method, **displayBook(HashMap book)**, which will display the book title in a TextView with a large font size, and the author in a TextView with a slightly smaller font size, whenever the method is invoked. Additionally it should have a factory method (**newInstance(HashMap book)**) that creates a fragment using a book to set up the fragment's initial state.
4. Your activity will then use the Master-Detail pattern when determining when to display a

certain fragment.

1. If the activity is using a layout that contains a single container (portrait mode on a small screen), it should display an instance of the BookListFragment at startup. When the user clicks a book from the BookListFragment, an instance of BookDetailsFragment should be launched to replace the BookListFragment, displaying information on the book that was clicked. The user should then be able to reverse this operation and select a new book.
 2. If the activity is using a layout that contains two containers, then one container is used to display a **BookListFragment**, and the other to display a **BookDetailsFragment**. Clicking a book title in the BookListFragment should have the BooksDetailsFragment display the book title, but it should not load a new instance for each click. Instead, a single instance of BookDetailsFragment should be loaded, and it should change the book that is displayed whenever the user clicks a new book from the BookListFragment.
5. **Push your project to GitHub and upload your link to Canvas.**

Rubric

Application has configuration for small-portrait, large portrait, and landscape orientations	20%
Both BookListFragment and BookDetailsFragment have factory methods that can create new fragment instances	20%
BookListFragment has ListView that shows both title and author in their own TextViews	10%
BookListFragment properly communicates selected book to parent using arraylist index	10%
Proper master-detail implementation when in small-portrait mode (New instance of BookDetailsFragment being created when book is clicked)	20%
Proper master-detail implementation when in large-portrait or landscape mode (Single instance of BookDetailsFragment being updated when book is clicked)	20%