Tutorial Week 4 – 31285 MAD

4%
Due before next class

Objectives:

- Get familiar with AsyncTask and ProgressDialog.
- Spinners

1. (5%) Create a new Android Studio project with the following characteristics:
    - Application name: Exercise 4
    - Package name: com.mad.exercise4
    - Min SDK: API15
    - Empty Activity
    - Activity name: MainActivity

2. (5%) Create a layout as shown in Fig. 1 (left), with one TextView (containing a joke) and 2 buttons at the bottom.

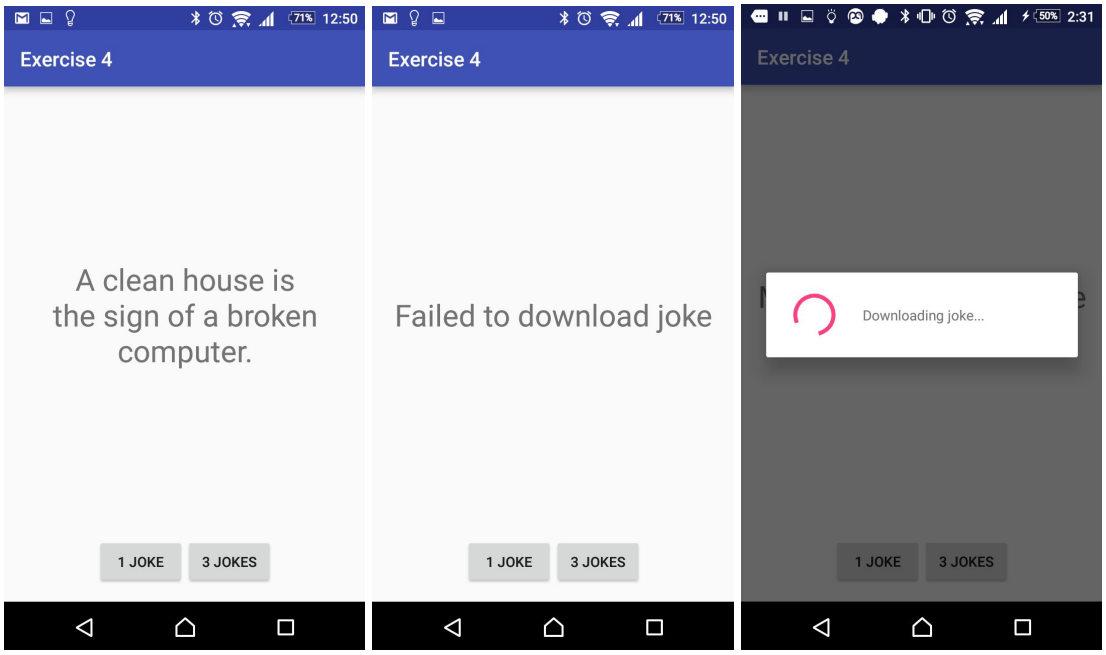

Fig. 1

Requirements: The buttons must be centred at the bottom and wrap content. The joke text must be 30sp, centred horizontally and vertically within the remaining space.

3. (10%) When you click the “1 JOKE” button you download and show a joke from the following web service: https://www.ryanheise.com/sarcastic.cgi

    Use the following technique to download the joke:

```
// Open a connection to the web service
URL url = new URL("https://www.ryanheise.com/sarcastic.cgi");
URLConnection conn = url.openConnection();
```

```
// Obtain the input stream
BufferedReader in = new BufferedReader(new InputStreamReader(conn.getInputStream()));
// The joke is a one liner, so just read one line.
String joke = in.readLine();
// Close the connection
in.close();
```

This web service is not 100% robust and WILL occasionally fail. If it fails, the above code will throw an exception. You should handle this exception by showing an error message instead of the joke as shown in Fig. 1, centre.

Requirements: You must implement an OnClickListener in the Activity class declaration check the button id with view.getId() in the onClick implementation.

Hint: You must grant the **android.permission.INTERNET** permission in your manifest file!!

4. (20%) Since it can take a few seconds to download the joke, you may notice that the UI freezes while the user waits. Change the "1 JOKE" click handler so that it shows a ProgressDialog and downloads the joke in the background using an AsyncTask, as shown in Fig. 1, right.

   Check the following links for tutorials on the AsyncTask concept.

- https://developer.android.com/reference/android/os/AsyncTask.html
- http://www.vogella.com/tutorials/AndroidBackgroundProcessing/article.html
- http://programmerguru.com/android-tutorial/android-asynctask-example/

Requirements:

- You must use the AsyncTask construct to download the joke in the background thread and update the progress dialog in the UI thread.
- Create a private inner class class called Download1JokeAsyncTask which extends AsyncTask<Void, Void, **String**>. Override the methods onPreExecute, doInBackground and onPostExecute. Shortcut: ALT+Insert -> Override Methods… The **String** parameter represents the downloaded joke.
- Call the AsyncTask as follows: new Download1JokeAsyncTask().execute();
- The ProgressDialog must be indeterminate (no progress percentage shown, just a spinning indicator) and show the message "Downloading joke…".
- After the download is finished, dismiss the ProgressDialog and show the downloaded joke or error message if the download failed.

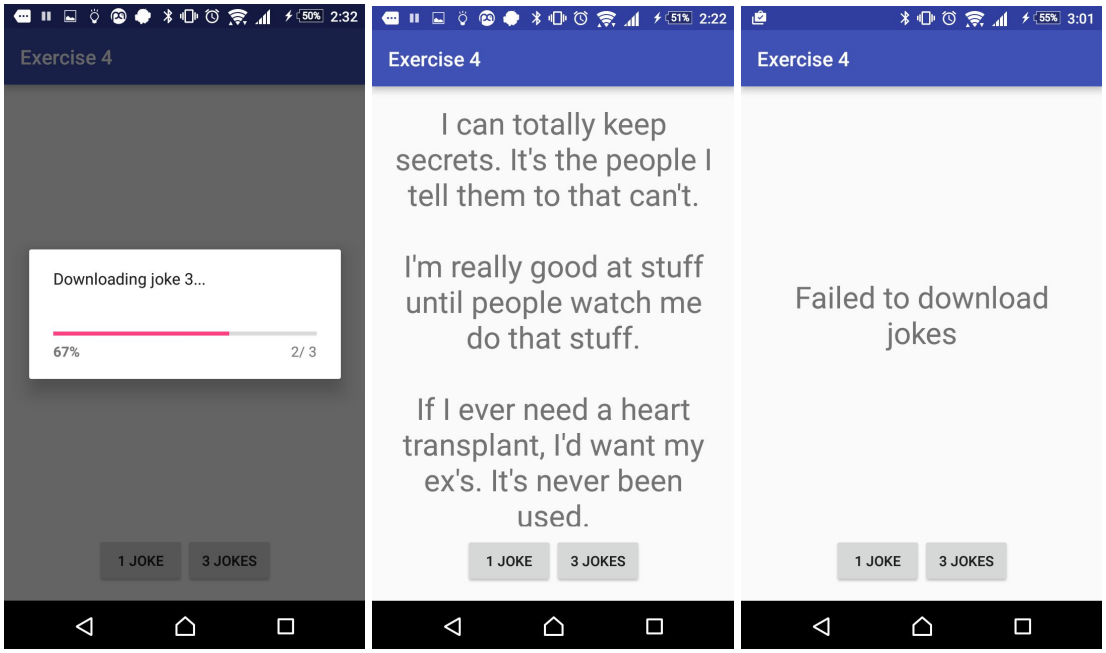

Fig. 2

5. (20%) When you click the “3 JOKES” button, you show a ProgressDialog as shown in Fig. 2, left, and download and show 3 jokes as shown in Fig. 2, centre. If any of the 3 jokes fails to download, you display an error as shown in Fig. 2, right.

   Requirements:

- Create another private inner class which extends AsyncTask<**Void, Integer, String[]**>. Override the methods onPreExecute, doInBackground, onProgressUpdate, onPostExecute. The **Integer** parameter represents the joke download progress (i.e. values 1, 2, 3). The **String[]** parameter represents the 3 downloaded jokes.
- Call AsyncTask as follows:
  new DownloadNJokesAsyncTask(*numJokes*).execute(); where *numJokes* is the number of jokes to download. It is a constructor parameter.
- The ProgressDialog must be determinate (i.e. show a horizontal progress bar, percentage complete and number downloaded out of 3).
- While the jokes are being downloaded, you should publish the progress so that the current progress is reflected in the ProgressDialog (See Fig 2, left).
- After all jokes have been downloaded, dismiss the progress dialog and display all jokes as shown in Fig 2, centre, or show a download error as shown in Fig 2, right.

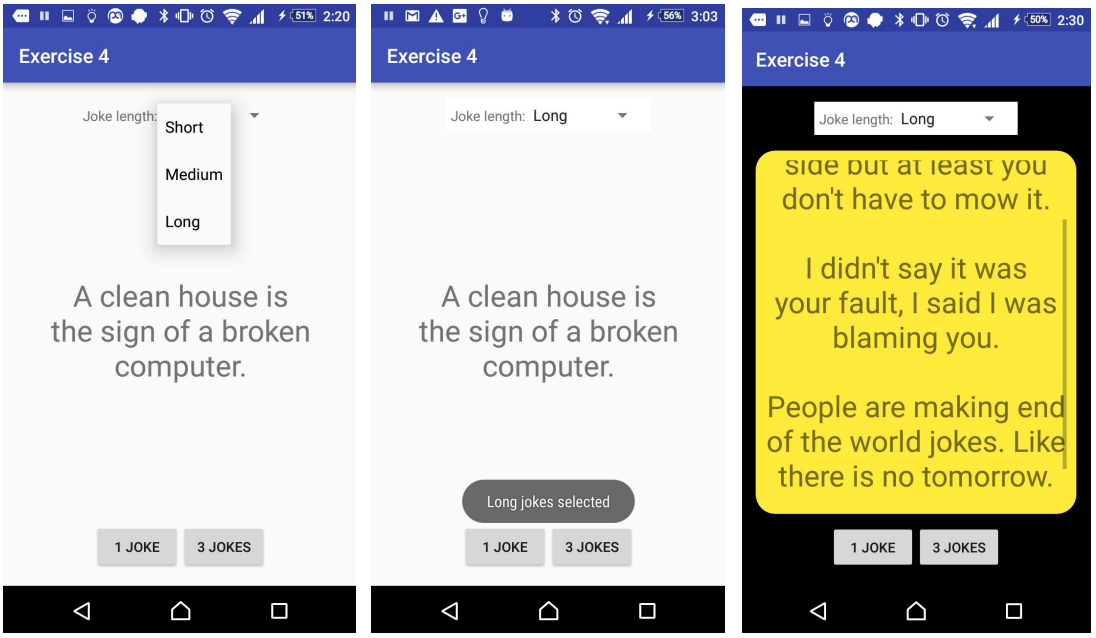

Fig. 3

6. (5%) Add a Spinner to your layout prefixed with the text “Joke length:” as shown in Fig. 3, left. Populate the spinner with the following entries: “Short”, “Medium”, “Long”. When you select an item you show a toast with the selected length as shown in Fig. 3, centre.

   Requirement:
   You must implement an OnItemSelectedListener in the Activity declaration.

7. (5%) All downloaded jokes should be filtered by the currently selected joke length in the Spinner. You can filter the jokes by passing a parameter to the web service. Note: the filter sets the maximum joke length. (e.g. “Long” allows long jokes but also short).
   - **https://www.ryanheise.com/sarcastic.cgi?len=Short**
   - **https://www.ryanheise.com/sarcastic.cgi?len=Medium**
   - **https://www.ryanheise.com/sarcastic.cgi?len=Long**

8. (10%) Modify your layout to look like Fig 3, right.

   Requirements:
   - Whole activity: Black background
   - Spinner section: white background, wrap content, centred, add padding/margin.
   - Joke TextView: yellow background, rounded borders, ScrollView.

     Hint: Create a custom drawable with a background colour and round borders. Google search keywords: android background drawable border

9. (5%) Eliminate all hard-coded strings and values from your code. User interface text should be defined in strings.xml or arrays.xml. Other hard-coded values that do not appear in the user interface should be defined as Java constants.

10. (15%) Make your code conform 100% to the android coding standards (variables, comments, exceptions). See: https://source.android.com/source/code-style.html

Homework: Watch the following YouTube videos about RecyclerView:

https://www.youtube.com/watch?v=Wq2o4EbM74k (part 1
https://www.youtube.com/watch?v=A5cUZlUnRUw (part 2, continues to part 3, 4)
https://www.youtube.com/watch?v=ZQd4qTWaL-o

Read the following RecyclerView tutorial:

http://www.androidhive.info/2016/01/android-working-with-recycler-view/

Watch videos in section 7 “Android App Development Essential Training” on LYNDA.com covering the topic of Menus.

Additional references for the older ListView widget (good for introducing the *adapter* concept):

http://www.edureka.co/blog/what-are-adapters-in-android/
http://www.vogella.com/tutorials/AndroidListView/article.html
http://www.tutorialspoint.com/android/android_list_view.htm