

Week 4 Overview

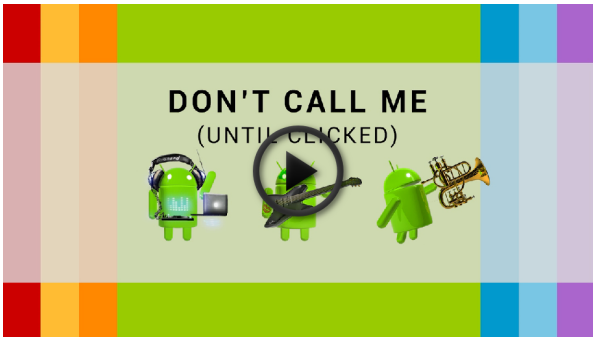


[Help](#)

An App for Emily

On this page:

[Video Lectures](#)
[Assignments](#)
[Time](#)
[Tips for Success](#)
[Getting and Giving Help](#)

Video Lectures

Video Lecture	Key Topics	Transcript	Video Download	SRT Caption File	
4.1. Don't Call Me (Until Clicked)					
 <p>(00:15:41)</p>	<ul style="list-style-type: none"> • Adding mp3 files as an Android raw resource • Creating a MediaPlayer object • Buttons • Calling methods when buttons are clicked • // Comment • Web browsing with Intent 		 (23.5 MB)		





Next Steps for 4.1

(Required) For everyone:

1. Create a new Android project and add an interesting sound or musical sample.
2. Insert an interesting mp3 file into your project's res/raw (there are plenty available just a web search awa

Video Lecture	Key Topics	Transcript	Video Download	SRT Caption File	
<p>file if necessary to ensure a valid resource filename.</p> <ol style="list-style-type: none"> Write the 2 lines of code in your onCreate method (plus an import statement) to create a new MediaPlayer music. Test it! Restart your device/emulator if sound output stops working. Create a clickable button. You will need to add the following code to your app's Activity: <pre>// Here's the openWebPage method. This belongs inside your Activity class the "public class ... {" line, and before the last "}" public void openWebPage(View v) { // Once you have this working, change the string to another web page String url = "http://developer.android.com/guide/topics/ui/controls/b Intent i = new Intent(Intent.ACTION_VIEW); i.setData(Uri.parse(url)); startActivity(i); } // Move import statements next to the others near the top of your Java fi import android.content.Intent; import android.net.Uri; import android.view.View;</pre> <p>(Optional) Proficient programmers may wish to learn more about the MediaPlayer. Note, later videos will intr state diagram.</p> <ul style="list-style-type: none"> MediaPlayer Guide 					

4.2. Introducing Activity Lifecycle

 <p>(00:04:52)</p>	<ul style="list-style-type: none"> Introducing the Activity lifecycle Responding to onResume and onPause events Declaring variables inside and outside of 		 <p>(8.7 MB)</p>	
---	--	---	---	---

Video Lecture	Key Topics	Transcript	Video Download	SRT Caption File	
	methods				

Next Steps for 4.2

(Everyone) Play with and learn to use buttons.

1. Add some more buttons. Fix it. Test it. Change it!
2. Make an app for you/your grandmother/your little sister/etc. that has several (e.g. 6) buttons to open their
3. (Optional) further reading: [Buttons](#)

(Everyone) Use the steps below to play with and learn about the Activity Lifecycle. Note that the quiz assume complete these steps.

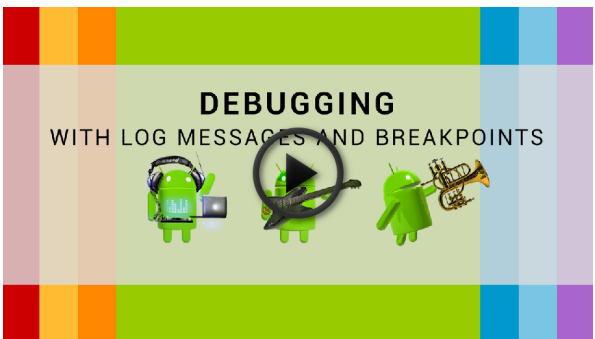



1. Change your code so that your url strings for each button are now defined outside of a method and inside instead. For example:

```
class ABC extends Activity{
    String url1 = "http://my-favorite-website/";
```
2. Use auto-complete (ctrl+space) to add onResume and onPause methods to your Activity. Add logging to your onCreate, onResume, and onPause methods. For example:

```
Log.e("Banana","onCreate!"); // Put this inside your onCreate method
Log.e("Banana","onResume"); // Put this inside your onResume method
Log.e("Banana","onPause!"); // You get the idea - Now test it!
```
3. Test your app on the emulator to find out when you see your 3 log messages. Telnet to localhost (port 5554) following conditions:
 - When the emulator receives a phone call. (Hint: gsm call 1234)
 - When the emulator receives an SMS message. (Hint: sms send 1234 Hello)
 - When you rotate the screen from portrait to landscape (and vice versa).
 - When you press the home button or switch to a different app.



Optional further reading about the [Activity Lifecycle](#).

4.3. Debugging with Log Messages and Breakpoints

Video Lecture	Key Topics	Transcript	Video Download	SRT Caption File	
 <p>DEBUGGING WITH LOG MESSAGES AND BREAKPOINTS</p> <p>(00:13:30)</p>	<ul style="list-style-type: none"> • Debugging • Breakpoints • Log messages • Error messages 		 (23.8 MB)		

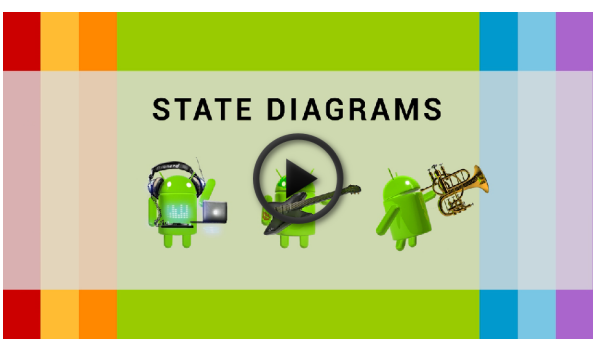



Next Steps for 4.3

(Everyone) Note that the quiz assumes you are able to complete these steps.

1. Add breakpoints to your Activity in the onCreate, onResume and onPause.
2. Run your app using the Debugger (Right/Control- click on the project name in the package view and select As>Android Application).
3. Switch to the Debug Perspective. Eventually you will want to switch/re-open the Java (Default) perspective the Java Browsing or Java Type Hierarchy perspective you will not be able to edit your project until you switch perspective.
4. When your app reaches a breakpoint it will freeze. While paused notice that you can see the stack trace (methods) and the value of your variables.
5. Click on the 'step over' button () single-step execute one line or resume button () to 'unpause' and of your app again - at least until it hits another breakpoint!

Optional further reading: [Debugging with log messages](#).

4.4. State Diagrams

 <p>STATE DIAGRAMS</p> <p>(00:18:52)</p>	<ul style="list-style-type: none"> • Introducing State Diagrams as a Computer Science concept • Media Player state diagram 		 (30.8 MB)		
---	--	---	--	---	--

Video Lecture	Key Topics	Transcript	Video Download	SRT Caption File	
---------------	------------	------------	----------------	------------------	--

Next Steps for 4.4

Add some background music to your app; create and start your MediaPlayer in the onResume method and stop it in the onPause method

```
@Override
protected void onResume() {
    Log.e("Pickle", "onResume");
    myPlayer = MediaPlayer.create(this, R.raw.your_music);
    myPlayer.start();
    super.onResume();
}

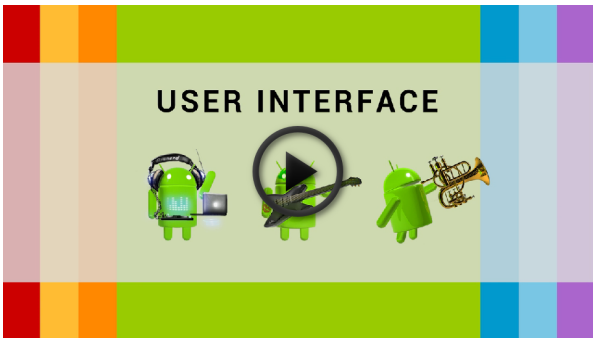



@Override
protected void onPause() {
    Log.e("Pickle", "onPause");
    myPlayer.stop();
    myPlayer.release();
    super.onPause();
}
```

(You will also need an import statement and create a variable - MediaPlayer myPlayer; - inside your class but outside of each method)

Proficient programmers can learn more about the MediaPlayer here:

- [MediaPlayer Guide](#)
- [MediaPlayer Reference](#)

4.5. User Interface

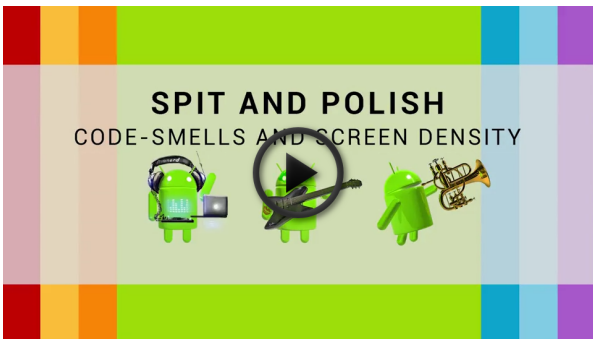



 <p>(00:22:26)</p>	<ul style="list-style-type: none"> • User Interface 		 (31.8 MB)	
---	--	---	--	---

Video Lecture	Key Topics	Transcript	Video Download	SRT Caption File	
---------------	------------	------------	----------------	------------------	--

Next Steps for 4.5

1. Add comments, use `//` or `/* ... */` to your Java code. For example explain how your code works or add your copyright information to the top of the file.
2. Let relative layout work for you: Play with and learn to use relative layout constraints (such as `android:layout_alignParentBottom`, `android:layout_toLeftOf`, `android:layout_below`).
3. Re-arrange your layout xml to avoid "forward references" i.e. constraints should refer to the id values of i defined in the xml file.

4.6. Spit and Polish (Code Smells & Screen Density)

 <p>(00:14:14)</p>	<ul style="list-style-type: none"> • Code smells • UI polish 		 (19.3 MB)	
--	--	---	--	---





Next Steps for 4.6

1. Add comments! Use `//` or `/* ... */` to comment your Java code. For example you can explain how your code works or add your name or copyright information or license to the top of your file.
2. Play with and learn about different screen density directories by moving (or placing different versions of) your app's `mdpi`/`xhdpi` etc resources.
3. Use the reference resources below (see 4.7) and what you've learned so far to improve the look and feel

(Optional) For the curious / insomniacs, you can read the full Android developer guide to supporting different

- http://developer.android.com/guide/practices/screens_support.html
- [How Android Finds the Best-matching Resource](#)

4.7. Manifest and Style Know-How

Video Lecture	Key Topics	Transcript	Video Download	SRT Caption File	
 <p>(00:15:15)</p>	<ul style="list-style-type: none"> Android Manifest tricks Using 		 (27.4 MB)		

Next Steps for 4.7





1. Download Emily's Song Eclipse project ([here](#))! After downloading, expand the contents of the zip file then Menu- Import... "Existing Android Code Into Workspace" and copy it into your workspace.
2. Play with and learn to use Android layout styles in your own app so that you can re-use padding or text s layouts or multiple items.
3. Share your wonderful app with your friends and also tell us about it in the course forum.

(A note to advanced programmers): To create a background music player that is independent of an Activity li need to use a media player inside an [Android Service](#).

(Optional) Useful reference materials that may be useful to your own app.:

- Android Asset Studio
 - [Launcher Icon Generator](#)
- Density Support
 - [Supporting Multiple Screens](#)
- Android Manifest
 - [The AndroidManifest.xml File](#)
 - [The Android Manifest's activity tag](#)
- Defining Styles
 - [Style Resource](#)
 - [Styles and Themes](#)
 - [Android style definitions](#)
- Configuration qualifiers and resource matching:
 - [How Android Finds the Best-matching Resource](#)
 - [Using configuration qualifiers](#)

John Bardeen and the Transistor Revolution

Video Lecture	Key Topics	Transcript	Video Download	SRT Caption File	
 (00:15:19)	<p>Round off your week with Lawrence's visit to the University of Illinois Spurlock Museum, and learn about ILLIAC II and two-time Nobel Prize Winner John Bardeen and his transistor revolution - the technology behind today's computing power.</p>		 (43.4 MB)		

Assignments

Once you have finished watching the videos for this week, complete the quiz on the information you learned.

To begin, access the quiz page below and click the **Start Quiz Now** button at the bottom of that page. You have 5 attempts to complete this quiz.

[Go to Week 4 Quiz](#)

This quiz is due by Sunday, January 26 at 11:55 PM Central Time ([time zone conversion](#)).

This week you will complete the evaluation phase of Assignment 1. To find out more about this process, access the Assignment 1 Evaluation page below.

[Go to Assignment 1 Evaluation](#)

This assignment is due by Sunday, January 26 at 11:55 PM Central Time ([time zone conversion](#)).

Time

This module will last **7 days** and should take **approximately 4-8 hours** of dedicated time to complete, including the videos and assignments.

Tips for Success

To do well this week, I recommend that you do the following:

- Review the video lectures a number of times to gain a solid understanding of the key questions and concepts introduced this week.
- When possible, provide tips and suggestions to your peers in this class. As a learning community, we can help each other learn and grow. One way of doing this is by helping to address the questions that your peers pose. By engaging with each other, we'll all learn better.
- It's always a good idea to refer to the video lectures in your responses. When appropriate, critique the information presented.
- Take notes while you watch the lectures for this week. By taking notes, you are interacting with the material and will find that it is easier to remember and to understand. With your notes, you'll also find that it's easier to complete your assignments. So, go ahead, do yourself a favor; take some notes!

Getting and Giving Help

We strongly encourage you to join the culture of the application development community. This means not struggling with problems in isolation! Rather, when you encounter a problem, please try the following:

- Turn to your favorite search engine and search the Internet for help. Often, you will be most successful in finding the help you need by searching for the exact text of an error message you might be encountering. Sometimes, adding the term **RESOLVED** to your search query will help you hone-in on Discussion forum posts where someone else has received advice that ultimately resolved the problem they were encountering.
- Form groups of friends, both here in this class and perhaps locally in your geographic area. You can explore the [Getting to Know Your Classmates](#) forum, reach out via the course's [social media](#) venues, or [join a Meetup](#).
- Use the [forums dedicated to each week's topics](#) for help solving technical problems on your computer or Android device. Please use the forum that most closely matches your problem. Explore the forum to see if others have encountered the same problem and received helpful advice that may be useful in your situation. If your problems persist, please do post in the forums to ask for help.

If you encounter a problem with the course itself, you have options! You can get help via any of the following means:

- You can report a specific problem by clicking on the **Help** link at the top right of any course page.
- Use the [Course Materials Errors](#) forum for problems with course materials such as typos, factual errors, or grading errors.
- Use the [Technical Issues](#) forum for problems related to the Coursera platform such as broken links, error messages, and other technical issues.

Due to the very large number of students enrolled in this course, the instructor is not able to answer emails sent directly to his account. Rather, all questions should be posted to one of the above forums. You are encouraged to help your fellow students by responding to posts made in these forums with solutions and by “voting up” the most important posts. University of Illinois staff will monitor these forums and will focus their attention on those that have been voted up the most.

Created Wed 12 Dec 2012 7:11 AM PST

Last Modified Fri 10 Jan 2014 7:17 AM PST

