ATHARVA KALE **PRACTICAL NO. 6** **ROLL NO: 27**

AIM: To Study Audio, Video And Location on Android.

❖ THEORY:

- **MediaRecorder:**
    1. It is used to record audio and video. The recording control is based on a simple state machine.
    2. The Android multimedia framework provides built-in support for capturing and encoding common audio and video formats.
    3. After recording the media, we can create a sound file that can be played later.
    4. In order to use MediaRecorder class ,you will first create an instance of MediaRecorder class.
    5. Syntax :

       MediaRecorder myAudioRecorder = new MediaRecorder();
    6. Some of the methods used in MediaRecorder:
        a. setAudioSource():This method specifies the source of audio to be recorded
        b. setVideoSource():This method specifies the source of video to be recorded.
        c. setOutputFormat():This method specifies the audio format in which audio to be stored.
        d. setAudioEncoder():This method specifies the audio encoder to be used.
        e. setOutputFile():This method configures the path to the file into which the recorded audio is to be stored.

f. stop():This method stops the recording process.
g. release():This method should be called when the recorder instance is needed.

- **Location:**
  1. A data class representing a geographic location.
  2. The Location object represents a geographic location which can consist of a latitude, longitude, time stamp, and other information such as bearing, altitude and velocity.
  3. All locations generated through LocationManager are guaranteed to have a valid latitude, longitude, and timestamp (both UTC time and elapsed real-time since boot).
  4. Methods of this object are:
     a. float distanceTo(Location dest)- Returns the approximate distance in meters between this location and the given location.
     b. float getAccuracy()- Get the estimated accuracy of this location, in meters.
     c. double getAltitude()- Get the altitude if available, in meters above sea level.
     d. float getBearing()- Get the bearing, in degrees.
     e. double getLatitude()- Get the latitude, in degrees.
     f. double getLongitude()- Get the longitude, in degrees.

ATHARVA KALE **PRACTICAL NO. 6** **ROLL NO: 27**

A) Write a program to:

a) Record an audio and play it.

b) Play a video in a video view.

- **CODE:**
    - **AndroidManifest.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.practical6_a">
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.RECORD_AUDIO" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.STORAGE" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Practical6_A">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

    - **activity_main.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">

    <Button
        android:id="@+id/button5"
        android:layout_width="96dp"
        android:layout_height="55dp"
        android:text="PLAY"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.295"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.318" />

    <Button
        android:id="@+id/button4"
        android:layout_width="96dp"
        android:layout_height="55dp"
        android:text="STOP PLAY"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.717"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.318" />

    <Button
        android:id="@+id/button3"
        android:layout_width="96dp"
        android:layout_height="55dp"
        android:text="PLAY VIDEO"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.458" />

    <Button
        android:id="@+id/button2"
        android:layout_width="96dp"
        android:layout_height="55dp"
        android:text="STOP RECORD"
```

```
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.717"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.176" />

  <TextView
    android:id="@+id/textView"
    android:layout_width="294dp"
    android:layout_height="37dp"
    android:text="AUDIO AND VIDEO"
    android:textAlignment="center"
    android:textSize="30sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.495"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.023" />

  <VideoView
    android:id="@+id/videoView"
    android:layout_width="365dp"
    android:layout_height="293dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.549"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.904" />

  <Button
    android:id="@+id/button"
    android:layout_width="96dp"
    android:layout_height="55dp"
    android:text="RECORD"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.295"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
```

```
    app:layout_constraintVertical_bias="0.176" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **MainActivity.java:**

```
package com.example.practical6_a;

import static android.Manifest.permission.RECORD_AUDIO;
import static android.Manifest.permission.WRITE_EXTERNAL_STORAGE;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;
import android.annotation.SuppressLint;
import android.content.pm.PackageManager;
import android.media.MediaPlayer;
import android.media.MediaRecorder;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.MediaController;
import android.widget.Toast;
import android.widget.VideoView;
import java.io.IOException;
import java.util.Random;

public class MainActivity extends AppCompatActivity {
   Button buttonStart, buttonStop, buttonPlayLastRecordAudio, playvideo,buttonStopPlayingRecording ;
   String AudioSavePathInDevice = null;
   MediaRecorder mediaRecorder ;
   Random random ;
   VideoView vw;

   String RandomAudioFileName = "ABCDEFGHIJKLMNOP";
   public static final int RequestPermissionCode = 1;
   MediaPlayer mediaPlayer ;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
```

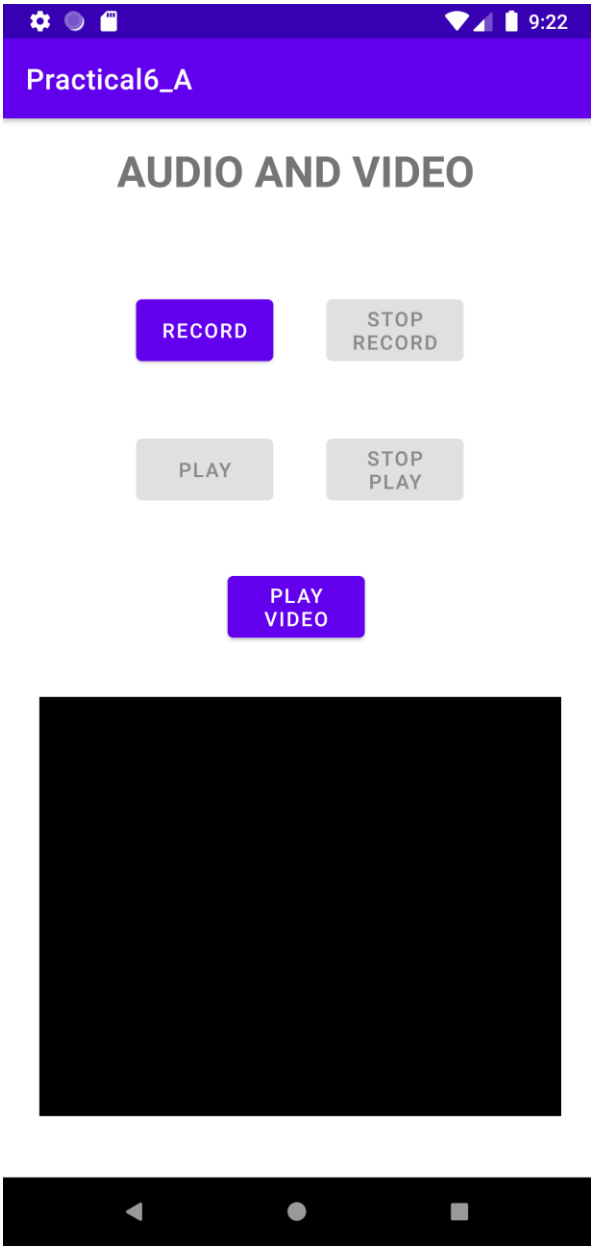
```
        vw = findViewById(R.id.videoView);
        buttonStart = (Button) findViewById(R.id.button);
        buttonStop = (Button) findViewById(R.id.button2);
        buttonPlayLastRecordAudio = (Button) findViewById(R.id.button5);
        buttonStopPlayingRecording = (Button)findViewById(R.id.button4);
        playvideo=findViewById(R.id.button3);
        buttonStop.setEnabled(false);
        buttonPlayLastRecordAudio.setEnabled(false);
        buttonStopPlayingRecording.setEnabled(false);

        random = new Random();
        playvideo.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Toast.makeText(MainActivity.this, "STARTING VIDEO", Toast.LENGTH_LONG).show();
                Uri vidUri =
Uri.parse("https://ia800201.us.archive.org/22/items/ksnn_compilation_master_the_internet/ksnn_com
pilation_master_the_internet_512kb.mp4");
                vw = findViewById(R.id.videoView);
                vw.setVideoURI(vidUri);
                vw.setMediaController(new MediaController(MainActivity.this));
                vw.requestFocus();
                vw.start();
            }
        });
        buttonStart.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                if(checkPermission()) {

                    AudioSavePathInDevice =
                            Environment.getExternalStorageDirectory().getAbsolutePath() + "/" +
                                    CreateRandomAudioFileName(5) + "AudioRecording.3gp";

                    MediaRecorderReady();

                    try {
                        mediaRecorder.prepare();
                        mediaRecorder.start();
                        Toast.makeText(MainActivity.this, "Recording started", Toast.LENGTH_LONG).show();
                    } catch (IllegalStateException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
```

```
            } catch (IOException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }

            buttonStart.setEnabled(false);
            buttonStop.setEnabled(true);


        } else {
            requestPermission();
        }
    }
});

buttonStop.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        mediaRecorder.stop();
        buttonStop.setEnabled(false);
        buttonPlayLastRecordAudio.setEnabled(true);
        buttonStart.setEnabled(true);
        buttonStopPlayingRecording.setEnabled(false);

        Toast.makeText(MainActivity.this, "Recording Completed",
                Toast.LENGTH_LONG).show();
    }
});

buttonPlayLastRecordAudio.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) throws IllegalArgumentException,
            SecurityException, IllegalStateException {

        buttonStop.setEnabled(false);
        buttonStart.setEnabled(false);
        buttonStopPlayingRecording.setEnabled(true);
        mediaPlayer = new MediaPlayer();
        try {
            mediaPlayer.setDataSource(AudioSavePathInDevice);
            mediaPlayer.prepare();
        } catch (IOException e) {
            e.printStackTrace();
        }
```

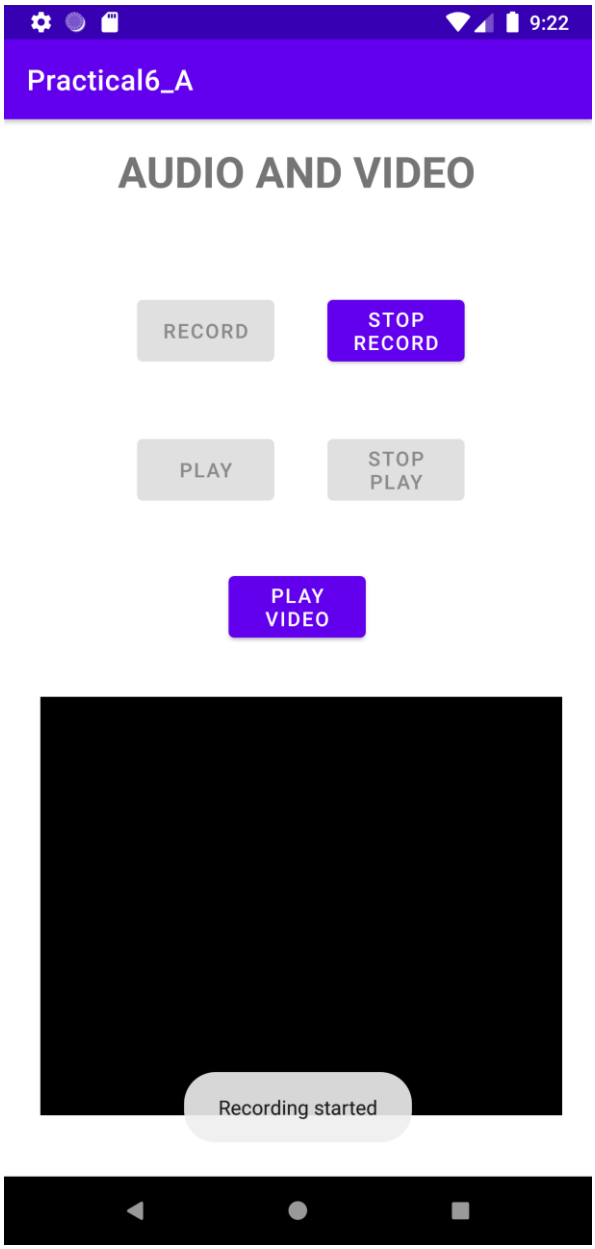
```
            mediaPlayer.start();
            Toast.makeText(MainActivity.this, "Recording Playing",
                    Toast.LENGTH_LONG).show();
        }
    });
    buttonStopPlayingRecording.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            buttonStop.setEnabled(false);
            buttonStart.setEnabled(true);
            buttonStopPlayingRecording.setEnabled(false);
            buttonPlayLastRecordAudio.setEnabled(true);

            if(mediaPlayer != null){
                mediaPlayer.stop();
                mediaPlayer.release();
                MediaRecorderReady();
            }
        }
    });
}
public void MediaRecorderReady(){
    mediaRecorder=new MediaRecorder();
    mediaRecorder.setAudioSource(MediaRecorder.AudioSource.MIC);
    mediaRecorder.setOutputFormat(MediaRecorder.OutputFormat.THREE_GPP);
    mediaRecorder.setAudioEncoder(MediaRecorder.AudioEncoder.AMR_NB);
    mediaRecorder.setOutputFile(AudioSavePathInDevice);
}
public String CreateRandomAudioFileName(int string){
    StringBuilder stringBuilder = new StringBuilder( string );
    int i = 0 ;
    while(i < string ) {
        stringBuilder.append(RandomAudioFileName.
                charAt(random.nextInt(RandomAudioFileName.length())));

        i++ ;
    }
    return stringBuilder.toString();
}
private void requestPermission() {
    ActivityCompat.requestPermissions(MainActivity.this, new
            String[]{WRITE_EXTERNAL_STORAGE, RECORD_AUDIO}, RequestPermissionCode);
}
public void onRequestPermissionsResult(int requestCode, String permissions[], int[] grantResults) {
```

```
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    switch (requestCode) {
      case RequestPermissionCode:
        if (grantResults.length > 0) {
          boolean StoragePermission = grantResults[0] ==
              PackageManager.PERMISSION_GRANTED;
          boolean RecordPermission = grantResults[1] ==
              PackageManager.PERMISSION_GRANTED;

          if (StoragePermission && RecordPermission) {
            Toast.makeText(MainActivity.this, "Permission Granted",
                Toast.LENGTH_LONG).show();
          } else {
            Toast.makeText(MainActivity.this, "Permission Denied", Toast.LENGTH_LONG).show();
          }
        }
        break;
    }
  }
  public boolean checkPermission() {
    int result = ContextCompat.checkSelfPermission(getApplicationContext(),
        WRITE_EXTERNAL_STORAGE);
    int result1 = ContextCompat.checkSelfPermission(getApplicationContext(),
        RECORD_AUDIO);
    return result == PackageManager.PERMISSION_GRANTED &&
        result1 == PackageManager.PERMISSION_GRANTED;
  }
}
```

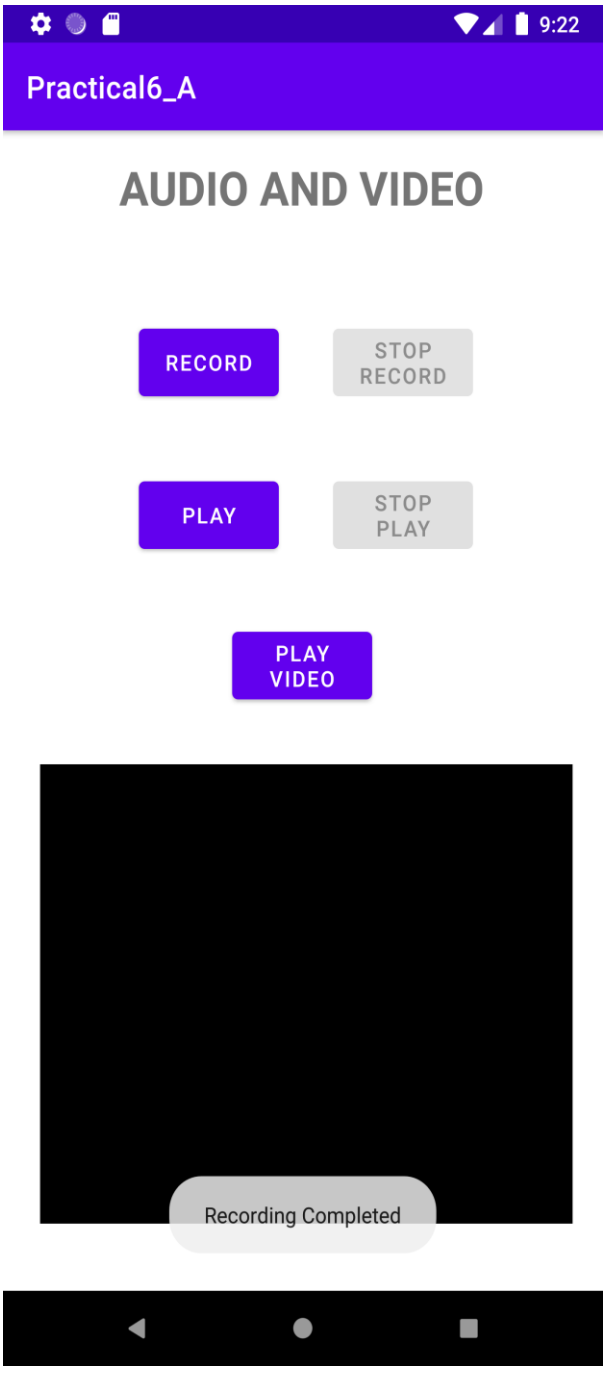
- **OUTPUT:**
  - **GUI**

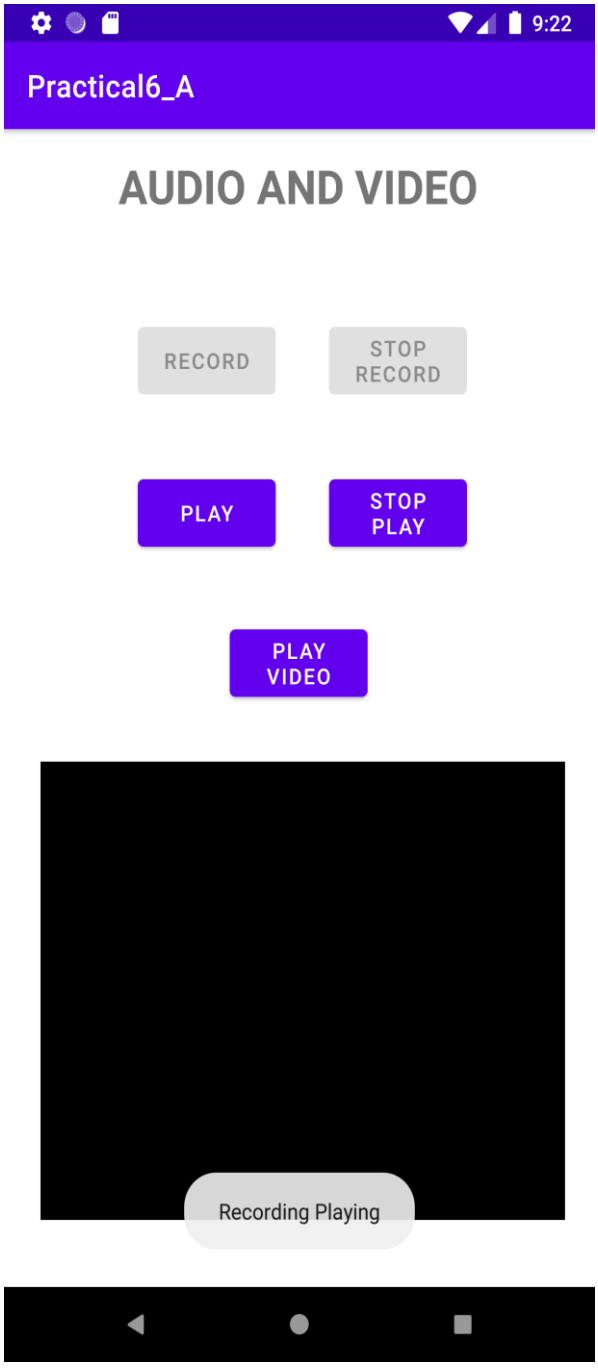


Recording Started

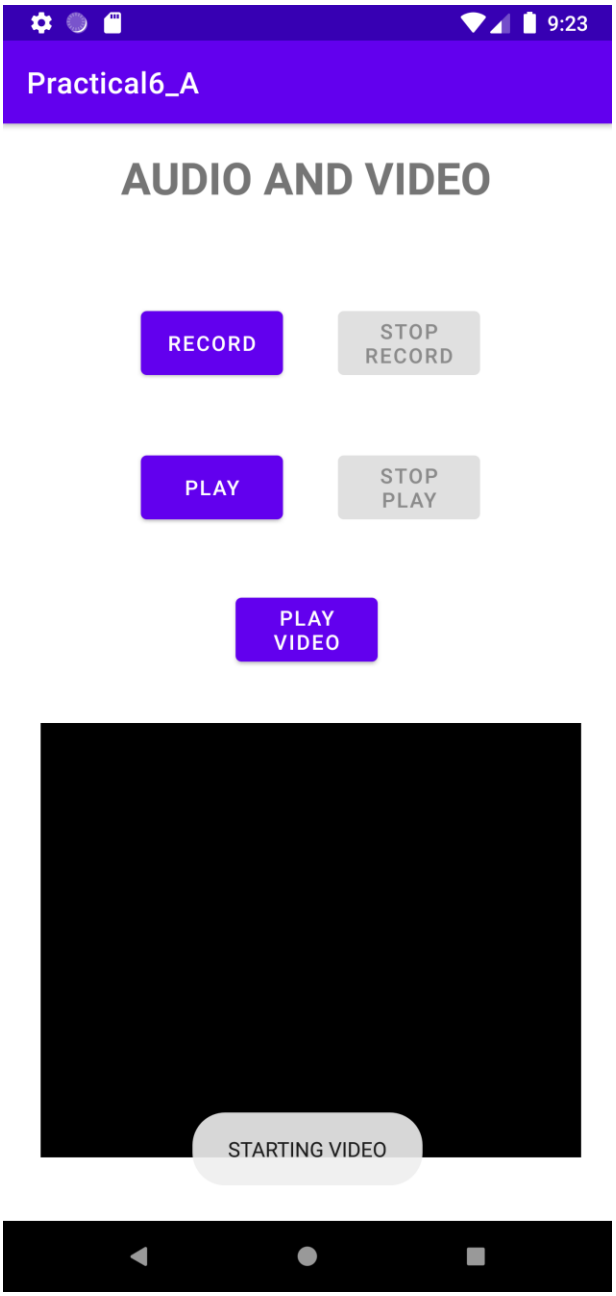


- **Recording Completed** **Recording Playing**

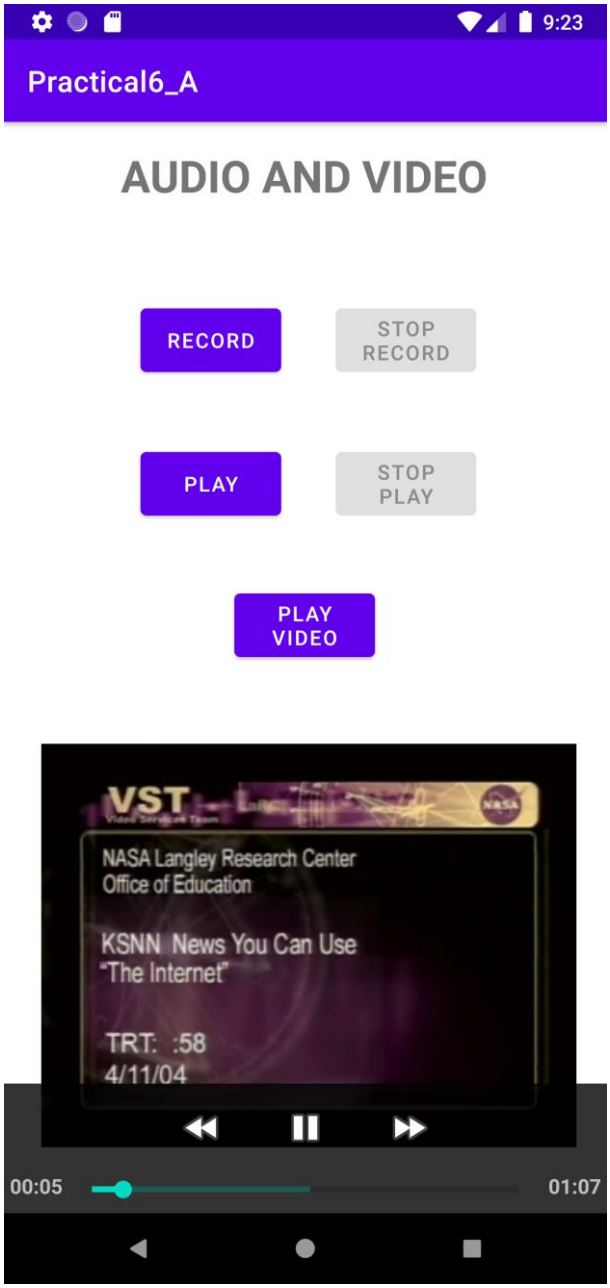




- **Starting Video**



Video Playing



B) Create an application to display the current location of your device. (Latitude & Longitude value):

- **CODE:**
  - **AndroidManifest.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.practical6b">
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Practical6B">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

  - **activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
```

```
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/showLocation"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="Location"
        android:textSize="24sp" />
    <Button
        android:id="@+id/btnGetLocation"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Get Location" />
</LinearLayout>
```

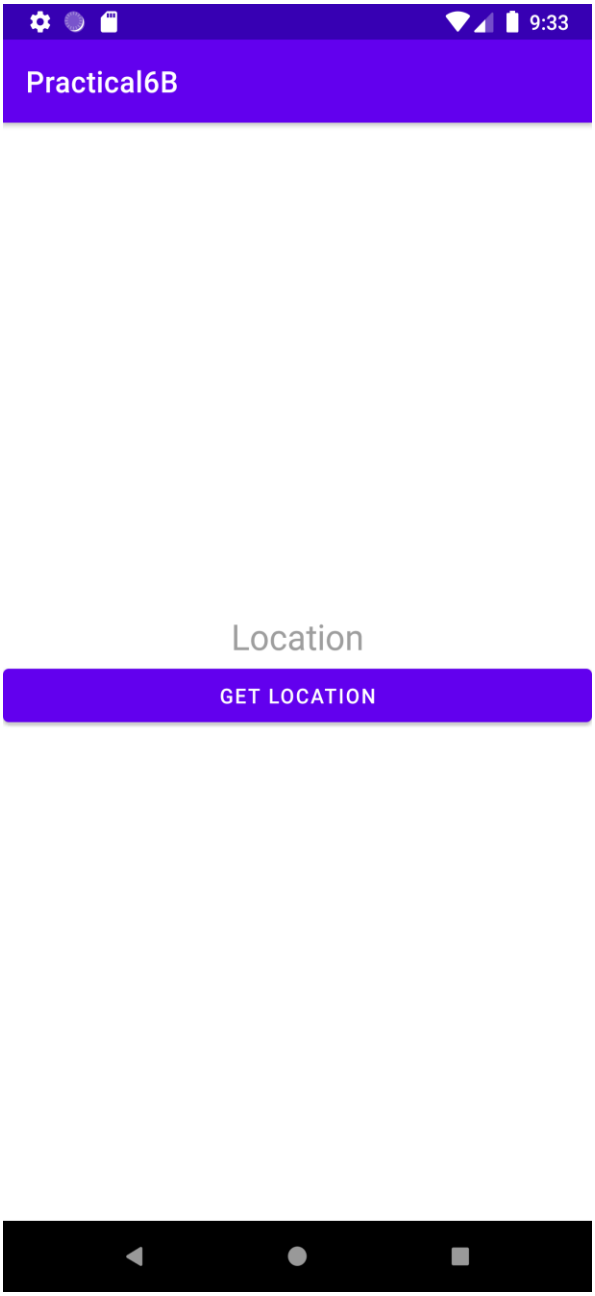
- **MainActivity.java:**

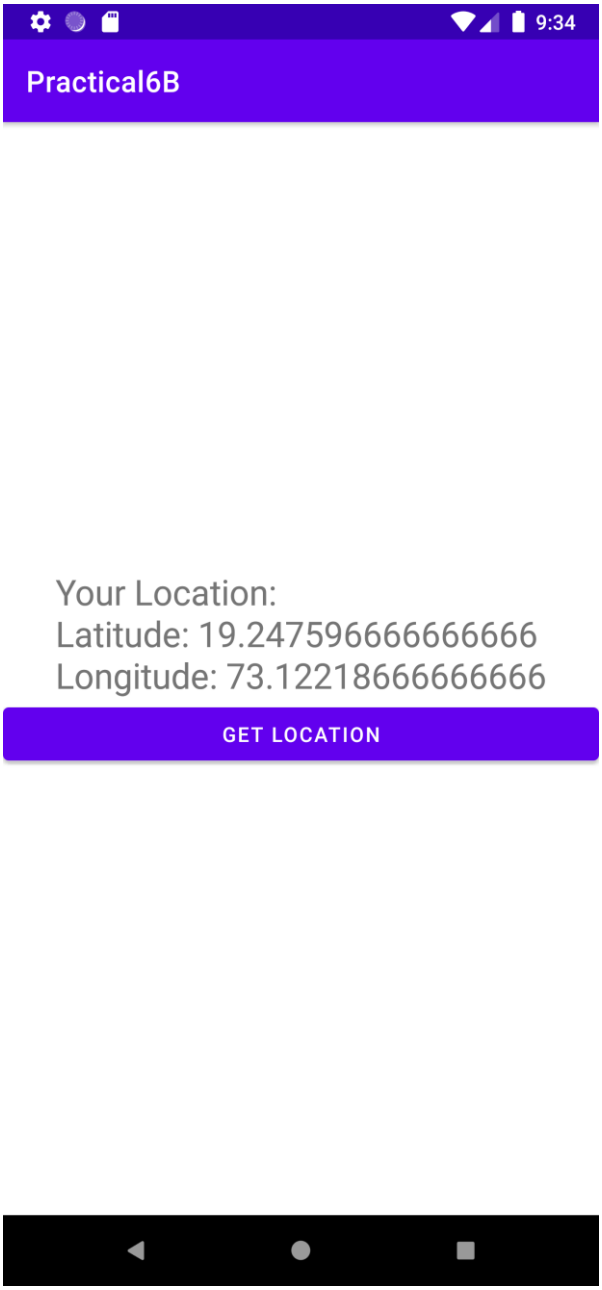
```
package com.example.practical6b;

import android.Manifest;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Location;
import android.location.LocationManager;
import android.provider.Settings;
import android.support.v4.app.*;
import android.support.*;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

public class MainActivity extends AppCompatActivity {
    private static final int REQUEST_LOCATION = 1;
    Button btnGetLocation;
    TextView showLocation;
    LocationManager locationManager;
```

```
    String latitude, longitude;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ActivityCompat.requestPermissions( this,
                new String[] {Manifest.permission.ACCESS_FINE_LOCATION}, REQUEST_LOCATION);
        showLocation = findViewById(R.id.showLocation);
        btnGetLocation = findViewById(R.id.btnGetLocation);
        btnGetLocation.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
                if (!locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER)) {
                    OnGPS();
                } else {
                    getLocation();
                }
            }
        });
    }
    private void OnGPS() {
        final AlertDialog.Builder builder = new AlertDialog.Builder(this);
        builder.setMessage("Enable GPS").setCancelable(false).setPositiveButton("Yes", new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                startActivity(new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS));
            }
        }).setNegativeButton("No", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel();
            }
        });
        final AlertDialog alertDialog = builder.create();
        alertDialog.show();
    }
    private void getLocation() {
        if (ActivityCompat.checkSelfPermission(
                MainActivity.this,Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(
                MainActivity.this, Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
```

```
        ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, REQUEST_LOCATION);
      } else {
        Location locationGPS =
locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
        if (locationGPS != null) {
          double lat = locationGPS.getLatitude();
          double longi = locationGPS.getLongitude();
          latitude = String.valueOf(lat);
          longitude = String.valueOf(longi);
          showLocation.setText("Your Location: " + "\n" + "Latitude: " + latitude + "\n" + "Longitude: " +
longitude);
        } else {
          Toast.makeText(this, "Unable to find location.", Toast.LENGTH_SHORT).show();
        }
      }
    }
}
```

- **OUTPUT:**
  - **GUI** **Location**





❖ **CONCLUSION:**

Hence we successfully studied audio, video and location on android.