Sabarivasan V
205001085
CSE-B

Ex. No. 5 Android Application using Multithreading

Aim:

Develop an android application to perform multithreading. Define 3 threads to run concurrently when the “start” button is clicked.

The first thread should change the color of the text indefinitely
The second thread should implement a moving banner
The third thread should display a counter starting from 0 to 1000

When the “Stop” button is pressed all the threads should be stopped

Layouts Used: None. Three textViews.

Code:

MainActivity.java:

```
package com.example.ex5;


import androidx.appcompat.app.AppCompatActivity;


import android.graphics.Color;

import android.os.Bundle;
import android.util.Log;

import android.view.View;
```

```
import android.widget.Button;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        TextView t1 = findViewById(R.id.t1);
        Thread1 th1 = new Thread1(t1);

        TextView t2 = findViewById(R.id.t2);
        Thread2 th2 = new Thread2(t2);

        TextView t3 = findViewById(R.id.t3);
        Thread3 th3 = new Thread3(t3);
        final boolean[] init = {false};
        Button start = findViewById(R.id.start);
        start.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
```

```
        if(!init[0]){
            th1.start();
            th2.start();
            th3.start();
            init[0] = true;
        }
        else{
            Log.d("debug","hello");
            th1.pause(false);
            th2.pause(false);
            th3.pause(false);
        }



    }
});
Button stop = findViewById(R.id.stop);
stop.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        th1.pause(true);
        th2.pause(true);
        th3.pause(true);
```

```
            }
        });


    }
}
```

Thread1.java:

```
package com.example.ex5;


import android.graphics.Color;
import android.util.Log;
import android.widget.TextView;
public class Thread1 extends Thread{
    TextView t;
    int red = 120;
    int green = 120;
    int blue = 120;
    boolean paused = false;
    Object lock = new Object();
    Thread1(TextView t){
        this.t=t;
    }
```

```
public void pause(boolean paused){
    synchronized (lock){
        if(paused)
        this.paused = true;
        else{
            this.paused = false;
            lock.notifyAll();
        }
    }
    Log.d("Debug",""+paused);
}

public void run(){

    while(true) {
        try {
            int color = Color.rgb(red, green, blue);
            t.setTextColor(color);

            red = (red + 20) % 255;
            green = (green + 10) % 255;
            blue = (blue + 5) % 255;
```

```
            Thread.sleep(500);


            synchronized (lock){
                while(paused){
                    try{
                        lock.wait();
                    }catch(InterruptedException e){
                    }
                }
            }
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }


}
```

Thread2.java:

```
package com.example.ex5;


import android.util.Log;
```

```
import android.view.animation.TranslateAnimation;

import android.widget.TextView;


public class Thread2 extends Thread{

    TextView t;
    int dir = 1;

    int translationDistance = 300;

    boolean paused=false;

    Object lock = new Object();

    Thread2(TextView t){

        this.t=t;

    }

    public void pause(boolean paused){

        synchronized (lock){

            if(paused)

                this.paused = true;

            else{

                this.paused = false;

                lock.notifyAll();

            }

        }

        Log.d("Debug",""+paused);

    }
```

```
    public void run(){
        while (!paused) {

            try {

                TranslateAnimation animation;

                if (dir == 1) {

                    animation = new TranslateAnimation(-translationDistance,
translationDistance, 0, 0);

                } else {

                    animation = new TranslateAnimation(translationDistance,
-translationDistance, 0, 0);

                }


                animation.setDuration(3000); // Keep the total duration the same

                animation.setFillAfter(true);

                t.startAnimation(animation);


                Thread.sleep(3000);

                dir = 1 - dir;


                synchronized (lock){

                    while(paused){

                        try{

                            lock.wait();
                        }catch(InterruptedException e){
```

```
                }
            }
        }
    } catch (InterruptedException e) {
        e.printStackTrace();

    }

  }
}
```

Thread3.java:

```
package com.example.ex5;

import android.util.Log;
import android.widget.TextView;
public class Thread3 extends Thread{
    TextView t;
    int ctr=0;
    boolean paused = false;
```

```
Object lock = new Object();
Thread3(TextView t){
    this.t=t;
}
public void pause(boolean paused){
    synchronized (lock){
        if(paused)
            this.paused = true;
        else{
            this.paused = false;
            lock.notifyAll();
        }
    }
    Log.d("Debug",""+paused);
}


public void run(){
    while (ctr < 3000 && !paused) {
        try {
            Thread.sleep(1000);
            ctr += 1;


            // Update the TextView on the UI thread
```

```
                t.post(new Runnable() {
                    @Override
                    public void run() {
                        t.setText(Integer.toString(ctr));
                    }
                });

                synchronized (lock){
                    while(paused){
                        try{
                            lock.wait();
                        }catch(InterruptedException e){

                        }
                    }
                }
            } catch (InterruptedException e) {
                    paused = true;
                    e.printStackTrace();
            }
        }
    }
}
```

Activity_main.xml:

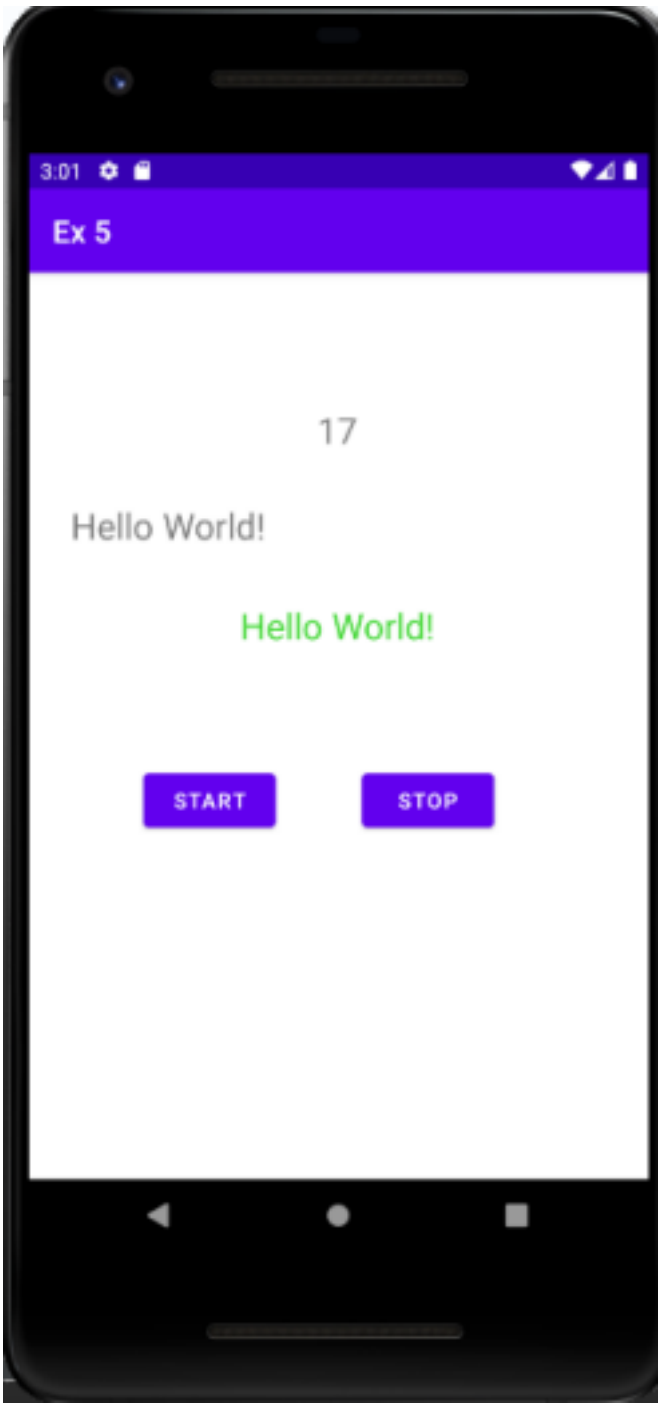
```
<?xml version="1.0" encoding="utf-8"?>

<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"

    tools:layout_editor_absoluteX="-1dp"

    tools:layout_editor_absoluteY="-83dp">


    <TextView
        android:id="@+id/t1"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:text="Hello World!"

        android:textSize="24sp"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintHorizontal_bias="0.498"

        app:layout_constraintLeft_toLeftOf="parent"

        app:layout_constraintRight_toRightOf="parent"
```

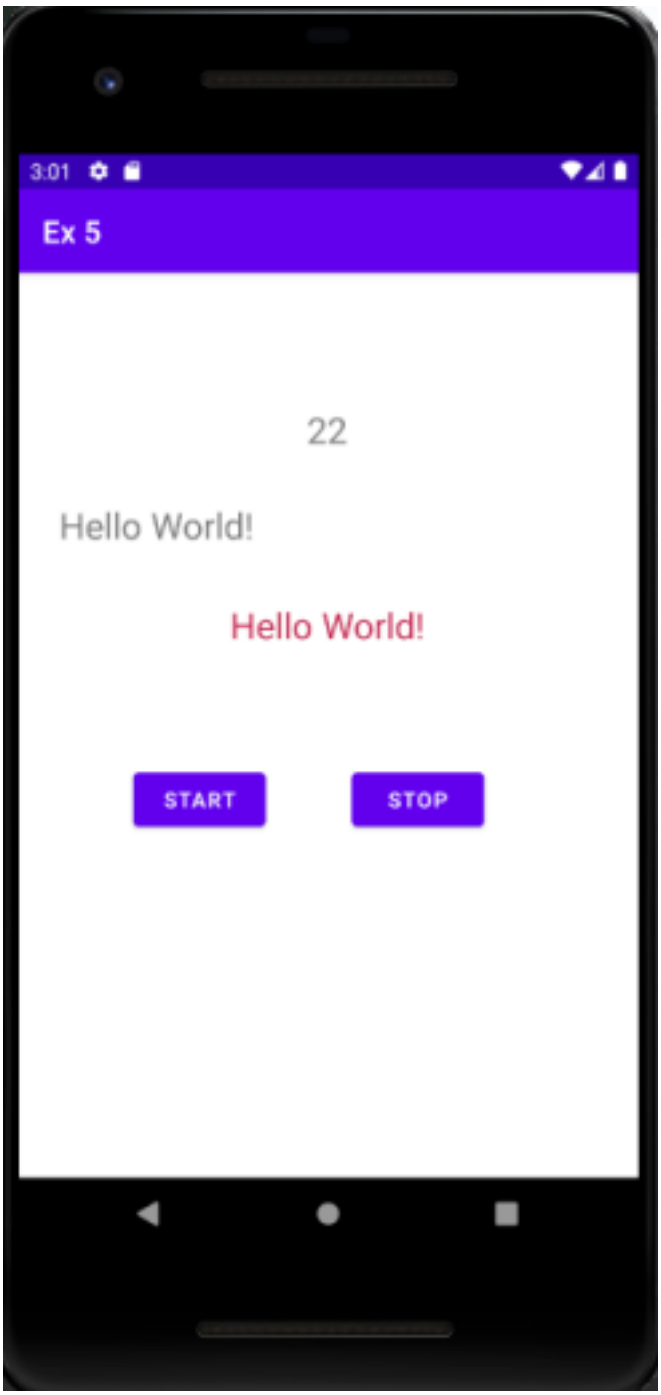
```
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.383" />

<TextView
    android:id="@+id/t2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hello World!"
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintHorizontal_bias="0.501"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.266" />

<TextView
    android:id="@+id/t3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="0"
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
```

```
    app:layout_constraintHorizontal_bias="0.498"

    app:layout_constraintLeft_toLeftOf="parent"

    app:layout_constraintRight_toRightOf="parent"

    app:layout_constraintTop_toTopOf="parent"

    app:layout_constraintVertical_bias="0.155" />


<Button

    android:id="@+id/stop"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"
    android:layout_marginTop="76dp"

    android:text="Stop"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintHorizontal_bias="0.684"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toBottomOf="@+id/t1" />


<Button

    android:id="@+id/start"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_marginTop="76dp"

    android:text="Start"
```

```
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.236"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/t1" />



</androidx.constraintlayout.widget.ConstraintLayout>
```

Output:


3:01
Ex 5
17
Hello World!
Hello World!
START
STOP


3:01
Ex 5
22
Hello World!
Hello World!
START
STOP

Best Practices:

● Names for ids of buttons were set meaningfully ● Implemented pause and resume in a single function
● Handled exceptions

Learning Outcomes:

● Learnt to implement multithreading ● Learnt to start, stop and resume threads