MAP524/DPS924 Lab #3

Learning Focus

1. Code Android views and event handling.
2. Work with ArrayAdapter.
3. Work with XML tags in res/layout.
4. Work with string resources and array resources.

Notes

1. Lab #3 is due by the start of your next lab period.
2. You should give a demo to the professor in the computer lab before leaving to get feedback.
3. You are allowed to use Android resources from the Internet but you must add comments to your code (and/or xml files) saying where you get that code.
4. You are welcome to complete the starter code we develop together or start from scratch.

Have a look at:
https://developer.android.com/reference/android/widget/AutoCompleteTextView.html
You'll notice from the code snippet there that AutoCompleteTextView also uses ArrayAdapter as we demonstrated with ListView. Use the same pattern for populating the AutoCompleteTextView's "vocabulary".

At the start of the lab I'll demonstrate how to change the default RelativeLayout into a LinearLayout (vertical orientation), as well as demonstrate adding Views programmatically using the Java API to a LinearLayout.
If you want to get started right away you can research the documentation for how to do this yourself.

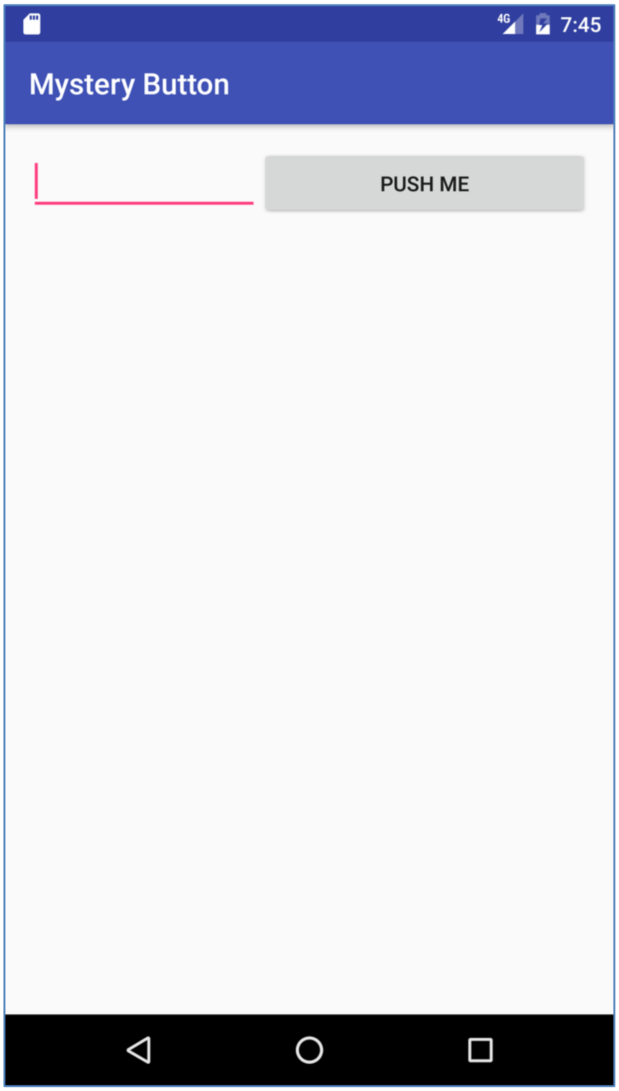
Problem Description

You are asked to develop an Android app according to the following functional specification:

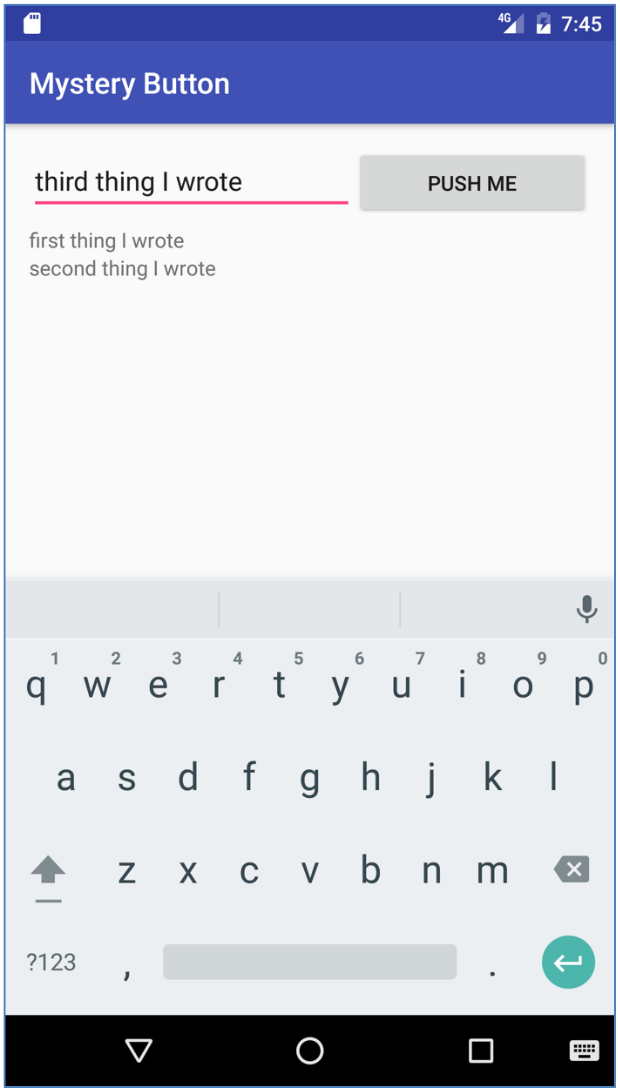
1. The title of the app is "Mystery Button." It should appear on the action bar/app bar. The activity has all views as children of a LinearLayout (using `android:orientation="vertical"`). The first view added to At the top of the LinearLayout is a button that says "press me", and BESIDE it is an AutoCompleteTextView. Think about how to accomplish this (hint: you will need to add a child layout within the LinearLayout that makes up your main Activity).

When the button is pressed, and ONLY IF THE CONTENTS OF the AutoCompleteTextView is not empty, add another TextView to the LinearLayout that contains the current text contents of the AutoCompleteTextView.

Screenshot of what your app should look like BEFORE and AFTER the button is pressed a few times



(BEFORE)



third thing I wrote

first thing I wrote
second thing I wrote

(AFTER)

Easy, right? Below is list of the AutoCompleteTextView suggestions I want you to include. All vocabularies and their meanings/definitions should be stored in /res/values/strings.xml.
(see: https://developer.android.com/guide/topics/ui/controls/text.html at the bottom)

Vocabulary for the AutoCompleteTextView (definitions not shown – those are for you to add – one or two setences maximum for each definition)

1.activity
2. AVD
3. ART
4. Dalvik
5. intent
6. intent filter
7. explicit intent
8. implicit intent
9. LogCat
10. bundle

11. Gradle
12. Android Device Monitor
13. SDK manager
14. minSdkVersion

Note: You should be familiar with the meanings/definitions of those words/phrases by now. If not, good chance to review and/or research!

Question: What happens when you keep adding TextViews to your LinearLayout? Does the screen start scrolling? Why or why not? What could you do to enable scrolling?
(hint: see ScrollView from this week's notes at the bottom).

Submission Requirements

1. Take three screenshots of your app running in the emulator, one before adding a new TextView using the button, one with the AutoCompleteTextView showing suggestions, and one screenshot after one or more TextViews have been added to the "list".

When the lab is due, submit to Blackboard **AS SEPARATE ATTACHMENTS**:

a) the Word document that has i) all the screenshots and ii) the answers to the question(s) above.

b) a zip file of your Android project.
The name of your zip file should be "Lab3_<userid>.zip".

3. No re-submission will be accepted. Late penalty policy will be applied.

Optional – no hand in required:

Modify your project so that scrolling is enabled in your main Activity