To create a simple media player in Android Studio using Kotlin, follow these steps:

**Step 1: Add Audio File**

First, add an audio file (e.g., **audio.mp3**) to the **res/raw** directory of your project.

**Step 2: Set Up UI**

Design your activity layout (**activity\_main.xml**) with play and pause buttons.

#### activity\_main.xml

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Media Player"

android:textSize="24sp"

android:layout\_gravity="center"/>

<Button

android:id="@+id/btn\_play"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Play"

android:layout\_gravity="center"

android:layout\_marginTop="20dp"/>

<Button

android:id="@+id/btn\_pause"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Pause"

android:layout\_gravity="center"

android:layout\_marginTop="20dp"/>

This layout consists of a TextView displaying "Media Player", a Play button (**btn\_play**), and a Pause button (**btn\_pause**).

### Step 3: Set Up MediaPlayer

In your MainActivity, initialize the MediaPlayer and set onClickListeners for the play and pause buttons.

import android.media.MediaPlayer

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.widget.Button

class MainActivity : AppCompatActivity() {

private lateinit var mediaPlayer: MediaPlayer

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

mediaPlayer = MediaPlayer.create(this, R.raw.audio)

val btnPlay = findViewById<Button>(R.id.btn\_play)

val btnPause = findViewById<Button>(R.id.btn\_pause)

btnPlay.setOnClickListener { mediaPlayer.start() }

btnPause.setOnClickListener { mediaPlayer.pause() }

}

override fun onDestroy() {

mediaPlayer.release()

super.onDestroy()

}

}

This code initializes the MediaPlayer with the audio file **audio.mp3** from the **res/raw** directory. It sets onClickListeners for the Play and Pause buttons to start and pause the playback, respectively. The MediaPlayer is released in the onDestroy method to free up resources when the activity is destroyed.

### Step 4: Test Your App

Run your app on a device or emulator, and you should be able to play and pause the audio file using the buttons in your UI.