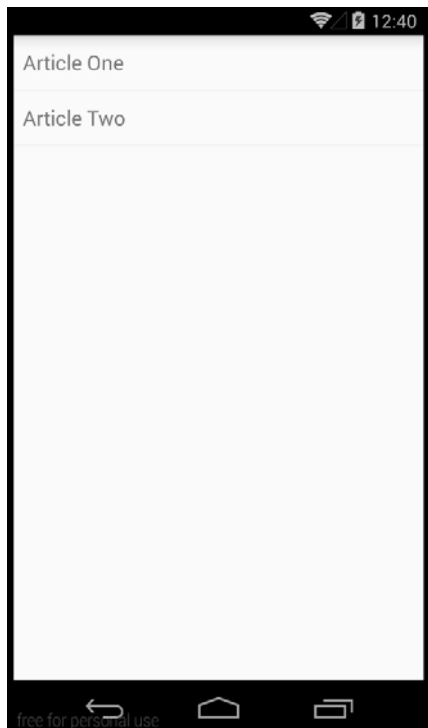




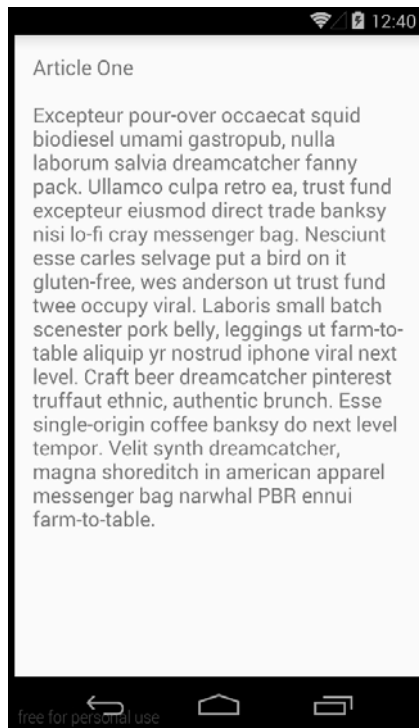
Assignment Background:

The focus of this lab is to create a lab which will replace the communication between fragments with Singleton logic. The data entered by the user should be stored, so if the app is closed it will reopen and use the past data that was entered. Note, the base of this lab was created as *DemoFragmentBasics*

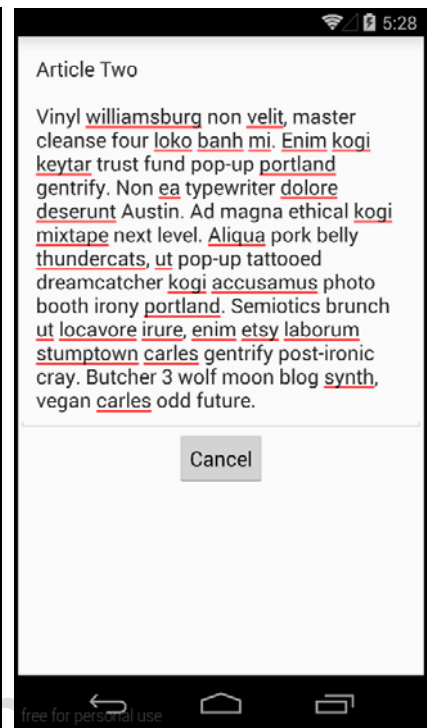
- 1) Using the Lab2_1 starting application (DemoFragmentBasics from 152-189 Android Apps) as a base, create Android application project and name it:
 - a) Application Name: *YourUsernameLab2_1*
 - b) Project Name: *YourUsernameLab2_1*
 - c) Package Name: *com.yourLastNameFirstName.yourUsernameLab2_1*
 - d) Use all other appropriate values to finish out creation of project
- 2) Modify the application
 - a) Use Singleton to store information regarding the Ipsum information
 - i) Create objects to store the Ipsum headline and article information and
 - (1) Include logic in the Singleton to load the starting information
 - (2) Modify the Ipsum class so it no longer has hardcoded String arrays
 - (3) Store the Ipsum objects in the Singleton
 - b) Modify the MainActivity
 - i) You will no longer require passing information thru callbacks (although you could) for the Activity to pass the information to the fragment.
 - ii) Instead allow the fragments to pull or put the information it needs to from the Singleton.
 - c) Modify the HeadlineFragment
 - i) Pull the current Ipsum object information from the Singleton
 - ii) The Fragment will now update the Singleton to store the position instead of passing it to the *onArticleSelected* method
 - iii) Although it would be logical to also include logic to add new headlines, do not add logic to do so (you may do so for a lab challenge)
 - d) Modify the ArticleFragment
 - i) Allow the user to edit the article information
 - ii) Add a cancel button to allow the user to not save any changes
 - iii) When the user pressed the back button, save any changes
- 3) When the app is closed, it should reopen with all previously stored data reloaded



Original Headline (Fragment 1)



Original Article (Fragment 2)



Modified Article (Fragment 2)

- 4) Follow Java coding guidelines and standards
 - a) Java code should be written with reuse in mind
 - b) Use naming conventions (m prefix, s prefix, etc)
 - c) Code should include javadoc comments for each method
- 5) Test all functionality of the app
 - a) Contact can still be added and updated with the including navigation logic
 - b) Error logic still functions when adding/updating contact
 - c) Rotation display ActionBar properly
 - d) Allow the User to rotate the device portrait and landscape without losing position or data
 - i) All functionality should remain regardless of orientation
 - ii) Be sure that all buttons and functionality is visible on the screen preventing the user from having to scroll to find the buttons or information