|  |  |
| --- | --- |
| **Ex No: 5** | **CREATE A MEDIA PLAYER APPLICATION** |
| **Date:** | **12/10/2023** |

**AIM:**

To create a Media Player Application.

**DESCRIPTION:**

A RecyclerView is a popular and flexible widget in Android that is used to display a scrollable list or grid of items in a more efficient and memory-friendly way compared to the older ListView. It is a fundamental component for creating dynamic and interactive user interfaces in Android apps.

RecyclerView is a versatile and efficient UI component in Android that facilitates the creation of dynamic, scrollable lists or grids of items. Its key advantages include efficient item recycling, flexibility in layout management, support for item animations, and ease of use with adapters, making it a crucial element for creating responsive and memory-efficient Android apps.

Key Components of the RecyclerView:

Adapter- The Adapter is the bridge between your data source and the RecyclerView. It's responsible for creating ViewHolders and binding data to them. To use the RecyclerView, you need to create a custom Adapter class that extends RecyclerView.Adapter. This class defines how your data is presented in the RecyclerView and how to respond to user interactions.

ViewHolder**-** ViewHolders are lightweight objects that hold references to the views within each item of the RecyclerView. They play a crucial role in optimizing performance by recycling views as the user scrolls. By caching references to views, ViewHolders reduce the need for findViewById calls, which can be resource-intensive.

**PROGRAM:**

XML Code-

**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">  
  
 <TextView  
 android:id="@+id/cp\_label"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/currently\_playing"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.116"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.045" />  
  
 <TextView  
 android:id="@+id/s\_name"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text=""  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.158"  
 app:layout\_constraintStart\_toEndOf="@+id/cp\_label"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.045" />  
  
 <ImageButton  
 android:id="@+id/fast\_reverse"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.081"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.947"  
 app:srcCompat="@android:drawable/ic\_media\_rew"  
 android:contentDescription="@string/previous" />  
  
 <ImageButton  
 android:id="@+id/fast\_forward"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:contentDescription="@string/forward"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.896"  
 app:layout\_constraintStart\_toEndOf="@+id/fast\_reverse"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.947"  
 app:srcCompat="@android:drawable/ic\_media\_ff" />  
  
 <ImageButton  
 android:id="@+id/pause\_btn"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="48dp"  
 android:contentDescription="@string/pause"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/fast\_forward"  
 app:layout\_constraintHorizontal\_bias="0.079"  
 app:layout\_constraintStart\_toEndOf="@+id/fast\_reverse"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.947"  
 app:srcCompat="@android:drawable/ic\_media\_pause" />  
  
 <ImageButton  
 android:id="@+id/play\_btn"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:contentDescription="@string/play"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/fast\_forward"  
 app:layout\_constraintHorizontal\_bias="0.748"  
 app:layout\_constraintStart\_toEndOf="@+id/fast\_reverse"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.947"  
 app:srcCompat="@android:drawable/ic\_media\_play" />  
  
 <TextView  
 android:id="@+id/current\_time"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/\_0\_min\_0\_sec"  
 android:textSize="15sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.076"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.837" />  
  
 <TextView  
 android:id="@+id/song\_time"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/\_0min\_0sec"  
 android:textSize="15sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.901"  
 app:layout\_constraintStart\_toEndOf="@+id/current\_time"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.837" />  
  
 <SeekBar  
 android:id="@+id/seekBar"  
 android:layout\_width="180dp"  
 android:layout\_height="16dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/song\_time"  
 app:layout\_constraintHorizontal\_bias="0.433"  
 app:layout\_constraintStart\_toEndOf="@+id/current\_time"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.832" />  
</androidx.constraintlayout.widget.ConstraintLayout>

Java Code-

**MainActivity.java**

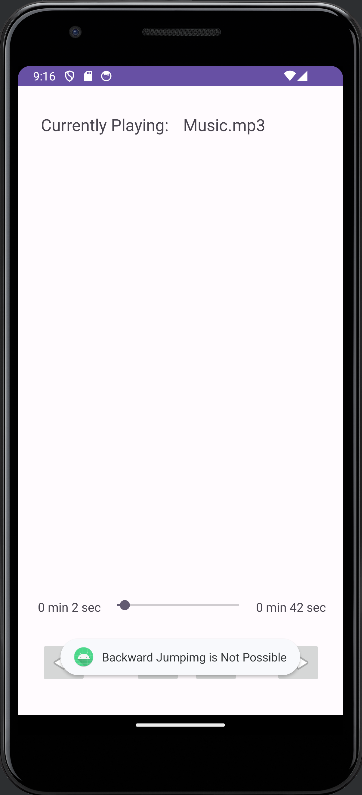
package com.example.madex5;  
import androidx.appcompat.app.AppCompatActivity;  
import android.media.MediaPlayer;  
import android.os.Bundle;  
import android.os.Handler;  
import android.view.View;  
import android.widget.ImageButton;  
import android.widget.SeekBar;  
import android.widget.TextView;  
import android.widget.Toast;  
import java.util.concurrent.TimeUnit;

public class MainActivity extends AppCompatActivity {  
 private ImageButton f\_btn,b\_btn,play\_btn,pause\_btn;  
 private MediaPlayer mplayer;  
 private TextView song\_name, current\_time, song\_time;  
 private SeekBar song\_progress;  
 private static int *otime*=0, *stime*=0, *etime*=0, *ftime*=5000, *btime*=5000;  
 private Handler hdlr=new Handler();  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 f\_btn=findViewById(R.id.*fast\_forward*);  
 b\_btn=findViewById(R.id.*fast\_reverse*);  
 play\_btn=findViewById(R.id.*play\_btn*);  
 pause\_btn=findViewById(R.id.*pause\_btn*);  
 song\_name=findViewById(R.id.*s\_name*);  
 current\_time=findViewById(R.id.*current\_time*);  
 song\_time=findViewById(R.id.*song\_time*);  
 song\_name.setText("Music.mp3");  
 mplayer=MediaPlayer.*create*(this,R.raw.*music*);  
 song\_progress=findViewById(R.id.*seekBar*);  
 song\_progress.setClickable(false);  
 pause\_btn.setClickable(false);  
 play\_btn.setOnClickListener(new View.OnClickListener(){  
  
 @Override  
 public void onClick(View view) {  
 Toast.*makeText*(MainActivity.this,"Playing Audio", Toast.*LENGTH\_SHORT*).show();  
 mplayer.start();  
 *etime*=mplayer.getDuration();  
 *stime*=mplayer.getCurrentPosition();  
 if (*otime*==0){  
 song\_progress.setMax(*etime*);  
 *otime*=1;  
 }  
 song\_time.setText(String.*format*("%d min %d sec", TimeUnit.*MILLISECONDS*.toMinutes(*etime*),TimeUnit.*MILLISECONDS*.toSeconds(*etime*)-TimeUnit.*MINUTES*.toSeconds(TimeUnit.*MILLISECONDS*.toMinutes(*etime*))));  
 current\_time.setText(String.*format*("%d min %d sec", TimeUnit.*MILLISECONDS*.toMinutes(*stime*),TimeUnit.*MILLISECONDS*.toSeconds(*stime*)-TimeUnit.*MINUTES*.toSeconds(TimeUnit.*MINUTES*.toMinutes(*stime*))));  
 song\_progress.setProgress(*stime*);  
 hdlr.postDelayed(UpdateSongTime, 100);  
 pause\_btn.setEnabled(true);  
 play\_btn.setEnabled(false);  
 }  
 });  
 pause\_btn.setOnClickListener(new View.OnClickListener(){  
  
 @Override  
 public void onClick(View view) {  
 mplayer.pause();  
 pause\_btn.setEnabled(false);  
 play\_btn.setEnabled(true);  
 Toast.*makeText*(getApplicationContext(),"Audio Paused",Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
 f\_btn.setOnClickListener(new View.OnClickListener(){  
  
 @Override  
 public void onClick(View view) {  
 if ((*stime*+*ftime*)<=*etime*){  
 *stime*=*stime*+*ftime*;  
 mplayer.seekTo(*stime*);  
 }  
 else{  
 Toast.*makeText*(getApplicationContext(),"Forward Jumping is not Possible",Toast.*LENGTH\_SHORT*).show();  
 }  
 if(!play\_btn.isEnabled()){  
 play\_btn.setEnabled(true);  
 }  
 }  
 });  
 b\_btn.setOnClickListener(new View.OnClickListener(){  
  
 @Override  
 public void onClick(View view) {  
 if((*stime*-*btime*)>0){  
 *stime*=*stime*-*btime*;  
 mplayer.seekTo(*stime*);  
 }  
 else{  
 Toast.*makeText*(getApplicationContext(),"Backward Jumpimg is Not Possible",Toast.*LENGTH\_SHORT*).show();  
 }  
 if (!play\_btn.isEnabled()){  
 play\_btn.setEnabled(true);  
 }  
 }  
 });  
 }  
 private Runnable UpdateSongTime=new Runnable(){  
  
 @Override  
 public void run() {  
 *stime*=mplayer.getCurrentPosition();  
 current\_time.setText(String.*format*("%d min %d sec",TimeUnit.*MILLISECONDS*.toMinutes(*stime*),TimeUnit.*MILLISECONDS*.toSeconds(*stime*)-TimeUnit.*MINUTES*.toSeconds(TimeUnit.*MILLISECONDS*.toMinutes(*stime*))));  
 song\_progress.setProgress(*stime*);  
 hdlr.postDelayed(this,100);  
 }  
 };  
}

**OUTPUT:**

 A black rectangular frame with a white screen

Description automatically generated

**RESULT:**

Thus, the experiment for creating a Media Player Application has been coded, compiled, and executed successfully.