



Media in Apps

Upon completion of this module, a student will be able to

- play audio from app resources
- play audio from the local file system or internet
- play video files from either location
- prepare media asynchronously
- manipulate media playback



Assignment

- Task
 - For this app, you'll build a simple media player app which will allow a user to retrieve audio files and play them
- Repo
 - https://github.com/LambdaSchool/Android_MediaPlayer
- Submission
 - Fork on github and submit pull request





A Student Can

play audio from app resources

Play Audio

- MediaPlayer
- Audio file in resource

```
MediaPlayer mediaPlayer = MediaPlayer.create(context, R.raw.sound_file_1);  
mediaPlayer.start();
```





A Student Can

play audio from the local file system or
internet

Play from URL

- URL
- Audio Stream type
- Set data source
- Prepare
 - Blocks, could take long

```
MediaPlayer mediaPlayer = new MediaPlayer();  
mediaPlayer.setAudioStreamType(AudioManager.STREAM_MUSIC);  
mediaPlayer.setDataSource(url);  
mediaPlayer.prepare(); // might take long! (for buffering, etc)  
mediaPlayer.start();
```





A Student Can
play video files

Play Video

- `VideoView` (wrapper around media player)

```
final VideoView videoView = findViewById(R.id.video_view);
videoView.setVideoURI(Uri.parse("android.resource://" + getPackageName() + "/" + R.raw.rec_longer));
/* OR */
videoView.setVideoURI(Uri.parse("http://87.98.202.49/securestream.php/6L46NDtjFEcgq6fsFgj4w/39935_540p30.mp4"));
```





A Student Can
prepare media asynchronously

Asynchronous Media

- `prepareAsync()`
- `setOnPreparedListener`





A Student Can manipulate media playback

Media Callbacks

- `setOnBufferingUpdateListener`
- `setOnCompletionListener`
- `setOnSeekCompleteListener`



Media Control

- `getCurrentPosition`
- `seekTo`
- `stop`
- `pause`

