Advanced Mobile Programming
Practical #10

Name	Sanjeev Gupta	**Roll Number**	21302B0023
Subject/Course:	Advanced Mobile Programming		
Topic	Programming Media API and Telephone API		

Media API

10.a) To create a program to access music (media) in your mobile phone.

```
package com.example.amp_p10_1
import android.content.ContentResolver
import android.content.pm.PackageManager
import android.os.Bundle
import android.provider.MediaStore
import android.support.v7.app.AppCompatActivity
import android.support.v4.app.ActivityCompat
import android.widget.ArrayAdapter
import android.widget.ListView
class MainActivity : AppCompatActivity() {
    private val REQUEST_PERMISSION_CODE = 123
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        if (ActivityCompat.checkSelfPermission(
                this,
                android.Manifest.permission.READ_EXTERNAL_STORAGE
            ) != PackageManager.PERMISSION_GRANTED
        ) {
            ActivityCompat.requestPermissions(
                this,
                arrayOf(android.Manifest.permission.READ_EXTERNAL_STORAGE),
                REQUEST_PERMISSION_CODE
            )
        } else {
            loadMusic()
        }
    }
    private fun loadMusic() {
        val musicList = mutableListOf<String>()
        val musicListView = findViewById<ListView>(R.id.musicListView)
        val contentResolver: ContentResolver = contentResolver
        val musicUri = MediaStore.Audio.Media.EXTERNAL_CONTENT_URI
        val cursor = contentResolver.query(
            musicUri,
            null,
            null,
            null,
      null
    )
    if (cursor != null && cursor.moveToFirst()) {
      val titleColumn = cursor.getColumnIndex(MediaStore.Audio.Media.TITLE)
      do {
        val title = cursor.getString(titleColumn)
        musicList.add(title)
      } while (cursor.moveToNext())
      cursor.close()
    }
    val adapter = ArrayAdapter(this, android.R.layout.simple_list_item_1, musicList)
    musicListView.adapter = adapter
  }
```

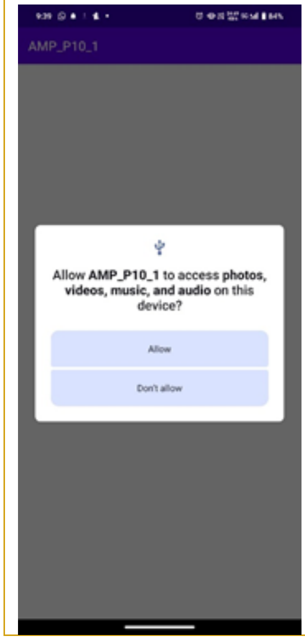
AndroidManifest.xml

```
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.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">
  <ListView
    android:id="@+id/musicListView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
</android.support.constraint.ConstraintLayout>
```

OUTPUT:



Allow **AMP_P10_1** to access **photos, videos, music, and audio** on this device?

Telephone API

10.b) To create a program that uses the calling feature of android mobile phones.

MainActivity.kt

```
package com.example.amp_7_1

import android.content.Intent
import android.content.pm.PackageManager
import android.net.Uri
import android.support.v7.app.AppCompatActivity
import android.os.Bundle
import android.support.v4.app.ActivityCompat
import android.widget.Button

class MainActivity : AppCompatActivity() {
   val mobno:String="8888888888"
   val REQUEST_PHONE_CALL=1
   override fun onCreate(savedInstanceState: Bundle?) {
      super.onCreate(savedInstanceState)
      setContentView(R.layout.activity_main)
      val call=findViewById<Button>(R.id.phonecall)
      call.setOnClickListener{

if(ActivityCompat.checkSelfPermission(this,android.Manifest.permission.CALL_PHONE)!=PackageManager.PERMISSION_GRANTED)
         {
            ActivityCompat.requestPermissions(this,
arrayOf(android.Manifest.permission.CALL_PHONE),REQUEST_PHONE_CALL)
         }
         else
         {
            makecall()
         }
      }
   }
   private fun makecall(){
      val intent=Intent(Intent.ACTION_CALL, Uri.fromParts("tel",mobno,null))
      startActivity(intent)
   }

   override fun onRequestPermissionsResult(
      requestCode: Int,
      permissions: Array<out String>,
      grantResults: IntArray
   ) {
      if (requestCode==REQUEST_PHONE_CALL)
      {
         makecall()
```

```
        }
    }
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.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:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Place Call"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        android:id="@+id/phonecall"
        android:layout_marginTop="200dp"
        app:layout_constraintTop_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>
```

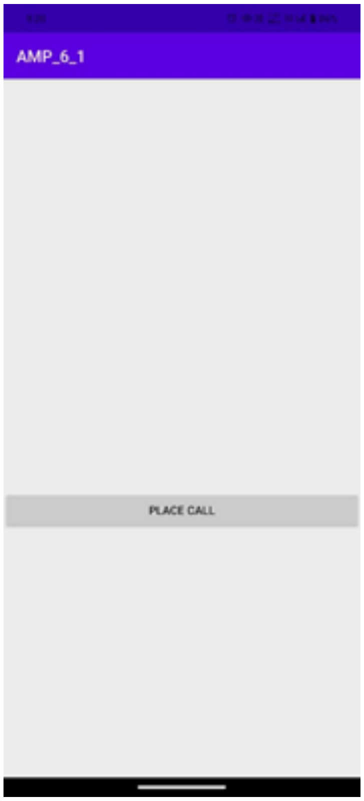
AndroidManifest.xml

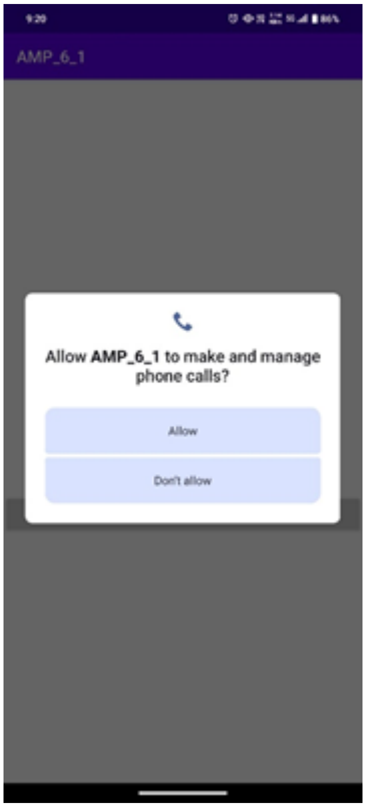
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.example.amp_6_1">
    <uses-permission android:name="android.permission.CALL_PHONE"></uses-permission>
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.AMP_6_1"
        tools:targetApi="31">
        <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>
```

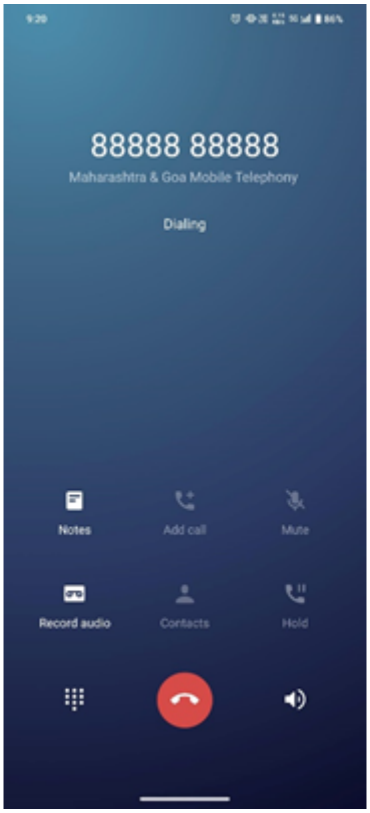
```
</manifest>
```

OUTPUT:





Allow **AMP_6_1** to make and manage phone calls?



88888 88888

Maharashtra & Goa Mobile Telephony

Dialing