

ITC 162 Grading Criteria (please read carefully and ask for clarification if needed).

Does the code run? – 50 points

- No more than 20 points will be given if the code doesn't run.

Correctness of code – 40 points

- Does the code perform per requirements as specified in the assignment?
- Does the program produce the correct prompts to the user?
- Does the program calculate the correct values?
- Does the program display the results correctly?
 - * As applicable for the assignment

Clarity of code – 10 points

- Does the program use good names for variables?
- Is the code commented and spaced so that it is easy to read and understand? Use blank lines as appropriate.

Incorrect submission

- Your assignment will not be graded if it is not submitted in the right place on Canvas or if the submission is not in the file format specified for the assignment.
- You will be allowed to resubmit, however you might lose points due to late submission.

Convert activities to fragments

1. In this exercise, you'll convert the News Reader app presented in chapter 10 so it uses fragments to allow this app to work better on devices with large screens. Although this app uses some new coding skills that you haven't been introduced to yet, you can ignore them for now and focus on converting this app's activities to fragments.

Test the app

1. Start Android Studio and open the project named Week 7/HMch09_ex4_NewsReader.

Here is also a link to the project

<https://drive.google.com/open?id=0B9lxzuK2ctFXM0FEX1N4LTRleHc>

2. Run the app and test it. Note that it uses two activities to allow you to view a list of items or a single item.

Convert the Items activity so it uses a fragment

1. Make a copy of the activity_items layout and name it fragment_items.
2. Add the source code for a fragment named ItemsFragment. You can move most of the code that provides the functionality from the ItemsActivity class. To get this code to work

correctly, you can use the `getActivity` method instead of the `this` keyword whenever you need to get a reference to the current activity.

3. Modify the `activity_items` layout so it displays the `fragment_items` layout.
4. Modify the `ItemsActivity` class so it only displays the `activity_items` layout.
5. Test the app to make sure it works. At this point, it should work the same as it did in step 2.

Convert the `Item` activity so it uses a fragment

1. Make a copy of the `activity_item` layout and name it `fragment_item`.
2. Add the source code for a fragment named `ItemFragment`. You can move most of the code that provides the functionality from the `ItemActivity` class. To get this code to work correctly, you can use the `getActivity` method instead of the `this` keyword whenever you need to get a reference to the current activity.
3. Modify the `activity_item` layout so it displays the `fragment_item` layout.
4. Modify the `ItemActivity` class so it only displays the `activity_item` layout.
5. Test the app to make sure it works. At this point, it should work the same as it did in step 2.