

Chapter 6 - Media Player

Media is a crucial component of Android.

Media = Audio + Video + Images

The MediaPlayer class

This class provides APIs for playing variety of Media types. We can simply create an instance, add a music & play it.

Adding music to the android "raw" directory

We can add an mp3 file to our res/raw folder. If the raw folder is not present, you need to create one.

Playing our first music

We can create an instance of MediaPlayer and play our first music by adding following lines inside onCreate

```
MediaPlayer = MediaPlayer.create(this, R.raw.music);  
mediaPlayer.start();
```

↓
plays music.mp3

Similarly we can use mediaPlayer.pause() to pause the music from the mediaPlayer

Playing music from the web

We can play music from the web using the setDataSource method of media player.

The following steps plays music from the web

Step 1: Create a new mediaPlayer instance


```
mediaPlayer = new MediaPlayer();
```

Step 2 : Set the datasource

```
mediaPlayer.setDataSource("https://audio-source.com")
```

→ Use mp3/m... URL

Step 3 : Add android.permission.INTERNET to manifest

Step 4 : Add android:usesCleartextTraffic="true" in the application tag of manifest (AndroidManifest.xml)

Step 5 : Add OnPrepared listener to the mediaPlayer & override required methods.

Step 6 : Run mediaPlayer.prepareAsync(); method to start preparing the mediaPlayer.

Adding SeekBar

Android SDK provides a SeekBar widget which allows developers to add a progress bar for media execution.

Further we can add OnSeekBarChangeListener() to take actions when the seekbar is changed.

This can be done like this :

```
SeekBar.setMax(mediaPlayer.getDuration());
```

```
SeekBar.setOnSeekBarChangeListener(new SeekBar.OnSeekBar...
```

→ Use autocomplete & override methods

Playing Videos

We can use the same MediaPlayer class to play videos in Android.

In order to display videos we use a SurfaceView like this:

```
MediaPlayer = MediaPlayer.create(this, R.raw.vid)
```

```
SurfaceHolder h = surfaceView.getHolder();
```

```
h.addCallback(new SurfaceHolder.Callback() {
```

```
    // override methods here
```

```
});
```