**MediaPlayer**

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout 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:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

android:paddingBottom="@dimen/activity\_vertical\_margin" tools:context=".MainActivity">

<TextView android:text="Music Palyer" android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/textview"

android:textSize="35dp"

android:layout\_alignParentTop="true"

android:layout\_centerHorizontal="true" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Media Player"

android:id="@+id/textView"

android:layout\_below="@+id/textview"

android:layout\_centerHorizontal="true"

android:textColor="#ff7aff24"

android:textSize="35dp" />

<ImageView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/imageView"

android:layout\_below="@+id/textView"

android:layout\_centerHorizontal="true"

android:src="@drawable/abc"/>

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/forward"

android:id="@+id/button"

android:layout\_alignParentBottom="true"

android:layout\_alignParentLeft="true"

android:layout\_alignParentStart="true" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/pause"

android:id="@+id/button2"

android:layout\_alignParentBottom="true"

android:layout\_alignLeft="@+id/imageView"

android:layout\_alignStart="@+id/imageView" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/back"

android:id="@+id/button3"

android:layout\_alignTop="@+id/button2"

android:layout\_toRightOf="@+id/button2"

android:layout\_toEndOf="@+id/button2" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/rewind"

android:id="@+id/button4"

android:layout\_alignTop="@+id/button3"

android:layout\_toRightOf="@+id/button3"

android:layout\_toEndOf="@+id/button3" />

<SeekBar

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/seekBar"

android:layout\_alignLeft="@+id/textview"

android:layout\_alignStart="@+id/textview"

android:layout\_alignRight="@+id/textview"

android:layout\_alignEnd="@+id/textview"

android:layout\_above="@+id/button" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceSmall"

android:text="Small Text"

android:id="@+id/textView2"

android:layout\_above="@+id/seekBar"

android:layout\_toLeftOf="@+id/textView"

android:layout\_toStartOf="@+id/textView" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceSmall"

android:text="Small Text"

android:id="@+id/textView3"

android:layout\_above="@+id/seekBar"

android:layout\_alignRight="@+id/button4"

android:layout\_alignEnd="@+id/button4" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceMedium"

android:text="Medium Text"

android:id="@+id/textView4"

android:layout\_alignBaseline="@+id/textView2"

android:layout\_alignBottom="@+id/textView2"

android:layout\_centerHorizontal="true" />

</RelativeLayout>

package com.example.demomediaplay;

import android.media.MediaPlayer;

import android.os.Handler;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.ImageView;

import android.widget.SeekBar;

import android.widget.TextView;

import android.widget.Toast;

import java.util.concurrent.TimeUnit;

public class MainActivity extends AppCompatActivity {

private Button b1,b2,b3,b4;

private ImageView iv;

private MediaPlayer mediaPlayer;

private double startTime = 0;

private double finalTime = 0;

private Handler myHandler = new Handler();;

private int forwardTime = 5000;

private int backwardTime = 5000;

private SeekBar seekbar;

private TextView tx1,tx2,tx3;

public static int oneTimeOnly = 0;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

b1 = (Button) findViewById(R.id.button);

b2 = (Button) findViewById(R.id.button2);

b3 = (Button)findViewById(R.id.button3);

b4 = (Button)findViewById(R.id.button4);

iv = (ImageView)findViewById(R.id.imageView);

tx1 = (TextView)findViewById(R.id.textView2);

tx2 = (TextView)findViewById(R.id.textView3);

tx3 = (TextView)findViewById(R.id.textView4);

tx3.setText("Song.mp3");

mediaPlayer = MediaPlayer.create(this, R.raw.song);

seekbar = (SeekBar)findViewById(R.id.seekBar);

seekbar.setClickable(false);

b2.setEnabled(false);

b3.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Toast.makeText(getApplicationContext(), "Playing sound",Toast.LENGTH\_SHORT).show();

mediaPlayer.start();

finalTime = mediaPlayer.getDuration();

startTime = mediaPlayer.getCurrentPosition();

if (oneTimeOnly == 0) {

seekbar.setMax((int) finalTime);

oneTimeOnly = 1;

}

tx2.setText(String.format("%d min, %d sec",

TimeUnit.MILLISECONDS.toMinutes((long) finalTime),

TimeUnit.MILLISECONDS.toSeconds((long) finalTime) -

TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.toMinutes((long)

finalTime)))

);

tx1.setText(String.format("%d min, %d sec",

TimeUnit.MILLISECONDS.toMinutes((long) startTime),

TimeUnit.MILLISECONDS.toSeconds((long) startTime) -

TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.toMinutes((long)

startTime)))

);

seekbar.setProgress((int)startTime);

myHandler.postDelayed(UpdateSongTime,100);

b2.setEnabled(true);

b3.setEnabled(false);

}

});

b2.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Toast.makeText(getApplicationContext(), "Pausing sound",Toast.LENGTH\_SHORT).show();

mediaPlayer.pause();

b2.setEnabled(false);

b3.setEnabled(true);

}

});

b1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

int temp = (int)startTime;

if((temp+forwardTime)<=finalTime){

startTime = startTime + forwardTime;

mediaPlayer.seekTo((int) startTime);

Toast.makeText(getApplicationContext(),"You have Jumped forward 5 seconds",Toast.LENGTH\_SHORT).show();

}else{

Toast.makeText(getApplicationContext(),"Cannot jump forward 5 seconds",Toast.LENGTH\_SHORT).show();

}

}

});

b4.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

int temp = (int)startTime;

if((temp-backwardTime)>0){

startTime = startTime - backwardTime;

mediaPlayer.seekTo((int) startTime);

Toast.makeText(getApplicationContext(),"You have Jumped backward 5 seconds",Toast.LENGTH\_SHORT).show();

}else{

Toast.makeText(getApplicationContext(),"Cannot jump backward 5 seconds",Toast.LENGTH\_SHORT).show();

}

}

});

}

private Runnable UpdateSongTime = new Runnable() {

public void run() {

startTime = mediaPlayer.getCurrentPosition();

tx1.setText(String.format("%d min, %d sec",

TimeUnit.MILLISECONDS.toMinutes((long) startTime),

TimeUnit.MILLISECONDS.toSeconds((long) startTime) -

TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.

toMinutes((long) startTime)))

);

seekbar.setProgress((int)startTime);

myHandler.postDelayed(this, 100);

}

};

}

**ViewPagger and TabLayout**

<?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"

xmlns:app="http://schemas.android.com/apk/res-auto"

android:orientation="vertical"

app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"

tools:context="com.example.newviewpandtab.MainActivity">

<android.support.design.widget.TabLayout

android:id="@+id/tabs"

android:layout\_width="match\_parent"

android:background="#cdc36e"

android:layout\_height="40dp"

app:tabMode="fixed"

app:tabGravity="fill"

app:tabTextColor="#FF4081" />

<android.support.v4.view.ViewPager

android:id="@+id/viewpager"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

/>

</LinearLayout>

<fragment1>

<fragment2>

package com.example.newviewpandtab;

import android.support.design.widget.TabLayout;

import android.support.v4.app.Fragment;

import android.support.v4.app.FragmentManager;

import android.support.v4.app.FragmentPagerAdapter;

import android.support.v4.view.ViewPager;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

TabLayout tabLayout;

ViewPager viewPager;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

viewPager=(ViewPager)findViewById(R.id.viewpager);

tabLayout = (TabLayout)findViewById(R.id.tabs);

viewPager.setAdapter(new CustomAdapter(getSupportFragmentManager()));

tabLayout.setupWithViewPager(viewPager);

}

private class CustomAdapter extends FragmentPagerAdapter {

private String titles[]={"Fragment 1","Fragment 2"};

public CustomAdapter(FragmentManager supportFragmentManager) {

super(supportFragmentManager);

}

@Override

public Fragment getItem(int position) {

switch (position){

case 0:

return new Fragment1();

case 1:

return new Fragment2();

}

return null;

}

@Override

public int getCount() {

return titles.length;

}

@Override

public CharSequence getPageTitle(int position) {

return titles[position];

}

}

}

**OpenActivity Using Thread**

package com.example.openactivityusingthread;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

Thread th = new Thread() {

@Override

public void run() {

try {

Thread.sleep(5000);

} catch (InterruptedException e)

{

e.printStackTrace();

}

Intent intent = new Intent(MainActivity.this, NextActivity.class);

startActivity(intent);

}

};

th.start();

}

}

**ProgressBar**

<?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"

tools:context=".MainActivity"

android:orientation="vertical">

<ProgressBar

android:id="@+id/progressBar"

style="?android:attr/progressBarStyleHorizontal"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignParentLeft="true"

android:layout\_alignParentTop="true"

android:layout\_marginLeft="23dp"

android:layout\_marginTop="20dp"

android:indeterminate="false"

android:max="100"

android:minHeight="50dp"

android:minWidth="200dp"

android:progress="1" />

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/progressBar"

android:layout\_below="@+id/progressBar"/>

<Button

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Open Activity"

android:layout\_marginLeft="23dp"

android:id="@+id/OpenActivity"

/>

</LinearLayout>

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:id="@+id/rl"

android:layout\_height="match\_parent"

android:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context="com.example.prograssbarexample.Next">

<ProgressBar

android:id="@+id/pb"

android:layout\_width="250dp"

android:layout\_height="wrap\_content"

android:visibility="gone"

style="@android:style/Widget.Holo.ProgressBar.Horizontal"

android:indeterminate="true"

/>

<TextView

android:id="@+id/tv"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:visibility="gone"

android:layout\_toRightOf="@id/pb"

android:textSize="15dp"

android:fontFamily="sans-serif-condensed"

android:textColor="#ff2f33"

android:layout\_marginLeft="10dp"

/>

<Button

android:id="@+id/btn"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Start Operation"

android:layout\_below="@id/tv"

android:layout\_marginTop="25dp"

/>

</RelativeLayout>

package com.example.prograssbarexample;

import android.content.Intent;

import android.os.Handler;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.ProgressBar;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

private ProgressBar progressBar;

private int progressStatus = 0;

private TextView textView;

private Handler handler = new Handler();

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main); progressBar = (ProgressBar) findViewById(R.id.progressBar);

textView = (TextView) findViewById(R.id.textView);

// Start long running operation in a background thread

new Thread(new Runnable() {

public void run() {

while (progressStatus < 100) {

progressStatus += 1;

// Update the progress bar and display the

//current value in the text view

handler.post(new Runnable() {

public void run() {

progressBar.setProgress(progressStatus);

textView.setText(progressStatus+"/"+progressBar.getMax());

}

});

try {

// Sleep for 200 milliseconds.

Thread.sleep(200);

} catch (InterruptedException e) {

e.printStackTrace();

}

}

}

}).start();

Button button=(Button)findViewById(R.id.OpenActivity);

button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent intent=new Intent(MainActivity.this,Next.class);

startActivity(intent);

}

});

}

}

package com.example.prograssbarexample;

import android.os.Handler;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.RelativeLayout;

import android.widget.TextView;

import android.widget.ProgressBar;

public class Next extends AppCompatActivity {

private int progressStatus = 0;

private Handler handler = new Handler();

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_next);

final RelativeLayout rl = (RelativeLayout) findViewById(R.id.rl);

final Button btn = (Button) findViewById(R.id.btn);

final TextView tv = (TextView) findViewById(R.id.tv);

final ProgressBar pb = (ProgressBar) findViewById(R.id.pb);

btn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

// Set the progress status zero on each button click

progressStatus = 0;

// Visible the progress bar and text view

pb.setVisibility(View.VISIBLE);

tv.setVisibility(View.VISIBLE);

// Start the lengthy operation in a background thread

new Thread(new Runnable() {

@Override

public void run() {

while(progressStatus < 100){

// Update the progress status

progressStatus +=1;

// Try to sleep the thread for 20 milliseconds

try{

Thread.sleep(20);

}catch(InterruptedException e){

e.printStackTrace();

}

// Update the progress bar

handler.post(new Runnable() {

@Override

public void run() {

pb.setProgress(progressStatus);

// Show the progress on TextView

tv.setText(progressStatus+"");

// If task execution completed

if(progressStatus == 100){

// Hide the progress bar from layout after finishing task

pb.setVisibility(View.GONE);

// Set a message of completion

tv.setText("Operation completed...");

}

}

});

}

}

}).start(); // Start the operation

}

});

}

}

**SplashActivity**

package com.example.sgs.testscheduling;

import android.content.Context;

import android.content.Intent;

import android.os.Handler;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

public class SplashActivty extends AppCompatActivity {

Context context;

private Handler handler;

Runnable runnable;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_splash);

context = SplashActivty.this;

handler = new Handler();

runnable = new Runnable() {

@Override

public void run() {

context.startActivity(new Intent(context, MainActivity.class));

finish();

}

};

handler.postDelayed(runnable, 5000);

}

@Override

public void onBackPressed() {

handler.removeCallbacks(runnable);

super.onBackPressed();

}

}

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="com.example.sgs.testscheduling" >

<application

android:allowBackup="true"

android:icon="@mipmap/ic\_launcher"

android:label="@string/app\_name"

android:supportsRtl="true"

android:theme="@style/AppTheme" >

<activity android:name=".MainActivity" >

</activity>

<activity

android:name=".SplashActivty"

android:label="@string/title\_activity\_splash"

android:theme="@style/AppTheme.NoActionBar" >

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

</application>

</manifest>