Video Program

ActivityMain.xml

<?*xml version*="1.0" *encoding*="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout *xmlns:android*="http://schemas.android.com/apk/res/android"  
 *xmlns:app*="http://schemas.android.com/apk/res-auto"  
 *xmlns:tools*="http://schemas.android.com/tools"  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="match\_parent"  
 *tools:context*=".MainActivity">  
  
 <SurfaceView  
 *android:id*="@+id/surfaceView"  
 *android:layout\_width*="0dp"  
 *android:layout\_height*="291dp"  
 *android:layout\_marginStart*="10dp"  
 *android:layout\_marginTop*="35dp"  
 *android:layout\_marginEnd*="10dp"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent" />  
  
 <Button  
 *android:id*="@+id/b1"  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:layout\_marginStart*="53dp"  
 *android:layout\_marginBottom*="64dp"  
 *android:text*="Play"  
 *app:layout\_constraintBottom\_toTopOf*="@+id/seekBar"  
 *app:layout\_constraintStart\_toStartOf*="parent" />  
  
 <Button  
 *android:id*="@+id/b2"  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:layout\_marginEnd*="87dp"  
 *android:text*="Stop"  
 *app:layout\_constraintBaseline\_toBaselineOf*="@+id/b1"  
 *app:layout\_constraintEnd\_toEndOf*="parent" />  
  
 <SeekBar  
 *android:id*="@+id/seekBar"  
 *android:layout\_width*="0dp"  
 *android:layout\_height*="21dp"  
 *android:layout\_marginStart*="62dp"  
 *android:layout\_marginEnd*="62dp"  
 *android:layout\_marginBottom*="175dp"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent" />  
</androidx.constraintlayout.widget.ConstraintLayout>

MainActivity.java

*package* com.example.videomediaplayer;  
  
*import* androidx.annotation.NonNull;  
*import* androidx.appcompat.app.AppCompatActivity;  
  
*import* android.media.MediaPlayer;  
*import* android.os.Bundle;  
*import* android.view.*SurfaceHolder*;  
*import* android.view.SurfaceView;  
*import* android.view.View;  
*import* android.widget.Button;  
*import* android.widget.SeekBar;  
  
*public class* MainActivity *extends* AppCompatActivity {  
 *private* MediaPlayer md1;  
 *private* Button b1;  
 *private* Button b2;  
 *private* SeekBar seekBar;  
 *private* SurfaceView surfaceView;  
  
 @Override  
 *protected void* onCreate(Bundle savedInstanceState) {  
 *super*.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 b1=findViewById(R.id.b1);  
 b2=findViewById(R.id.b2);  
 seekBar=findViewById(R.id.seekBar);  
 surfaceView=findViewById(R.id.surfaceView);  
 md1=MediaPlayer.*create*(*this*,R.raw.abcd);  
  
 b1.setOnClickListener(*new* View.OnClickListener() {  
 @Override  
 *public void* onClick(View view) {  
 md1.start();  
 }  
 });  
 b2.setOnClickListener(*new* View.OnClickListener() {  
 @Override  
 *public void* onClick(View view) {  
 md1.pause();  
 }  
 });  
 surfaceView.setKeepScreenOn(*true*);  
  
 *SurfaceHolder* sh1=surfaceView  
 .getHolder();  
  
 sh1.addCallback(*new SurfaceHolder*.Callback() {  
 @Override  
 *public void* surfaceCreated(@NonNull *SurfaceHolder* surfaceHolder) {  
 md1.setDisplay(sh1);  
 }  
  
 @Override  
 *public void* surfaceChanged(@NonNull *SurfaceHolder* surfaceHolder, *int* i, *int* i1, *int* i2) {  
  
 }  
  
 @Override  
 *public void* surfaceDestroyed(@NonNull *SurfaceHolder* surfaceHolder) {  
  
 }  
 });  
  
 seekBar.setMax(md1.getDuration());  
  
 seekBar.setOnSeekBarChangeListener(*new* SeekBar.OnSeekBarChangeListener() {  
 @Override  
 *public void* onProgressChanged(SeekBar seekBar, *int* i, *boolean* b) {  
 *if*(b){  
  
 md1.seekTo(i);  
 }  
 }  
  
 @Override  
 *public void* onStartTrackingTouch(SeekBar seekBar) {  
  
 }  
  
 @Override  
 *public void* onStopTrackingTouch(SeekBar seekBar) {  
  
 }  
 });  
  
  
  
 }  
  
 }

Output:

