

**ANDROID OPERATING SYSTEM**

**LABORATORY**

***Submitted by***

**Name:-**

**Reg. No:-**

*In partial fulfillment for the award of the degree* **BACHELOR OF COMPUTER APPLICATION**

*In the branch of study*

**MACT**

**SEMESTER: IV**

**ACADEMIC YEAR: 2017-2018**

**TABLE OF CONTENTS**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **SL.No** |  |  | **DATE** |  |  | **TITLE OF THE EXPERIMENTS** |  |  | **PAGE** | | |  |  |  | **FACULTY'S** | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  | **NO** | |  |  | **SIGNATURE** | | |  |
|  | |  |  |  |  |  | | |  |  |  |  |  |  |  |  |  |  |
| 1 | |  |  |  |  | Display Hello World | | |  |  |  |  |  |  |  |  |  |  |
|  | |  |  |  |  |  | | |  |  |  |  |  |  |  |  |  |  |
| 2 | |  |  |  |  | Add two Edit Texts. When a number | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | is Entered in EditText1 The square | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | of that number should be displayed | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | in EditText2 | | |  |  |  |  |  |  |  |  |  |  |
|  | |  |  |  |  |  | | |  |  |  |  |  |  |  |  |  |  |
| 3 | |  |  |  |  | Add an Edit Text and a Button. | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | When the button is clicked, the text | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | inputted in Edit Text should be | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | retrieved and displayed back to the | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | user | | |  |  |  |  |  |  |  |  |  |  |
| 4 | |  |  |  |  | Add two Edit Texts. When the | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | button is clicked, the text inputted in | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | EditText1 should be retrieved and | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | displayed back to the user in | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | EditText2 | | |  |  |  |  |  |  |  |  |  |  |
| 5 | |  |  |  |  | Program a calculator | | |  |  |  |  |  |  |  |  |  |  |
|  | |  |  |  |  |  | | |  |  |  |  |  |  |  |  |  |  |
| 6 | |  |  |  |  | Create a Unit convertor for Height | | |  |  |  |  |  |  |  |  |  |  |
|  | |  |  |  |  |  | | |  |  |  |  |  |  |  |  |  |  |
| 7 | |  |  |  |  | Create a Unit convertor for Height | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | and weight in the same application. | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | Selection of height and weight can | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | be done using a Spinner | | |  |  |  |  |  |  |  |  |  |  |
|  | |  |  |  |  |  | | |  |  |  |  |  |  |  |  |  |  |
| 8 | |  |  |  |  | Add a spinner. When the spinner is | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | selected there should be different | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | options. When you click on each | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | option, it should go to another page. | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | Each of these pages should have a | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | “Back” button, which on pressing | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | will take you back to the page with | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | the spinner. | | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 9 |  | Create applications to include Action |  |  |
|  |  | Bar, Menus, Dialogs and |  |  |
|  |  | Notifications |  |  |
| 10 |  | Create a user login form and |  |  |
|  |  | registration form in database |  |  |
|  |  |  |  |  |
| 11 |  | Create a camera application, where |  |  |
|  |  | you can click a picture and then save |  |  |
|  |  | it as the wallpaper. |  |  |
|  |  |  |  |  |
| 12 |  | Create a media player which plays |  |  |
|  |  | an mp3 song. |  |  |
|  |  |  |  |  |
| 13 |  | Create a media player which records |  |  |
|  |  | the sound. |  |  |
|  |  |  |  |  |
| 14 |  | Testing application in Android |  |  |
|  |  |  |  |  |

**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 1**



**Aim:**

* Display hello world

**Learning Objective:**

* + Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE:**

<?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="com.king.cia1.MainActivity">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Hello World!"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>

1. **MANIFEST FILE:**

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

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

package="com.king.cia1">

<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/AppTheme"> <activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

</application>

</manifest>

1. **JAVA CODE:** package com.king.cia1;

import android.app.Activity;

import android.os.Bundle;

public class MainActivity extends Activity {

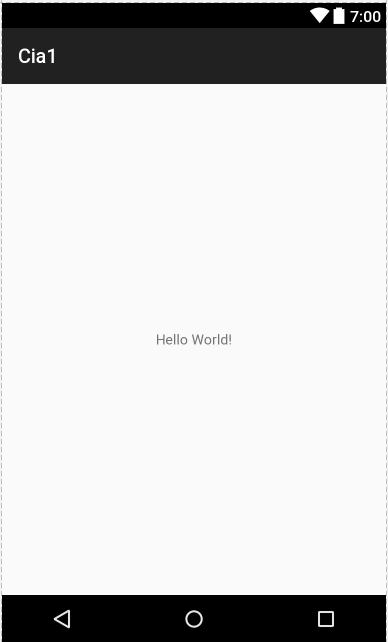
@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

}

}

1. **OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 2**



**Aim:**

* Add two Edit Texts. When a number is Entered in Edittext\_1 The square of that number should be displayed in Edittext\_2



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE**

<?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" android:orientation="vertical" tools:context="com.king.cia1.MainActivity">

<EditText android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:id="@+id/et1" android:hint="ENTER NUMBER"/>

<EditText android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:id="@+id/et2" android:hint="DISPLAYING"/>

</LinearLayout>

1. **MANIFEST FILE:**

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

<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.king.cia1">

<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/AppTheme"> <activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

</application>

</manifest>

1. **JAVA CODE:**

package com.king.cia1; import android.app.Activity; import android.os.Bundle; import android.text.Editable; import android.text.TextWatcher; import android.widget.EditText;

public class MainActivity extends Activity { EditText e1,e2;

int num; @Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main); e1=(EditText)findViewById(R.id.et1); e2=(EditText)findViewById(R.id.et2);

e1.addTextChangedListener(new TextWatcher() { @Override

public void beforeTextChanged(CharSequence charSequence, int i, int i1, int i2) {

}

@Override

public void onTextChanged(CharSequence charSequence, int i, int i1, int i2) { num=Integer.parseInt(e1.getText().toString().trim());

}

@Override

public void afterTextChanged(Editable editable) { e2.setText(""+num\*num);

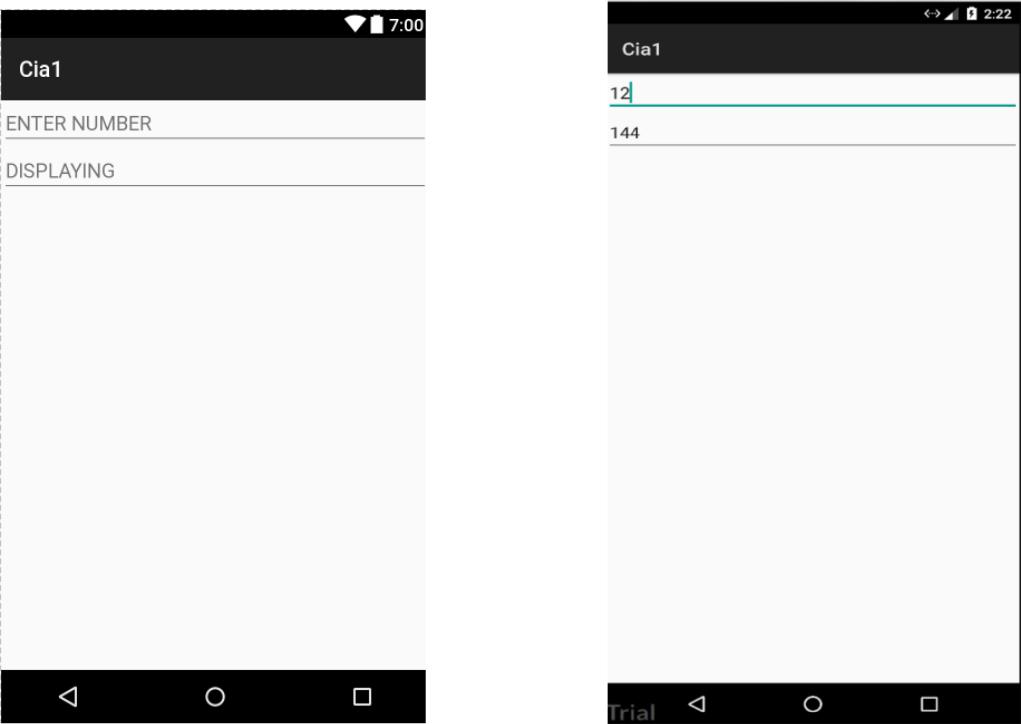
}

});

}

}

1. **OUTPUT**:



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 3**



**Aim:**

* Add an Edit Text and a Button. When the button is clicked, the text inputted in Edit Text should be retrieved and displayed back to the user



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE**

<?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"

android:orientation="vertical"

tools:context="com.king.cia1.MainActivity">

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/et1"

android:hint="ENTER NUMBER"/>

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/et2"

android:hint="DISPLAYING"/>

</LinearLayout>

1. **MANIFEST FILE:**

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

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

package="com.king.cia1">

<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/AppTheme"> <activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

</application>

</manifest>

1. **JAVA CODE:**

package tvk.spc.mact3;

import android.support.v7.app.AppCompatActivity; import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

EditText etip;

TextView tvOutput;

Button btnClickME;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

etip = findViewById(R.id.editTextInput); tvOutput = findViewById(R.id.textViewResult); btnClickME = findViewById(R.id.buttonClickMe);

btnClickME.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) {

tvOutput.setText(etip.getText());

}

});

}

}

1. **OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 4**



**Aim:**

 Add two Edit Texts. When the button is clicked, the text inputted in EditText1 should be retrieved and displayed back to the user in EditText2



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE**

<?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" android:orientation="vertical" tools:context="com.king.cia1.MainActivity">

<EditText android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:id="@+id/et1" android:hint="ENTER NUMBER"/>

<EditText android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:id="@+id/et2" android:hint="DISPLAYING"/>

</LinearLayout>

1. **MANIFEST FILE:**

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

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

package="com.king.cia1">

<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/AppTheme"> <activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

</application>

</manifest>

1. **JAVA CODE:**

package tvk.spc.mact4;

import android.support.v7.app.AppCompatActivity; import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

EditText etip , etop;

Button btnClickME;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

etip = findViewById(R.id.editTextInput); etop = findViewById(R.id.editTextOutput); btnClickME = findViewById(R.id.buttonClickMe);

btnClickME.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) {

etop.setText(etip.getText());

}

});

}

}

**D. OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 5**



**Aim:**

* Program a calculator



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE:**

*<?*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"

android:orientation="vertical"

tools:context="com.king.cia2.MainActivity">

<EditText

android:id="@+id/edt1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Number1"

android:gravity="center"

/>

<EditText

android:id="@+id/edt2"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Number2"

android:gravity="center"

/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:weightSum="100"

>

<Button

android:id="@+id/b1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="+"

android:layout\_weight="25"/>

<Button

android:id="@+id/b2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="-"

android:layout\_weight="25"

/>

<Button

android:id="@+id/b3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="\*"

android:layout\_weight="25" />

<Button

android:id="@+id/b4"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="/"

android:layout\_weight="25"/>

</LinearLayout>

<Button

android:id="@+id/b5"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Clear"/>

<TextView

android:id="@+id/t1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="RESULT"

android:textSize="32dp"

android:gravity="center"

/>

</LinearLayout>

1. **MANIFEST CODE:**

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

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

package="com.king.cia2">

<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/AppTheme">

<activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

</application>

</manifest>

1. **JAVA CODE:**

package com.king.cia2;

import android.app.Activity;

import android.media.MediaPlayer;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

public class MainActivity extends Activity implements View.OnClickListener { Button add,sub,mul,div,clear;

EditText e1,e2;

TextView tv;

@Override

public void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.*activity\_main*);

e1=(EditText)findViewById(R.id.*edt1*);

e2=(EditText)findViewById(R.id.*edt2*);

tv=(TextView)findViewById(R.id.*t1*);

add=(Button)findViewById(R.id.*b1*);

add.setOnClickListener(this);

sub=(Button)findViewById(R.id.*b2*);

sub.setOnClickListener(this);

mul=(Button)findViewById(R.id.*b3*);

mul.setOnClickListener(this);

div=(Button)findViewById(R.id.*b4*);

div.setOnClickListener(this);

clear=(Button)findViewById(R.id.*b5*);

clear.setOnClickListener(this);

}

@Override

public void onClick(View view) {

double v1=Double.*parseDouble*(e1.getText().toString().trim()); double v2=Double.*parseDouble*(e1.getText().toString().trim());

switch (view.getId()) {

case R.id.*b1*:

tv.setText(""+(v1+v2));

break;

case R.id.*b2*:

tv.setText(""+(v1-v2));

break;

case R.id.*b3*:

tv.setText(""+(v1\*v2));

break;

case R.id.*b4*:

tv.setText(""+(v1/v2));

break;

case R.id.*b5*:

tv.setText("");

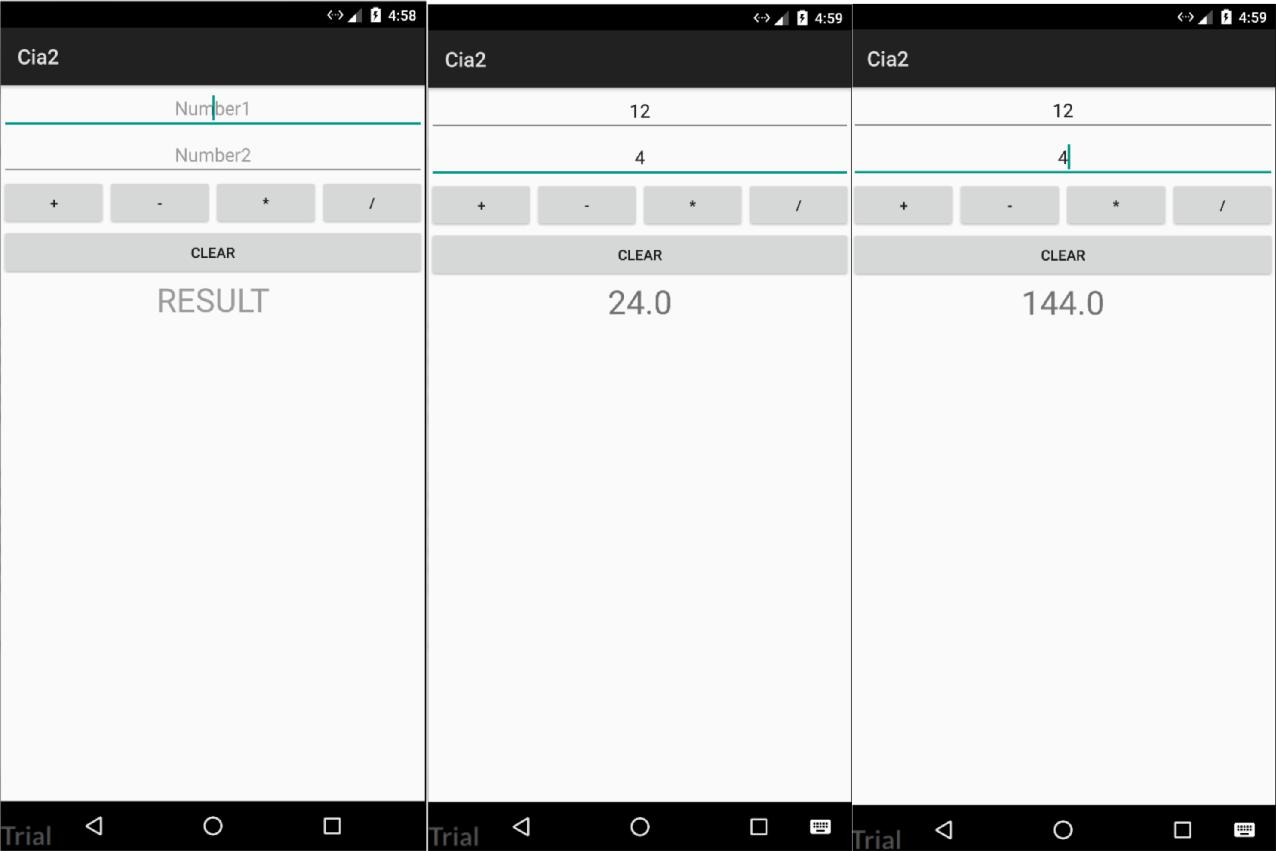
break;

}

}

}

1. **OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 6**



**Aim:**

* Create a Unit Convertor for Height



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **JAVA CODE 1:**

package spc.tvk.mact6;

import android.app.Activity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Spinner;

import android.widget.TextView;

import java.util.ArrayList;

import java.util.List;

public class MainActivity extends Activity implements AdapterView.OnItemSelectedListener {

Spinner spinner;

EditText et\_input;

Button btn\_submit;

TextView tv\_display;

Double n2,n1;

String item;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

// Spinner element

spinner = (Spinner) findViewById(R.id.spinner);

et\_input = (EditText) findViewById(R.id.ET\_input); btn\_submit= (Button) findViewById(R.id.BTN\_submit); tv\_display=(TextView) findViewById(R.id.TV\_display);

* Spinner click listener spinner.setOnItemSelectedListener(this);
* Spinner Drop down elements

List<String> categories = new ArrayList<String>(); categories.add("select options"); categories.add("Centimetre"); categories.add("Metre");

// Creating adapter for spinner

ArrayAdapter<String> dataAdapter = new ArrayAdapter<String>(this, android.R.layout.simple\_spinner\_item, categories);

dataAdapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_it em);

* attaching data adapter to spinner spinner.setAdapter(dataAdapter);

}

@Override

public void onItemSelected(AdapterView<?> parent, View view, int position, long id)

{

item = parent.getItemAtPosition(position).toString();

}

public void onNothingSelected(AdapterView<?> adapterView) {

}

public void activity(View view)

{

// Accepting the value from the user

n1= Double.parseDouble(et\_input.getText().toString().trim());

* Conversion switch(item)

{

case "Centimetre": n2=n1/10; break;

case "Metre": n2=n1/1000; break;

}

tv\_display.setText(" "+n2);

* Switching to second Activity

Intent i = new Intent(this,Main2Activity.class); i.putExtra("Result",n2);

startActivity(i);

}

}

1. **JAVA CODE 2:** package spc.tvk.mact6;

import android.app.Activity; import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.Button; import android.widget.TextView;

public class Main2Activity extends Activity implements View.OnClickListener {

Double res;

TextView tv\_display,tv\_title;

Button b1;

Intent i;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main2);

tv\_display=(TextView) findViewById(R.id.TV\_display);

tv\_title=(TextView) findViewById(R.id.TV\_title);

res = getIntent().getExtras().getDouble("Result");

tv\_display.setText(" "+res);

tv\_title.setText("converted value");

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

b1.setOnClickListener(this);

i= new Intent(this, MainActivity.class);

}

public void onClick(View view) {

if(view.getId()==R.id.btn)

{

startActivity(i);

}

}

}

1. **MANIFEST FILE:**

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

<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.king.cia2">

<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/AppTheme">

<activity android:name=".MainActivity">

<intent-filter>

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

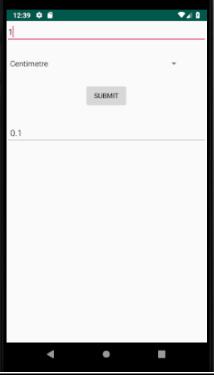
<category android:name="android.intent.category.LAUNCHER" /> </intent-filter>

</activity>

</application>

</manifest>

1. **OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 7**



**Aim:**

* Create a Unit Convertor for Height and Weight in the same application. Selection of height and weight can be done using a Spinner.



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

JAVA CODE 1:

package tvk.spc.mact7;

import android.app.Activity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Spinner;

import android.widget.TextView;

import java.util.ArrayList;

import java.util.List;

public class MainActivity extends Activity implements AdapterView.OnItemSelectedListener {

Spinner spinner;

EditText et\_input;

Button btn\_submit;

TextView tv\_display;

Double n2,n1;

String item;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

// Spinner element

spinner = (Spinner) findViewById(R.id.spinner);

et\_input = (EditText) findViewById(R.id.ET\_input); btn\_submit= (Button) findViewById(R.id.BTN\_submit); tv\_display=(TextView) findViewById(R.id.TV\_display);

* Spinner click listener spinner.setOnItemSelectedListener(this);
* Spinner Drop down elements

List<String> categories = new ArrayList<String>(); categories.add("select options"); categories.add("Centimetre");

categories.add("Metre");

categories.add("Kilometre");

categories.add("Gram");

categories.add("Kilogram");

// Creating adapter for spinner

ArrayAdapter<String> dataAdapter = new ArrayAdapter<String>(this, android.R.layout.simple\_spinner\_item, categories);

dataAdapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

* attaching data adapter to spinner spinner.setAdapter(dataAdapter);

}

@Override

public void onItemSelected(AdapterView<?> parent, View view, int position, long id) { item = parent.getItemAtPosition(position).toString();

}

public void onNothingSelected(AdapterView<?> adapterView) {

}

public void activity(View view)

{

// Accepting the value from the user

n1= Double.parseDouble(et\_input.getText().toString().trim());

* Conversion switch(item)

{

case "Centimetre": n2=n1/10; break;

case "Metre": n2=n1/1000; break;

case "Kilometre": n2=n1/1000000; break;

case "Gram": n2=n1/100; break;

case "Kilogram": n2=n1/1000000; break;

}

tv\_display.setText(" "+n2);

// Switching to second Activity

Intent i = new Intent(this,Main2Activity.class); i.putExtra("Result",n2);

startActivity(i);

}

}

JAVA CODE 2:

package tvk.spc.mact7;

import android.app.Activity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

public class Main2Activity extends Activity implements View.OnClickListener {

Double res;

TextView tv\_display,tv\_title;

Button b1;

Intent i;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main2);

tv\_display=(TextView) findViewById(R.id.TV\_display);

tv\_title=(TextView) findViewById(R.id.TV\_title);

res = getIntent().getExtras().getDouble("Result");

tv\_display.setText(" "+res);

tv\_title.setText("converted value");

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

b1.setOnClickListener(this);

i= new Intent(this, MainActivity.class);

}

public void onClick(View view) {

if(view.getId()==R.id.btn)

{

startActivity(i);

}

}

}

**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 8**



**Aim:**

* Add a spinner. When the spinner is selected there should be different options. When you click on each option, it should go to another page. Each of these pages should have a “Back” button, which on pressing will take you back to the page with the spinner.



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE:**

**XML ACTVITY 1**

*<?*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"

android:orientation="vertical"

tools:context="com.king.cia3.MainActivity">

<EditText

android:id="@+id/ET\_input"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Enter value" />

<Spinner

android:id="@+id/spinner"

android:layout\_marginTop="30dp"

android:layout\_width="368dp"

android:layout\_height="wrap\_content"/>

<Button

android:id="@+id/BTN\_submit"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="30dp"

android:layout\_gravity="center\_horizontal"

android:onClick="activity"

android:text="SUBMIT"/>

<EditText

android:id="@+id/TV\_display"

android:layout\_marginTop="30dp"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Converted Value" />

</LinearLayout>

**XML ACTIVITY 2:**

*<?*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"

tools:context="com.king.cia3.Main2">

<TextView

android:id="@+id/TV\_title"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:textSize="30dp"/>

<TextView

android:id="@+id/TV\_display"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:textSize="30dp"/>

<Button

android:id="@+id/btn"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="BACK"/>

</LinearLayout>

1. **MANIFEST FILE:**

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

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

package="com.king.cia3">

<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/AppTheme">

<activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

<activity android:name=".Main2"></activity> </application>

</manifest>

1. **JAVA CODE: ANDROID ACTIVITY:**

package spc.tvk.mact8;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity; import android.os.Bundle;

import android.view.View;

public class AndroidActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_android);

}

public void backFunction(View view) {

Intent i = new Intent(AndroidActivity.this, MainActivity.class); startActivity(i);

}

}

**JAVA ACTIVITY:**

package spc.tvk.mact8;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity; import android.os.Bundle;

import android.view.View;

public class JavaActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_java);

}

public void backFunction(View view) {

Intent i = new Intent(JavaActivity.this, MainActivity.class); startActivity(i);

}

}

**MAIN ACTIVITY:**

package spc.tvk.mact8;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity; import android.os.Bundle;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Spinner;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements AdapterView.OnItemSelectedListener{

String[] domainArray = { "select any domain..", "Android","JAVA", "Testing"};

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

Spinner spin = (Spinner) findViewById(R.id.spinnerDomainSelector); spin.setOnItemSelectedListener(this);

ArrayAdapter aa = new ArrayAdapter (this,android.R.layout.simple\_spinner\_item,domainArray);

aa.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

spin.setSelection(0);

spin.setAdapter(aa);

}

//Performing action onItemSelected and onNothing selected @Override

public void onItemSelected(AdapterView<?> arg0, View arg1, int position, long id) {

if (position ==1) {

Intent intentAndroid = new

Intent(MainActivity.this, AndroidActivity.class);

startActivity(intentAndroid);

}

if (position == 2) {

Intent intentJava = new

Intent(MainActivity.this, JavaActivity.class);

startActivity(intentJava);

}

if (position == 3) {

Intent intentTesting = new

Intent(MainActivity.this, TestingActivity.class);

startActivity(intentTesting);

}

}

@Override

public void onNothingSelected(AdapterView<?> arg0) { // TODO Auto-generated method stub

}

}

**TESTING ACTIVITY:**

package spc.tvk.mact8;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity; import android.os.Bundle;

import android.view.View;

public class TestingActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_testing);

}

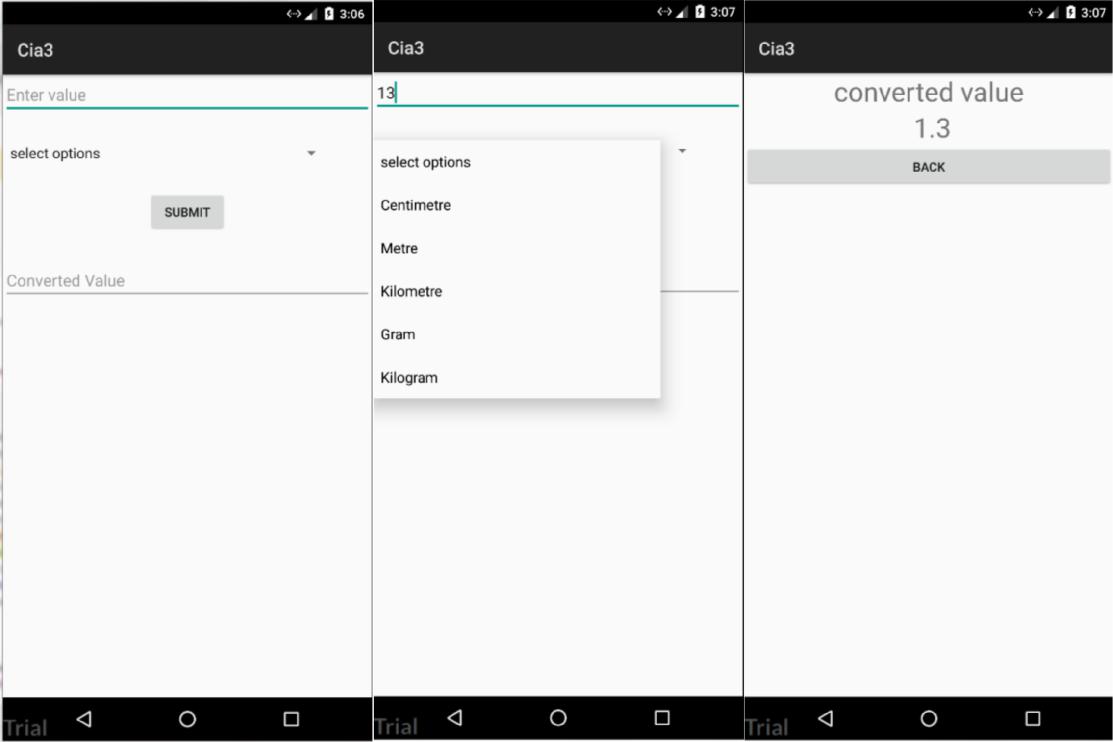
public void backFunction(View view) {

Intent i = new Intent(TestingActivity.this, MainActivity.class); startActivity(i);

}

}

1. **OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 9**



**Aim:**

* Create applications to include Action Bar, Menus, Dialogs and Notifications



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE:**

*<?*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="com.king.cia4.MainActivity">

</LinearLayout>

**MENU.XML CODE:**

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

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

<item

android:id="@+id/menu\_map"

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

android:title="Map"/>

<item

android:id="@+id/menu\_camera"

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

android:title="Camera"/>

<item

android:id="@+id/menu\_player"

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

android:title="Media Player"/>

<item

android:id="@+id/menu\_settings"

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

android:title="Settings"/>

<item

android:id="@+id/menu\_jain"

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

android:title="Jain"/>

</menu>

1. **MANIFEST FILE:**

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

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

package="com.king.cia4">

<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/AppTheme"> <activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

</application>

</manifest>

1. **JAVA CODE:**

package com.king.cia4;

import android.app.Activity;

import android.content.Intent;

import android.os.Bundle;

import android.provider.MediaStore;

import android.view.Menu;

import android.view.MenuInflater;

import android.view.MenuItem;

import android.widget.Toast;

public class MainActivity extends Activity {

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.*activity\_main*);

}

public boolean onCreateOptionsMenu(Menu menu){ MenuInflater inflater = getMenuInflater(); inflater.inflate(R.menu.*menu*,menu);

return true;

}

public boolean onOptionsItemSelected(MenuItem item){ switch (item.getItemId())

{

case R.id.*menu\_camera*:

Toast.*makeText*(this,"Camera Clicked",Toast.*LENGTH\_LONG*).show();

Intent camera = new Intent(MediaStore.*ACTION\_IMAGE\_CAPTURE*);

startActivity(camera);

return true;

case R.id.*menu\_map*:

Toast.*makeText*(this,"Map Clicked",Toast.*LENGTH\_LONG*).show(); return true;

case R.id.*menu\_player*:

Toast.*makeText*(this,"Media Player

Clicked",Toast.*LENGTH\_LONG*).show();

return true;

case R.id.*menu\_settings*:

Toast.*makeText*(this,"Settings Clicked",Toast.*LENGTH\_LONG*).show(); return true;

default:

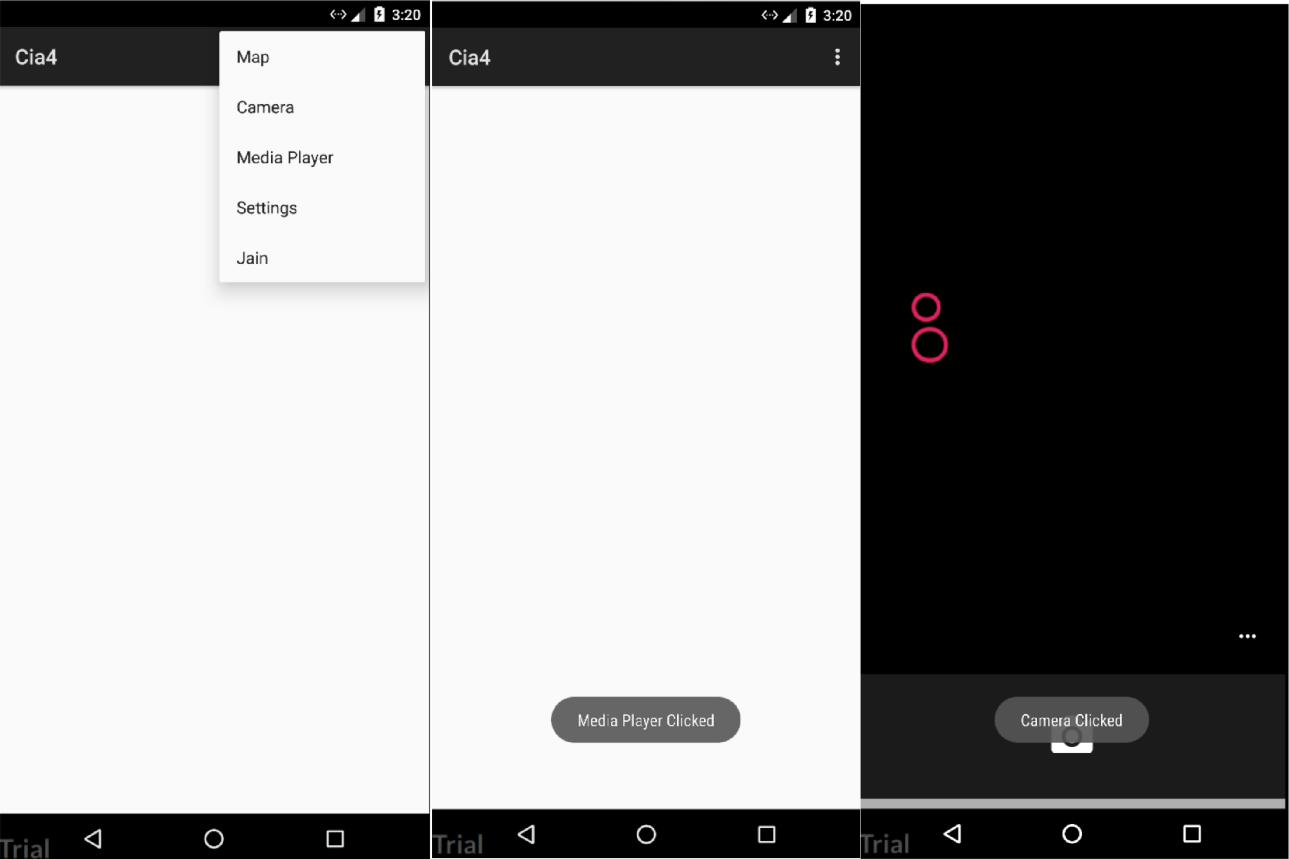
Toast.*makeText*(this,"KING PROGRAM",Toast.*LENGTH\_LONG*).show(); return super.onOptionsItemSelected(item);

}

}

}

1. **OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 10**



**Aim:**

* Create a User Login form and registration form using a Database



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE:**

*<?*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"

android:orientation="vertical"

tools:context="com.king.cia5.MainActivity">

<RelativeLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content">

<TextView

android:id="@+id/TV\_Rollno"

android:textSize="32dp"

android:text="ROLL NO: "

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" />

<EditText

android:id="@+id/ET\_Rollno"

android:layout\_toRightOf="@+id/TV\_Rollno"

android:hint="Rollno"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" />

<TextView

android:id="@+id/TV\_Name"

android:textSize="32dp"

android:layout\_below="@+id/TV\_Rollno"

android:text="NAME: "

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" />

<EditText

android:id="@+id/ET\_Name"

android:layout\_toRightOf="@+id/TV\_Name"

android:layout\_below="@id/ET\_Rollno"

android:hint="Name"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" />

<TextView

android:id="@+id/TV\_Marks"

android:textSize="32dp"

android:layout\_below="@+id/TV\_Name"

android:text="MARKS: "

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" />

<EditText

android:id="@+id/ET\_Marks"

android:layout\_toRightOf="@+id/TV\_Marks"

android:layout\_below="@+id/ET\_Name"

android:hint="Marks"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" />

</RelativeLayout>

<Button

android:id="@+id/BTN\_Add"

android:onClick="onClick"

android:text="ADD"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" />

<Button

android:id="@+id/BTN\_ViewAll"

android:onClick="onClick"

android:text="View All"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" />

</LinearLayout>

1. **MANIFEST FILE:**

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

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

package="com.king.cia5">

<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/AppTheme">

<activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

</application>

</manifest>

1. **JAVA CODE:**

import android.app.Activity;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase; import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

public class MainActivity extends Activity implements View.OnClickListener { EditText editRollno,editName,editMarks;

Button add,dis;

SQLiteDatabase db;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.*activity\_main*); editRollno=(EditText)findViewById(R.id.*ET\_Rollno*); editName=(EditText)findViewById(R.id.*ET\_Name*); editMarks=(EditText)findViewById(R.id.*ET\_Marks*); add=(Button)findViewById(R.id.*BTN\_Add*); add.setOnClickListener(this); dis=(Button)findViewById(R.id.*BTN\_ViewAll*); dis.setOnClickListener(this);

db=openOrCreateDatabase("StudentDB", Context.*MODE\_PRIVATE*, null); db.execSQL("CREATE TABLE IF NOT EXISTS student(rollno VARCHAR,name

VARCHAR,marks VARCHAR);");

}

@Override

public void onClick(View view) {

if(view.getId()==R.id.*BTN\_Add*)

{

if(editRollno.getText().toString().trim().length()==0|| editName.getText().toString().trim().length()==0|| editMarks.getText().toString().trim().length()==0)

{

Toast.*makeText*(this,"Error"+ "Please enter all values",Toast.*LENGTH\_SHORT*).show();

return;

}

db.execSQL("INSERT INTO student

VALUES('"+editRollno.getText()+"','"+editName.getText()+

"','"+editMarks.getText()+"');");

Toast.*makeText*(this,"Success "+ "Record added",Toast.*LENGTH\_SHORT*).show(); clearText();

}

else if(view.getId()==R.id.*BTN\_ViewAll*)

{

Cursor c=db.rawQuery("SELECT \* FROM student", null); if(c.getCount()==0)

{

Toast.*makeText*(this,"Error "+ "No records found",Toast.*LENGTH\_SHORT*).show();

return;

}

StringBuffer buffer=new StringBuffer();

while(c.moveToNext())

{

buffer.append("Rollno: "+c.getString(0)+"\n");

buffer.append("Name: "+c.getString(1)+"\n");

buffer.append("Marks: "+c.getString(2)+"\n\n");

}

Toast.*makeText*(this,"Student Details"+"\n"+ buffer.toString(),Toast.*LENGTH\_LONG*).show();

}

}

public void clearText()

{

editRollno.setText("");

editName.setText("");

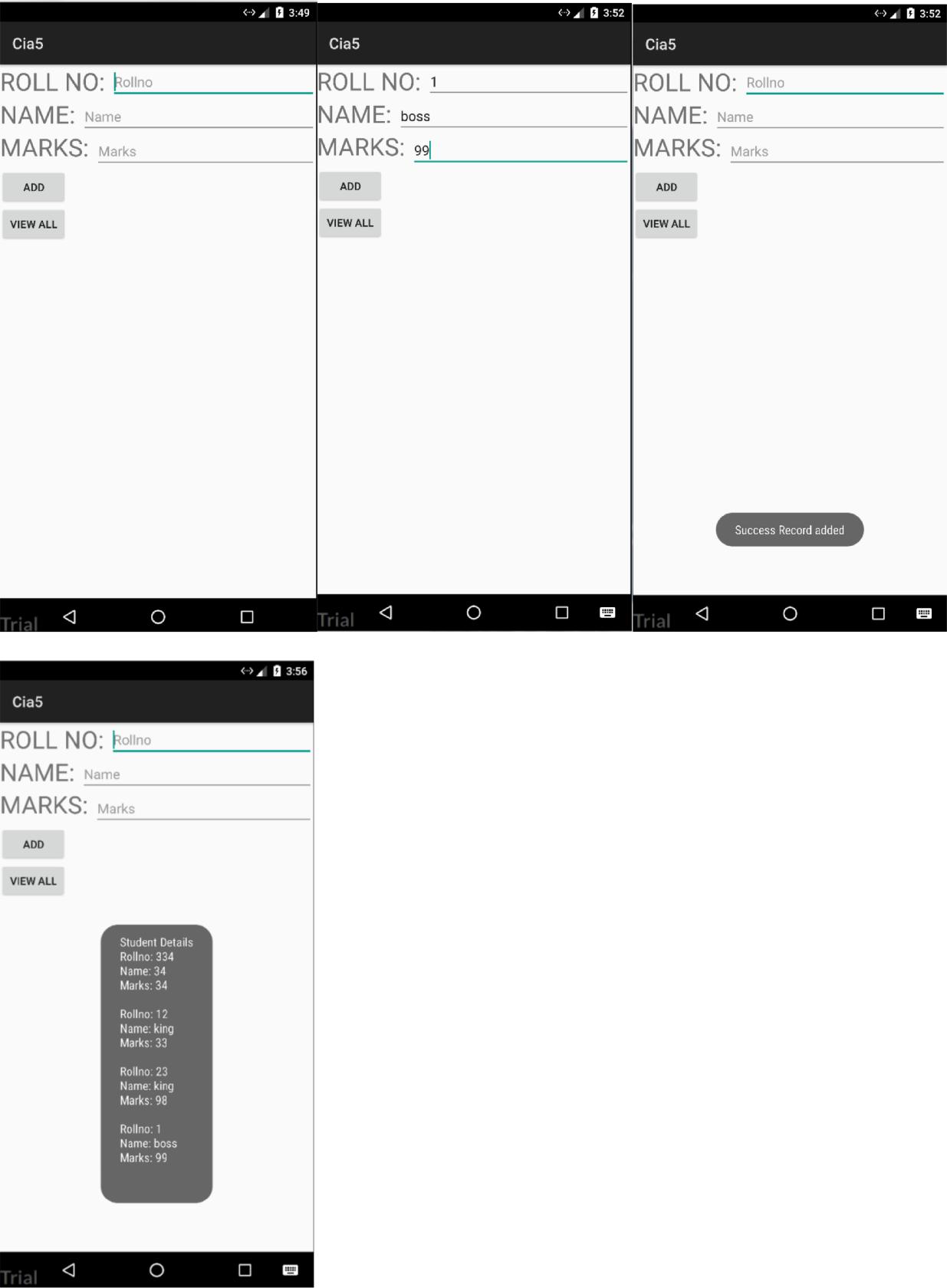
editMarks.setText("");

editRollno.requestFocus();

}

}

1. **OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 11**



**Aim:**

* Create a camera application, where you can click a picture and then save it as the wallpaper.



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE:**

*<?*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"

android:orientation="vertical"

tools:context="com.king.cia6.MainActivity">

<Button

android:id="@+id/button2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="TAKE PHOTO"

android:layout\_gravity="center"/>

<ImageView

android:id="@+id/imageView"

android:layout\_width="300dp"

android:layout\_height="300dp"

android:layout\_centerHorizontal="true"

android:layout\_centerVertical="true"

/>

</LinearLayout>

1. **MANIFEST CODE:**

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

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

package="com.king.cia6">

<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/AppTheme">

<activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

</application>

</manifest>

1. **JAVA CODE:**

package com.king.cia6;

import android.app.Activity;

import android.content.Intent;

import android.graphics.Bitmap;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.ImageView;

public class MainActivity extends Activity {

private static final int *CAMERA\_REQUEST* = 1888; private ImageView imageView;

@Override

public void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.*activity\_main*);

this.imageView = (ImageView)this.findViewById(R.id.*imageView*); Button photoButton = (Button) this.findViewById(R.id.*button2*); photoButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent cameraIntent = new

Intent(android.provider.MediaStore.*ACTION\_IMAGE\_CAPTURE*); startActivityForResult(cameraIntent, *CAMERA\_REQUEST*);

}

});

}

protected void onActivityResult(int requestCode, int resultCode, Intent data) {

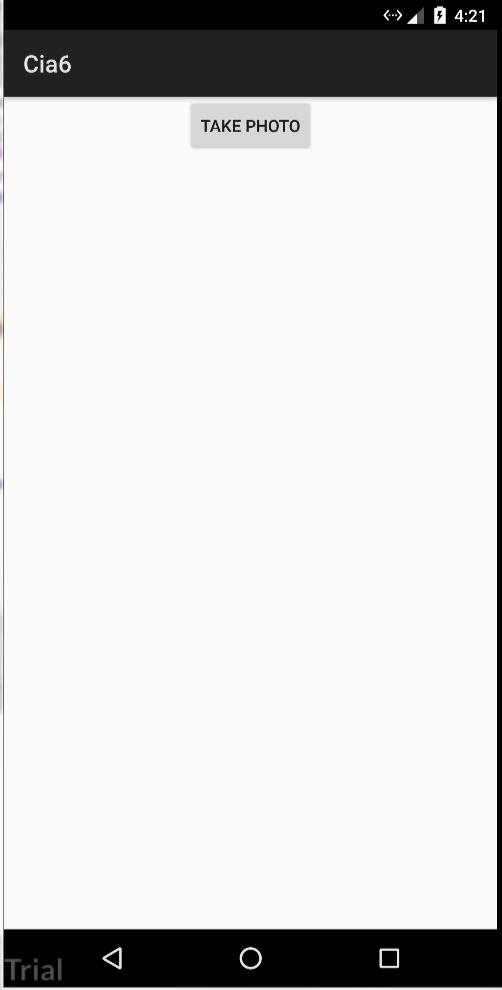
if (requestCode == *CAMERA\_REQUEST* && resultCode == Activity.*RESULT\_OK*) { Bitmap photo = (Bitmap) data.getExtras().get("data"); imageView.setImageBitmap(photo);

}

}

}

1. **OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 12**



**Aim:**

* Create a media player, which plays an mp3 song.



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE:**

*<?*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"

android:orientation="vertical"

tools:context="com.king.cia7.MainActivity">

<Button

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Music"

android:id="@+id/music"/>

<Button

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Stop"

android:id="@+id/stop"/>

</LinearLayout>

1. **MANIFEST FILE:**

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

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

package="com.king.cia7">

<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/AppTheme">

<activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

</application>

</manifest>

1. **JAVA CODE:**

package tvk.spc.mact12;

import android.media.MediaPlayer;

import android.support.v7.app.AppCompatActivity; import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

MediaPlayer mp = null;

Button btnPlay, btnStop;

@Override

public void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

mp = MediaPlayer.create(this, R.raw.backgroundmusic);

btnPlay = findViewById(R.id.play);

btnStop = findViewById(R.id.stop);

btnPlay.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { Toast.makeText(getApplicationContext(), "Playing

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

mp.start();

}

});

btnStop.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { Toast.makeText(getApplicationContext(), "Stop playing

music",Toast.LENGTH\_SHORT).show();

mp.stop();

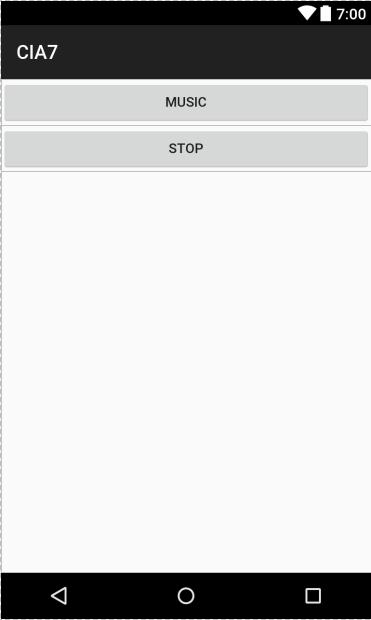
}

});

}

}

1. **OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 13**



**Aim:**

* Create a media player, which records an mp3 song.



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **XML CODE:**

*<?*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"

android:orientation="vertical"

tools:context="com.king.cia7.MainActivity">

<Button

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Music"

android:id="@+id/music"/>

<Button

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Stop"

android:id="@+id/stop"/>

</LinearLayout>

1. **MANIFEST FILE:**

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

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

package="com.king.cia7">

<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/AppTheme">

<activity android:name=".MainActivity">

<intent-filter>

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

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

</activity>

</application>

</manifest>

1. **JAVA CODE:**

package tvk.spc.mact13;

import android.media.MediaPlayer;

import android.media.MediaRecorder;

import android.os.Environment;

import android.support.v7.app.AppCompatActivity; import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.Toast;

import java.io.IOException;

import java.util.Random;

import static android.Manifest.permission.RECORD\_AUDIO;

import static android.Manifest.permission.WRITE\_EXTERNAL\_STORAGE; import android.support.v4.app.ActivityCompat;

import android.content.pm.PackageManager; import android.support.v4.content.ContextCompat;

public class MainActivity extends AppCompatActivity {

Button buttonStart, buttonStop, buttonPlayLastRecordAudio, buttonStopPlayingRecording ;

String AudioSavePathInDevice = null;

MediaRecorder mediaRecorder ;

Random random ;

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);

buttonStart = (Button) findViewById(R.id.button); buttonStop = (Button) findViewById(R.id.button2); buttonPlayLastRecordAudio = (Button) findViewById(R.id.button3); buttonStopPlayingRecording = (Button)findViewById(R.id.button4);

buttonStop.setEnabled(false); buttonPlayLastRecordAudio.setEnabled(false); buttonStopPlayingRecording.setEnabled(false);

random = new Random();

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();

} catch (IllegalStateException e) { e.printStackTrace();

} catch (IOException e) {

e.printStackTrace();

}

buttonStart.setEnabled(false);

buttonStop.setEnabled(true);

Toast.makeText(MainActivity.this, "Recording started", Toast.LENGTH\_LONG).show();

} 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.OutputFormat.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);

}

@Override

public void onRequestPermissionsResult(int requestCode,

String permissions[], int[] 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);

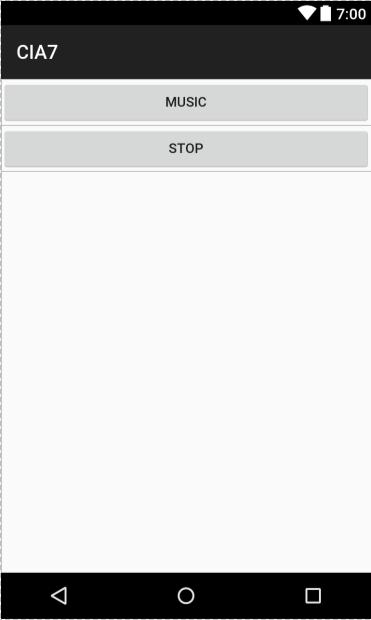
int result1 = ContextCompat.checkSelfPermission(getApplicationContext(), RECORD\_AUDIO);

return result == PackageManager.PERMISSION\_GRANTED && result1 == PackageManager.PERMISSION\_GRANTED;

}

}

1. **OUTPUT:**



**ANDROID OPERATING SYSTEM**

**LAB RECORD**

**LAB EXERCISE – 14**



**Aim:**

* Testing in Android



**Learning Objective:**

* Through this activity we will come to know how UI and Listeners are implemented in Android.



**Prerequisites:**

* Android Studio



**Standard Procedure:**

* XML Code
* Java Code
* Manifest Code

1. **JAVA CODE:**

package com.king.cia8;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import static org.junit.Assert.\*;

*/\*\**

* *Example local unit test, which will execute on the development machine (host).*
* *@see <a href="http://d.android.com/tools/testing">Testing documentation</a> \*/*

public class ExampleUnitTest {

int a;

@Before

public void default()

{

a=2;

}

@Test

public void addition\_isCorrect() throws Exception { *assertEquals*(4, 2 + a);

}

@Test

public void subtraction\_isCorrect() throws Exception { *assertEquals*(4, 6 - a);

}

@After

public void display(){

System.*out*.println("tested");

}

}