



---

# MEDIA PLAYER ASSIGNMENT

---

Submitted to :Sir Nisar Siddiqui



MUHAMMAD DANIYAL QURESHI

021-18-0009

BSCS\_VII

## Contents

Layout: .....	1
1.Main Activity .....	1
2. Player Activity .....	2
3.Playlist Activity .....	4
Java Code .....	5
1. Main Activity .....	5
2.Player Activity .....	12
3.Playlist Activity .....	16
Songs Class .....	17
Menu .....	18
Output .....	19

## Layout:

### 1.Main Activity

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:orientation="vertical"
    >

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#000000"
    android:backgroundTint="#000000"
    android:gravity="center"
    >
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Songs"
    android:id="@+id/playlistname"
    android:textColor="#ffffff"
    android:textSize="20sp"
```

```

        ></TextView>
    </LinearLayout>

    <ListView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/listView"
        android:choiceMode="multipleChoice"
    />

</LinearLayout>

```

## 2. Player Activity

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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"
    android:padding="30dp"
    tools:context=".PlayerActivity"
    >

    <ImageView
        android:layout_width="300dp"
        android:layout_height="300dp"
        android:id="@+id/headphoens"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="10dp"
        android:src="@drawable/b"
    />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:layout_marginTop="15dp"
        android:layout_marginBottom="8dp"
        android:gravity="center"
        android:id="@+id/songName"
        android:text="Song Name"

        android:layout_below="@+id/headphoens"/>

```

```
<SeekBar
    android:layout_width="320dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="30dp"
    android:layout_centerHorizontal="true"
    android:id="@+id/mSeekBarTime"
    android:layout_below="@id/songName"/>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <TextView
        android:id="@+id/elapsedTimeLabel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="0:11"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="400dp"
    />
    <TextView
        android:id="@+id/remainingTimeLabel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="-1:49"
        android:layout_marginLeft="200dp"
        android:layout_marginTop="400dp"
    />
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="30dp"
    android:id="@+id/linearlayout"
    android:layout_below="@+id/mSeekBarTime"
    android:gravity="center">
```

```
<ImageView
```

```

        android:layout_width="40dp"
        android:layout_height="40dp"
        android:src="@drawable/previous"
        android:id="@+id/previous"
        android:onClick="previous"

    />

    <ImageView
        android:layout_width="40dp"
        android:layout_height="40dp"
        android:src="@drawable/play"
        android:layout_marginRight="30dp"
        android:layout_marginLeft="30dp"
        android:id="@+id/play"
        android:onClick="play"
    />

    <ImageView
        android:layout_width="40dp"
        android:layout_height="40dp"
        android:src="@drawable/next"
        android:id="@+id/next"
        android:onClick="next"

    />

</LinearLayout>

</RelativeLayout>

```

### 3. Playlist Activity

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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=".playlist"
    android:orientation="vertical"

```

```

>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#000000"
    android:backgroundTint="#000000"
    android:gravity="center"
    >
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Playlists"
        android:textColor="#ffffff"
        android:textSize="20sp"
    ></TextView>
</LinearLayout>

<ListView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/listView"
    />

</LinearLayout>

```

## Java Code

### 1. Main Activity

```

2. package com.example.mediaplayer_assignment;
3.
4. import androidx.annotation.NonNull;
5. import androidx.appcompat.app.AlertDialog;
6. import androidx.appcompat.app.AppCompatActivity;
7. import androidx.core.app.ActivityCompat;
8. import androidx.core.content.ContextCompat;
9.
10. import android.Manifest;
11. import android.content.Context;
12. import android.content.DialogInterface;
13. import android.content.Intent;
14. import android.content.SharedPreferences;
15. import android.content.pm.PackageManager;
16. import android.media.MediaPlayer;

```

```
17.import android.net.Uri;
18.import android.os.Build;
19.import android.os.Bundle;
20.import android.os.Environment;
21.import android.util.Log;
22.import android.view.Menu;
23.import android.view.MenuInflater;
24.import android.view.MenuItem;
25.import android.view.View;
26.import android.widget.AdapterView;
27.import android.widget.AdapterView;
28.import android.widget.EditText;
29.import android.widget.ListAdapter;
30.import android.widget.ListView;
31.import android.widget.TextView;
32.import android.widget.Toast;
33.import java.io.File;
34.import java.io.FileNotFoundException;
35.import java.io.FileOutputStream;
36.import java.io.OutputStream;
37.import java.io.PrintStream;
38.import java.lang.reflect.Array;
39.import java.lang.reflect.Method;
40.import java.nio.file.Files;
41.import java.util.ArrayList;
42.import java.util.List;
43.import java.util.Scanner;
44.import java.util.zip.Inflater;
45.
46.public class MainActivity extends AppCompatActivity {
47.
48.    ListView listview;
49.    MediaPlayer mp;
50.    SharedPreferences.Editor editor;
51.    int counter;
52.    boolean check;
53.    boolean add;
54.    String playlist_name;
55.    PrintStream output;
56.    boolean checkboxes;
57.    @Override
58.    protected void onCreate(Bundle savedInstanceState) {
59.        super.onCreate(savedInstanceState);
60.        setContentView(R.layout.activity_main);
61.        check=false;
```

```

62.         add=false;
63.         checkboxes=false;
64.         Intent intent=getIntent();
65.         TextView playlistname=findViewById(R.id.playlistname);
66.         listview=findViewById(R.id.listView);
67.
68.         if(intent.getExtras()!=null) {
69.             playlist_name=intent.getStringExtra("name");
70.             playlistname.setText(playlist_name);
71.             add=true;
72.             try {
73.                 output = new PrintStream(openFileOutput(playlist_name+".txt", MODE_APPEND));
74.
75.             }
76.             catch (Exception e){}
77.
78.         }
79.         else
80.             playlist_name="songs";
81.
82.         SharedPreferences prefs=getSharedPreferences("playlistnames",MODE_PRIVATE);
83.         editor=prefs.edit();
84.
85.         counter=0;
86.         while(!prefs.getString(""+counter,"-1").equals("-1"))
87.             counter+=1;
88.
89.         if (ContextCompat.checkSelfPermission(this, Manifest.permission.WRITE_EXTERNAL_STORAGE) != PackageManager.PERMISSION_GRANTED &&
90.             ContextCompat.checkSelfPermission(this, Manifest.permission.READ_EXTERNAL_STORAGE) != PackageManager.PERMISSION_GRANTED) {
91.
92.             ActivityCompat.requestPermissions(this,
93.                 new String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE, Manifest.permission.READ_EXTERNAL_STORAGE},

```



```

94.             123);
95.         }
96.         else
97.             getSongs(playlist_name);
98.
99.     }
100.
101.     public void getSongs(String listname) {
102.         String[] song_names={};
103.         ArrayList<File> list=new ArrayList<File>();
104.
105.         Songs.files = (ArrayList<File>) Songs.getListFiles(Environment.getExternalStorageDirectory());
106.
107.         if (!listname.equals("songs"))
108.             list=getPlaylist();
109.
110.         if(list.size()==0)
111.             list=Songs.files;
112.
113.         Songs.current=list;
114.
115.         song_names = new String[list.size()];
116.         for (int i = 0; i < list.size(); i++)
117.             song_names[i] = list.get(i).getName().toString();
118.
119.         Intent intent=new Intent(this,PlayerActivity.class);
120.
121.         ListAdapter listAdapter= new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1,song_names);
122.
123.         if(check)
124.             listAdapter= new ArrayAdapter<String>(this, android.R.layout.simple_list_item_multiple_choice,song_names);
125.
126.         listview.setAdapter(listAdapter);
127.
128.         listview.setOnItemClickListener(
129.             new AdapterView.OnItemClickListener() {
130.                 @Override
131.                 public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
132.

```

```

133.                if(!checkboxes) {
134.                    Log.i("cl","click");
135.                    intent.putExtra("index", position);
136.                    startActivity(intent);
137.                }
138.            }
139.        }
140.
141.    );
142.
143.    }
144.
145.    public ArrayList<File> getPlaylist(){
146.        //Read from playlist songs file
147.        ArrayList<Integer> indices=new ArrayList<Integer>();
148.        try {
149.            Scanner scan = new Scanner(openFileInput(playlist_name+"
150.            .txt"));
151.            while(scan.hasNextLine())
152.                indices.add(Integer.parseInt(scan.nextLine()));
153.            scan.close();
154.
155.            ArrayList<File> list=new ArrayList<File>();
156.
157.            for (int i = 0; i < Songs.files.size(); i++)
158.                if(indices.contains(i))
159.                    list.add(Songs.files.get(i));
160.
161.            return list;
162.        }
163.        catch (Exception ee){}
164.
165.        return null;
166.    }
167.
168.    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
169.        switch (item.getItemId()){
170.            case R.id.addsong:

```

```

171.             check=true;
172.             checkboxes=true;
173.             invalidateOptionsMenu();
174.             getSongs("songs");
175.             break;
176.
177.             case R.id.addplaylist:
178.                 showForgotDialog(this);
179.
180.                 break;
181.             case R.id.playlist:
182.                 Intent intent=new Intent(this,playlist.class);
183.                 intent.putExtra("counter",counter);
184.                 startActivity(intent);
185.                 finish();
186.                 break;
187.
188.             case R.id.check:
189.                 for (int i=0;i<listview.getCount();i++)
190.                     if(listview.isItemChecked(i))
191.                         output.println(i);
192.                     output.close();
193.                 check=false;
194.                 add=false;
195.                 checkboxes=false;
196.                 getSongs(playlist_name);
197.
198.                 break;
199.         }
200.         return super.onOptionsItemSelected(item);
201.     }
202.
203.     @Override
204.     public void onRequestPermissionsResult(int requestCode, String[]
permissions, int[] grantResults) {
205.         super.onRequestPermissionsResult(requestCode, permissions, g
rantResults);
206.         switch (requestCode) {
207.             case 123:
208.                 if (grantResults.length > 0
209.                     && grantResults[0] == PackageManager.PERMISS
ION_GRANTED) {
210.                     getSongs(playlist_name);
211.                 }
212.                 break;

```

```

213.
214.     }
215. }
216.
217.     private void showForgotDialog(Context c) {
218.         final EditText taskEditText = new EditText(c);
219.         AlertDialog dialog = new AlertDialog.Builder(c)
220.             .setTitle("New Playlist")
221.             .setView(taskEditText)
222.             .setPositiveButton("Add", new DialogInterface.OnClick
                mListener() {
223.
224.                 @Override
225.                 public void onClick(DialogInterface dialog, int
                    which) {
226.
227.                     editor.putString(""+counter,String.valueOf(t
                        askEditText.getText()));
228.
229.                     editor.commit();
230.                     ++counter;
231.                 }
232.             })
233.             .setNegativeButton("Cancel", null)
234.             .create();
235.         dialog.show();
236.     }
237.
238.
239.     @Override
240.     public boolean onCreateOptionsMenu(Menu menu) {
241.         MenuInflater inflater=getMenuInflater();
242.         inflater.inflate(R.menu.menu,menu);
243.         menu.findItem(R.id.check).setVisible(check);
244.         menu.findItem(R.id.addsong).setVisible(add);
245.
246.         return true;
247.     }

```

## 2.Player Activity

```
package com.example.mediaplayer_assignment;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.media.MediaPlayer;
import android.net.Uri;
import android.os.Bundle;
import android.os.Handler;
import android.os.Message;
import android.view.View;
import android.widget.ImageView;
import android.widget.SeekBar;
import android.widget.TextView;

public class PlayerActivity extends AppCompatActivity {

    ImageView play;
    MediaPlayer mp;
    int index;
    TextView songname;
    SeekBar seekBar;

    TextView elapsedTimeLabel, remainingTimeLabel;
    int totalTime;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_player);
        songname=findViewById(R.id.songName);
        play=findViewById(R.id.play);
        seekBar=findViewById(R.id.mSeekBarTime);
        elapsedTimeLabel = findViewById(R.id.elapsedTimeLabel);
        remainingTimeLabel = findViewById(R.id.remainingTimeLabel);

        Intent intent=getIntent();
        index=intent.getIntExtra("index",0);
        songname.setText(Songs.current.get(index).getName());

        Uri path= Uri.parse(Songs.current.get(index).getAbsolutePath());
        mp=MediaPlayer.create(this,path);
        mp.setLooping(true);
```

```

        mp.seekTo(0);
        mp.setVolume(0.5f, 0.5f);
        totalTime = mp.getDuration();

        // Position Bar
        seekBar.setMax(totalTime);
        seekBar.setOnSeekBarChangeListener(
            new SeekBar.OnSeekBarChangeListener() {
                @Override
                public void onProgressChanged(SeekBar seekBar, int progress,
boolean fromUser) {
                    if (fromUser) {
                        mp.seekTo(progress);
                        seekBar.setProgress(progress);
                    }
                }

                @Override
                public void onStartTrackingTouch(SeekBar seekBar) {
                }

                @Override
                public void onStopTrackingTouch(SeekBar seekBar) {
                }
            }
        );

        // Thread (Update seekBar & timeLabel)
        new Thread(new Runnable() {
            @Override
            public void run() {
                while (mp != null) {
                    try {
                        Message msg = new Message();
                        msg.what = mp.getCurrentPosition();
                        handler.sendMessage(msg);
                        Thread.sleep(1000);
                    } catch (InterruptedException ignored) {}
                }
            }
        }).start();
    }
}

```

```

private Handler handler = new Handler(new Handler.Callback() {
    @Override
    public boolean handleMessage(Message msg) {
        int currentPosition = msg.what;
        // Update seekBar.
        seekBar.setProgress(currentPosition);

        // Update Labels.
        String elapsedTime = createTimeLabel(currentPosition);
        elapsedTimeLabel.setText(elapsedTime);

        String remainingTime = "- " + createTimeLabel(totalTime - currentPosi
tion);
        remainingTimeLabel.setText(remainingTime);

        return true;
    }
});

public String createTimeLabel(int time) {
    String timeLabel = "";
    int min = time / 1000 / 60;
    int sec = time / 1000 % 60;

    timeLabel = min + ":";
    if (sec < 10) timeLabel += "0";
    timeLabel += sec;

    return timeLabel;
}

public void play(View view){
    if(mp.isPlaying()) {
        mp.pause();
        play.setImageResource(R.drawable.play);
    }
    else {
        mp.start();
        play.setImageResource(R.drawable.pause);
    }
}

public void previous(View view){

```

```

        if(index>0)
            --index;
        else
            index=Songs.current.size()-1;

        mp.stop();
        Uri path= Uri.parse(Songs.current.get(index).getAbsolutePath());
        mp=MediaPlayer.create(this,path);
        mp.start();
        mp.setLooping(true);
        play.setImageResource(R.drawable.pause);
        songname.setText(Songs.current.get(index).getName());

    }

    public void next(View view){
        if(index<Songs.current.size()-1)
            ++index;
        else
            index=0;

        mp.stop();
        Uri path= Uri.parse(Songs.current.get(index).getAbsolutePath());
        mp=MediaPlayer.create(this,path);
        mp.setLooping(true);
        mp.start();

        play.setImageResource(R.drawable.pause);
        songname.setText(Songs.current.get(index).getName());

    }

    @Override
    public void onBackPressed() {
        mp.stop();
        finish();

    }

}

```



### 3.Playlist Activity

```
package com.example.mediaplayer_assignment;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.util.Log;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ListAdapter;
import android.widget.ListView;

public class playlist extends AppCompatActivity {
    ListView listview;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_playlist);
        Intent intent=getIntent();

        int counter=intent.getIntExtra("counter",-1);
        String[] playlist_names=new String[counter];
        SharedPreferences prefs=getSharedPreferences("playlistnames",MODE_PRIVATE);

        for (int i=0;i<counter;i++)
            playlist_names[i]=prefs.getString(""+i,"-1");

        listview=findViewById(R.id.listView);
        Intent intent1=new Intent(this,MainActivity.class);

        ListAdapter listAdapter= new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1,playlist_names);
        listview.setAdapter(listAdapter);
        listview.setOnItemClickListener(
            new AdapterView.OnItemClickListener() {
                @Override
```

```

        public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
            intent1.putExtra("name",playlist_names[position]);
            startActivity(intent1);
            finish();
        }
    }

    );

}

}

```

## Songs Class

```

package com.example.mediaplayer_assignment;

import java.io.File;
import java.util.ArrayList;
import java.util.List;

public class Songs {
    public static ArrayList<File> files;
    public static ArrayList<File> current;

    public static List<File> getListFiles(File parentDir) {
        ArrayList<File> inFiles = new ArrayList<File>();
        File[] files = parentDir.listFiles();
        for (File file : files) {
            if (file.isDirectory()) {
                inFiles.addAll(getListFiles(file));
            } else {
                if (file.getName().endsWith(".mp3")) {
                    inFiles.add(file);
                }
            }
        }
        return inFiles;
    }
}

```

## Menu

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto">

    <item android:title="Add"
          android:icon="@drawable/ic_baseline_add_24"
          android:id="@+id/addsong"
          app:showAsAction="always"

          android:visible="false"
        ></item>

    <item android:title="check"
          android:icon="@drawable/ic_baseline_check_24"
          android:id="@+id/check"
          app:showAsAction="always"
          android:visible="false"

        ></item>

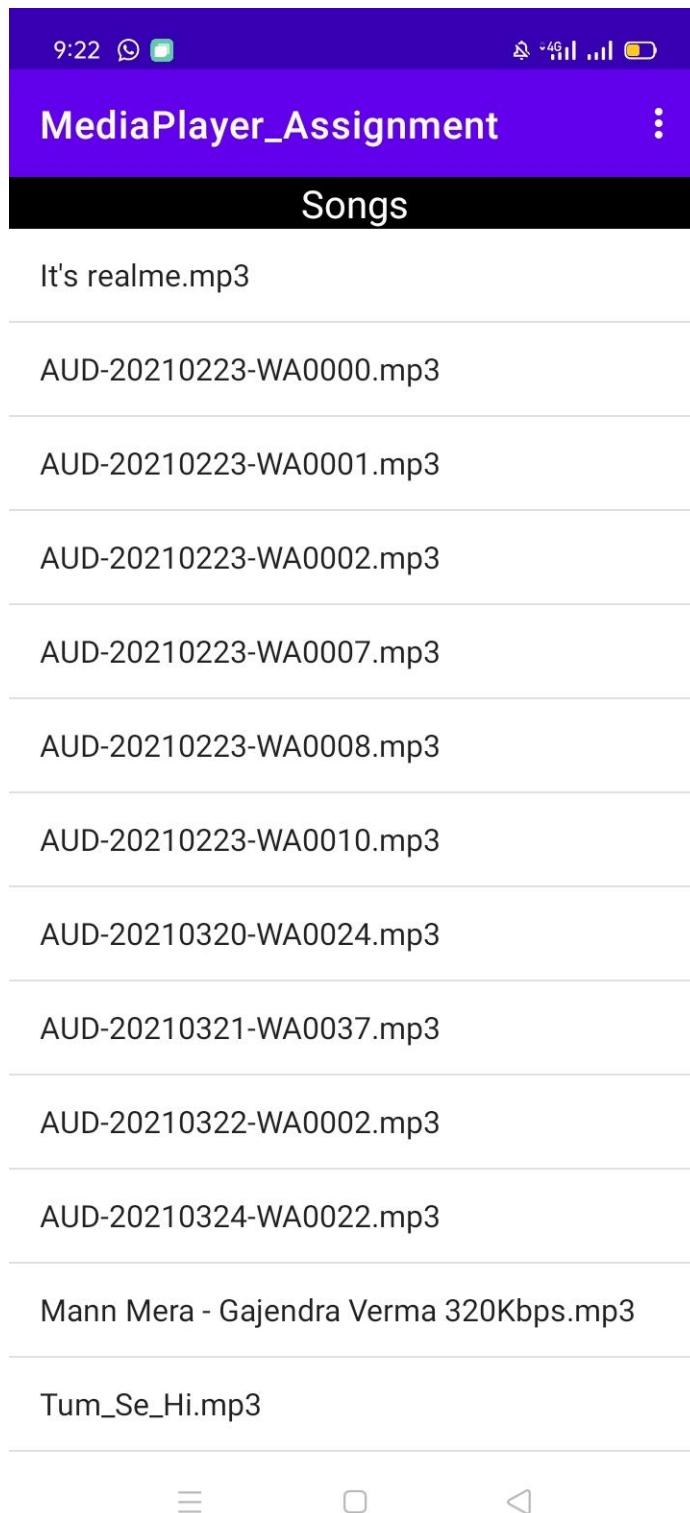
    <item
        android:title="New Playlist"
        android:id="@+id/addplaylist"
        android:icon="@drawable/addicon"

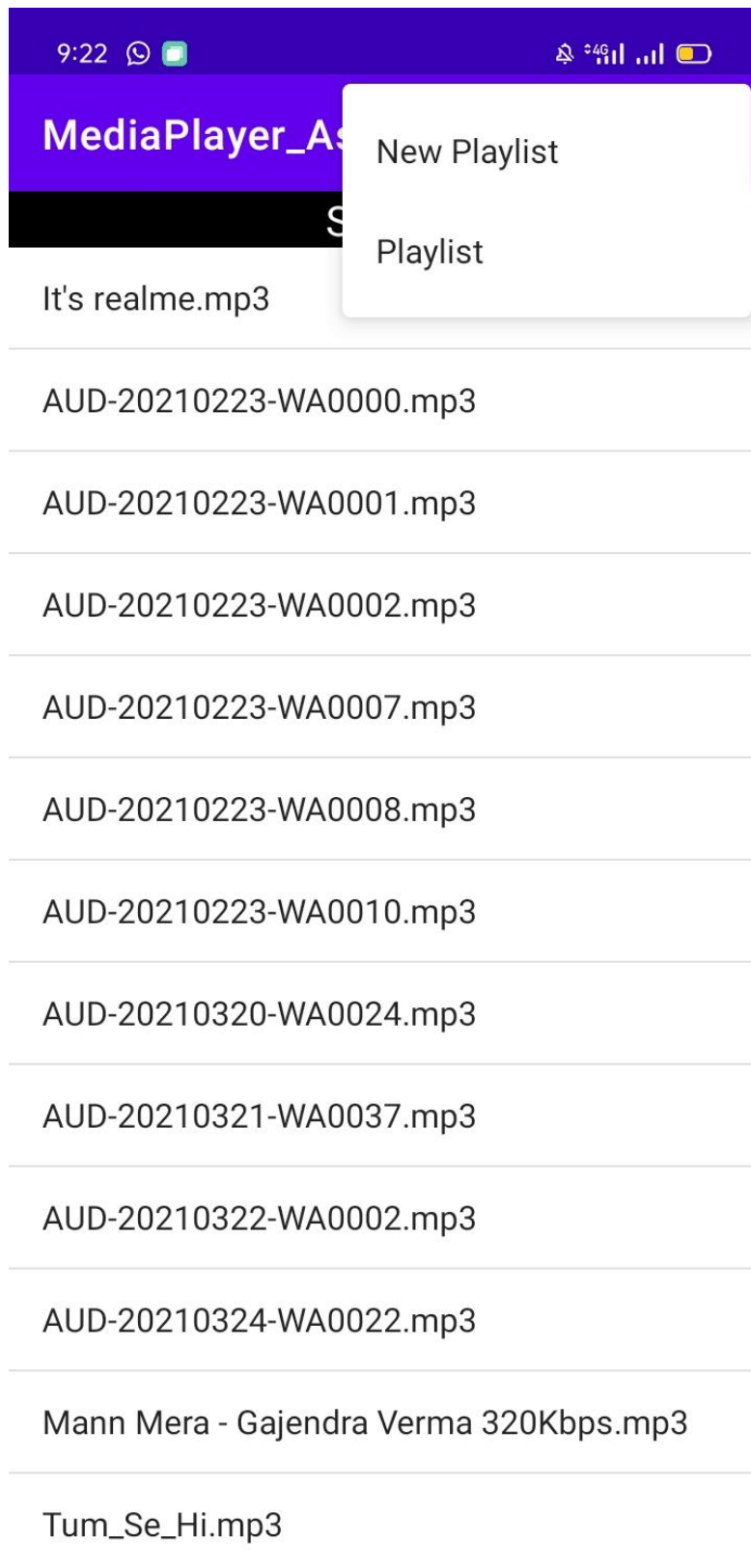
    ></item>

    <item
        android:title="Playlist"
        android:id="@+id/playlist"
        android:icon="@drawable/playlist_icon"
    ></item>

</menu>
```

## Output





9:42



MediaPlayer\_Assig...



love songs

AUD-20210320-WA0024.mp3



AUD-20210321-WA0037.mp3



AUD-20210322-WA0002.mp3



AUD-20210324-WA0022.mp3



Mann Mera - Gajendra Verma  
320Kbps.mp3



Tum\_Se\_Hi.mp3



Khud\_Se\_Zyada.mp3



Tera\_Ban\_Jaunga.mp3



Kaise\_Hua.mp3



Maikada\_|\_Muhammad\_Samie\_[HD]  
(128k).mp3



Paniyon\_Sa.mp3



Khairiyat.mp3



Pehli\_Dafa.mp3



9:42 📶 4G 📶 🔋

MediaPlayer\_Assig... + ✓ ⋮

love songs

Tum\_Se\_Hi.mp3

---

Khud\_Se\_Zyada.mp3

---

Tera\_Ban\_Jaunga.mp3



9:23



## MediaPlayer\_Assignment



Tera\_Ban\_Jaunga.mp3

2:13

- 1:43

