

Dr. B R Ambedkar National Institute of Technology
Grand Trunk Road, Barnala Amritsar Bypass Road,
Jalandhar, Punjab 144011



MOBILE APPLICATION DEVELOPMENT LAB
B.TECH IT III YEAR (6th SEMESTER) ITPC 326

Submitted To

Ms. Mridul Sharma

Department of Information Technology

Submitted By

Saloni

21124095

INDEX

[illegible]

[illegible]

LAB-1

AIM: Installation of Android Studio.

THEORY:

- Step 1 - System Requirements

The required tools to develop Android applications are open source and can be downloaded from the Web. Following is the list of software's you will need before you start your Android application programming.

Java JDK5 or later version

Java Runtime Environment (JRE) 6

Android Studio

- Step 2 - Setup Android Studio

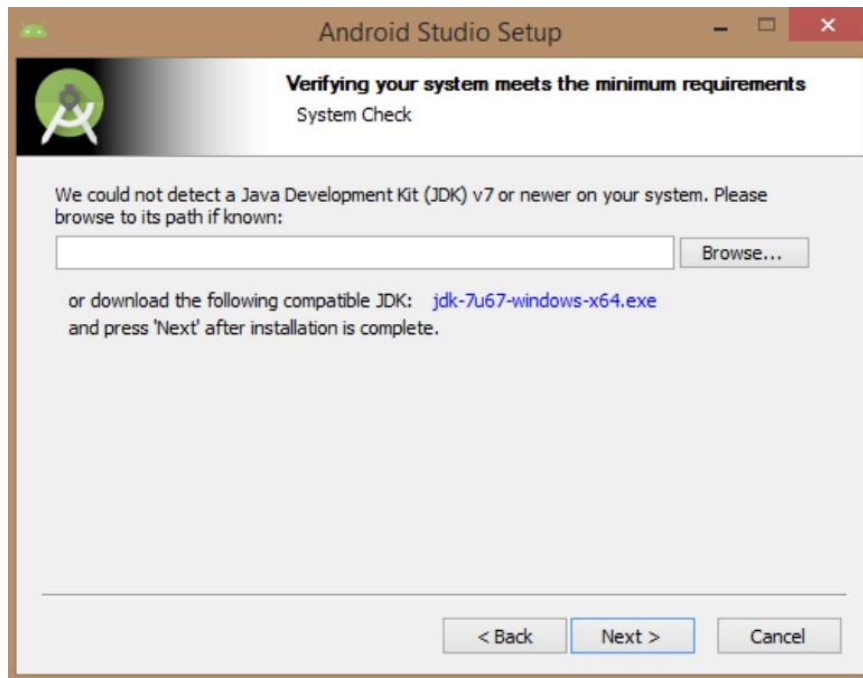
Android Studio is the official IDE for android application development. It works based on IntelliJ IDEA. You can download the latest version of android studio from Android Studio 2.2

Download, If you are new to installing Android Studio on windows, you will find a file, which is named as android-studio-bundle-143.3101438-windows.exe. So just download and run on windows machine according to android studio wizard guideline.

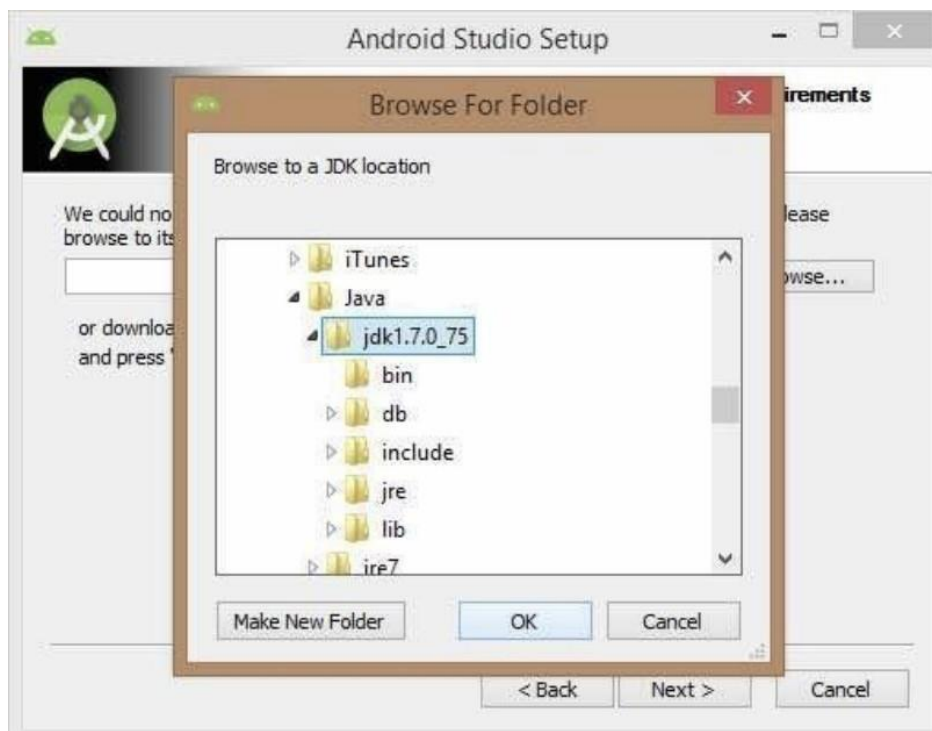
Installation

So let's launch Android Studio.exe. Make sure before launching Android Studio, Our Machine should require installed Java JDK. To install Java JDK, take references of Android environment setup.

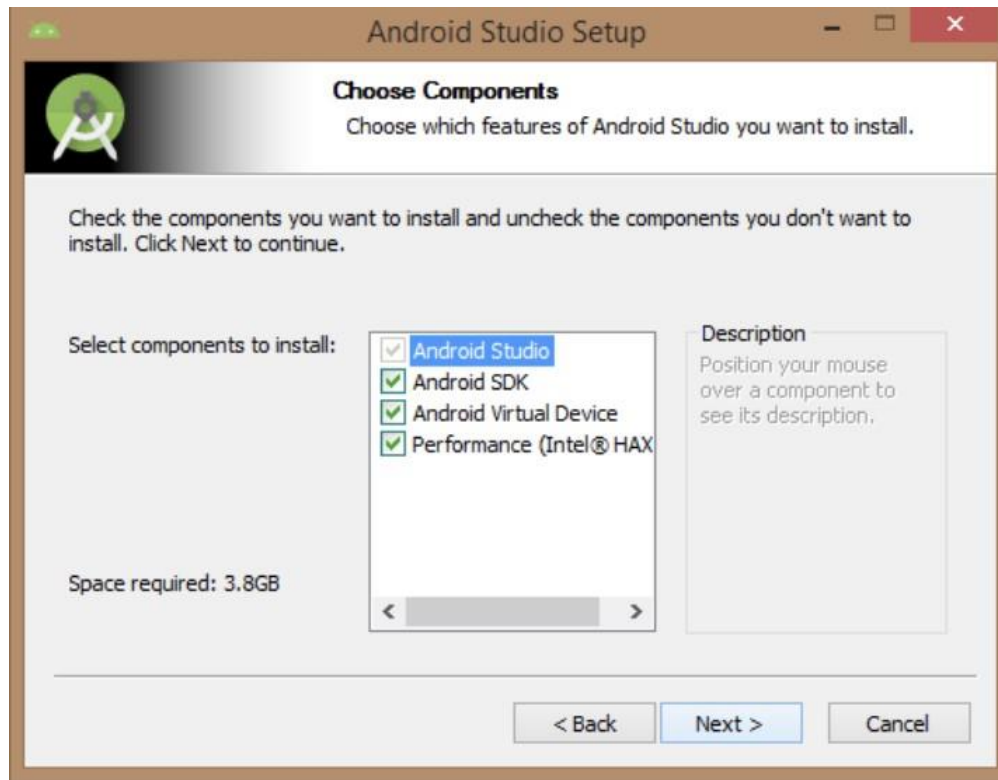




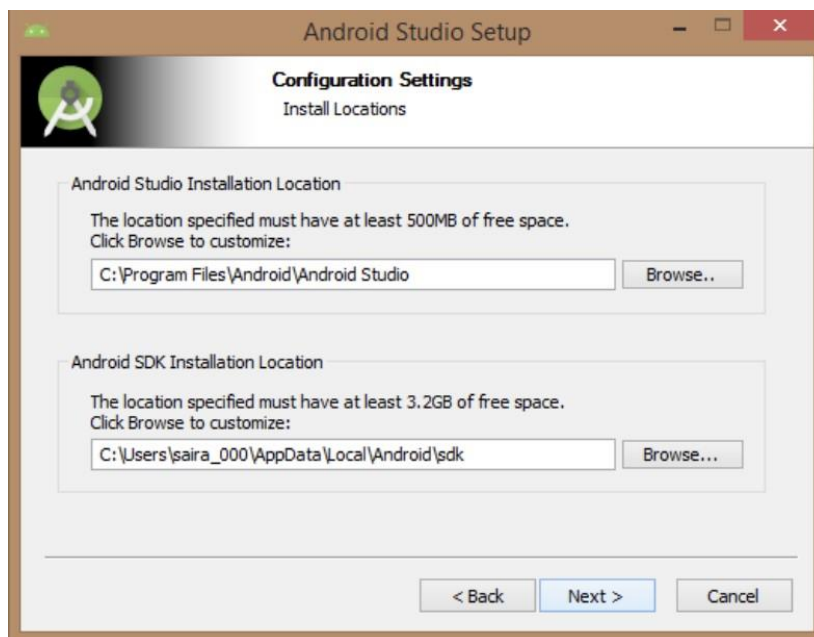
Below is the image initiating JDK to android SDK



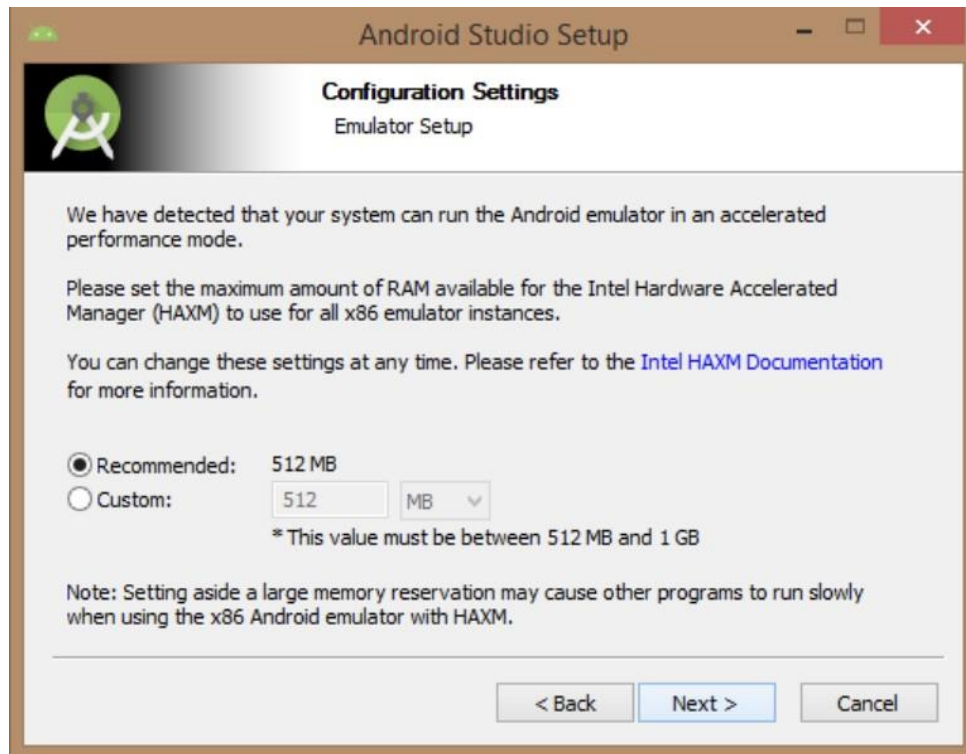
Need to check the components, which are required to create applications, below the image has selected Android Studio, Android SDK, Android Virtual Machine and performance(Intel chip).



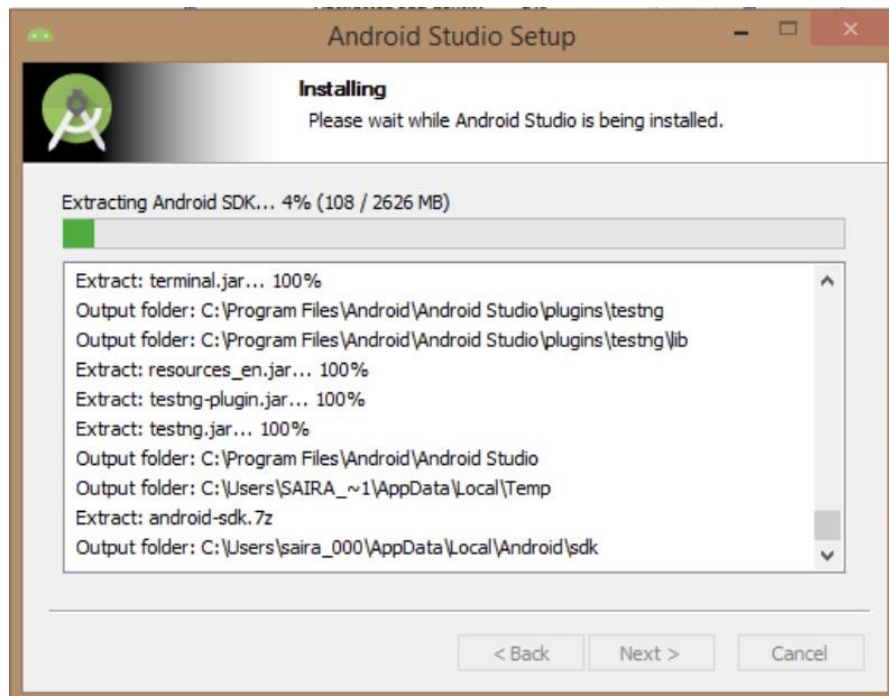
Need to specify the location of local machine path for Android studio and Android SDK, below the image has taken default location of windows 8.1 x64 bit architecture.



Need to specify the ram space for Android emulator by default it would take 512MB of local machine RAM.



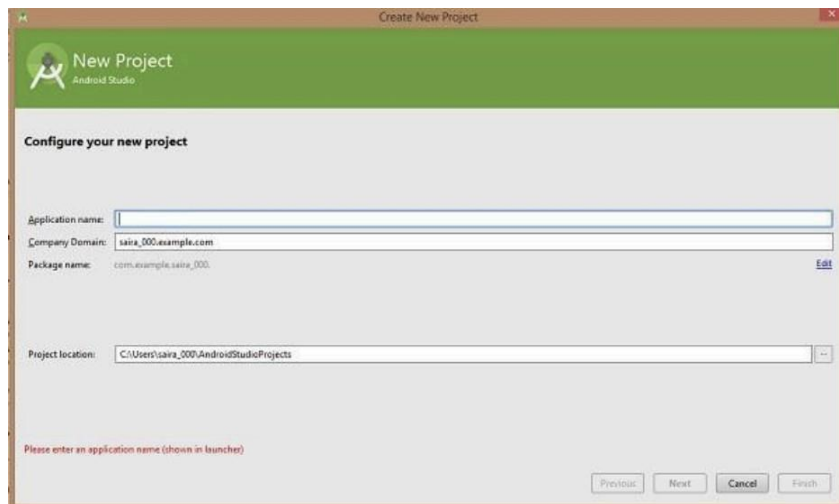
At final stage, it would extract SDK packages into our local machine, it would take a while time to finish the task and would take 2626MB of Hard disk space.

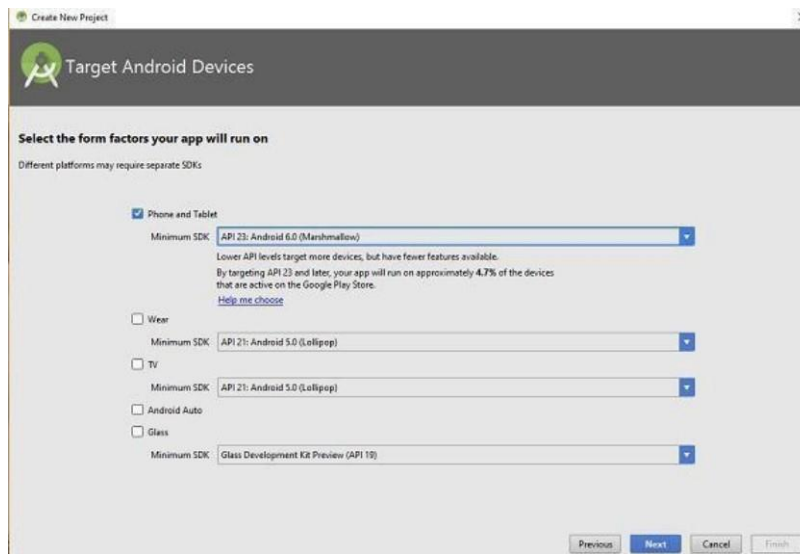


After done all above steps perfectly, you must get finish button and it gonna be open android studio project with Welcome to android studio message as shown below



You can start your application development by calling start a new android studio project. in a new installation frame should ask Application name, package information and location of the project.

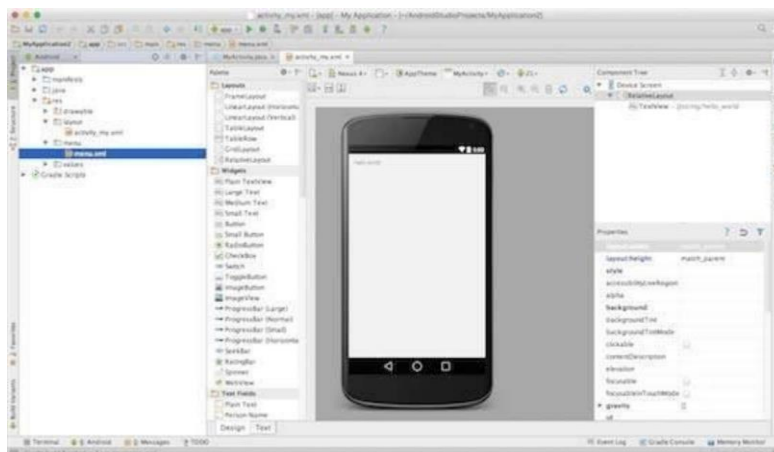




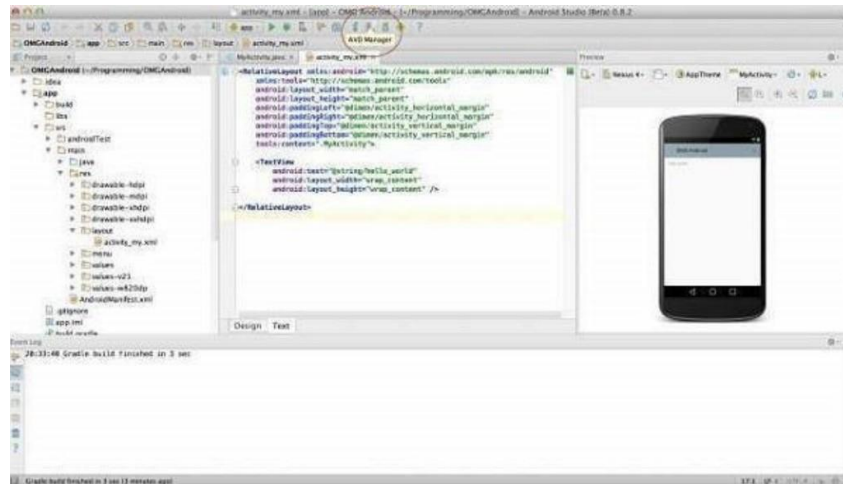
The next level of installation should contain selecting the activity to mobile, it specifies the default layout for Applications



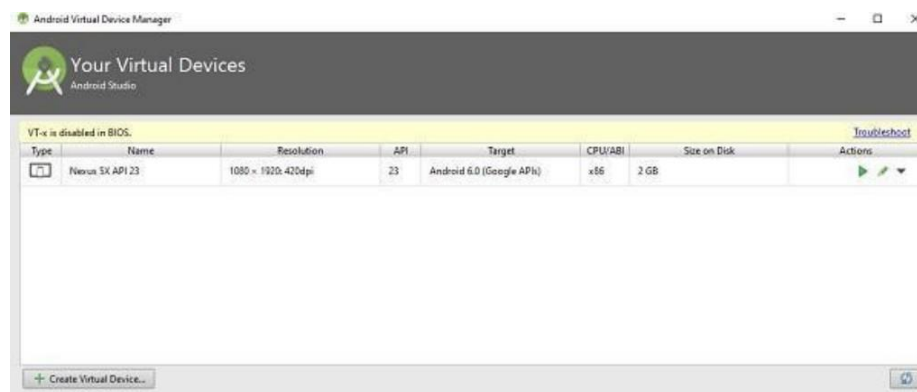
At the final stage it is going to be an open development tool to write the application code.



- Step 3 - Create Android Virtual Device To test your Android applications, you will need a virtual Android device. So before we start writing our code, let us create an Android virtual device. Launch Android AVD Manager Clicking AVD_Manager icon as shown below



After Click on a virtual device icon, it going to be shown by default virtual devices which are present on your SDK, or else need to create a virtual device by clicking Create new Virtual device button



If your AVD is created successfully it means your environment is ready for Android application development. If you like, you can close this window using the top-right cross button. Better you restart your machine and once you are done with this last step, you are ready to proceed for your first Android example but before that we will see a few more important concepts related to Android Application Development.

RESULT:

We have successfully installed Android Studio.

LAB-2

AIM- To develop an android application that uses GUI Components, Font and colors.

ALGORITHM

1. Create a New Android Project:
 - Click New in the toolbar.
 - In the window that appears, open the Android folder, select Android Application Project, and click next.
 - Provide the application name and the project name and then finally give the desired package name.
 - Choose a launcher icon for your application and then select Blank Activity and then click Next
 - Provide the desired Activity name for your project and then click Finish.
2. Create a New AVD (Android Virtual Device):
 - click Android Virtual Device Manager from the toolbar.
 - In the Android Virtual Device Manager panel, click New.
 - Fill in the details for the AVD. Give it a name, a platform target, an SD card size, and a skin (HVGA is default).
 - Click Create AVD and Select the new AVD from the Android Virtual Device Manager and click Start.
3. Design the graphical layout with a text view and two command buttons.
4. Run the application.
5. On pressing the change color button, color of the text gets changed.
6. On pressing the change font size button, the size of the font gets altered.
7. Close the Android project.

CODE-

Activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
```

```
android:layout_height="wrap_content"
android:layout_margin="30dp"
android:gravity="center"
android:text="Welcome"
android:textColor="#FF0000"
android:textSize="25sp"
android:textStyle="bold" />
```

```
<Button
    android:id="@+id/button1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:gravity="center"
    android:text="Font size"
    android:textSize="25sp" />
```

```
<Button
    android:id="@+id/button2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:gravity="center"
    android:text="Color"
    android:textSize="25sp" />
```

```
</LinearLayout>
```

ActivityMain.java:

```
package com.example.guiColor;

import android.os.Bundle;

import com.google.android.material.snackbar.Snackbar;

import androidx.appcompat.app.AppCompatActivity;

import android.view.View;
```

```

import androidx.core.view.WindowCompat;
import androidx.navigation.NavController;
import androidx.navigation.Navigation;
import androidx.navigation.ui.AppBarConfiguration;
import androidx.navigation.ui.NavigationUI;

import com.example.fontcolor.databinding.ActivityMainBinding;

import android.view.Menu;
import android.view.MenuItem;

import android.graphics.Color;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity
{
    int ch=1;
    float font=30;
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        final TextView t= (TextView) findViewById(R.id.textView);
        Button b1= (Button) findViewById(R.id.button1);

        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                t.setTextSize(font);
                font = font + 5;
                if (font == 50)
                {
                    font = 30;
                }
            }
        });
        Button b2= (Button) findViewById(R.id.button2);
        b2.setOnClickListener(new View.OnClickListener() {
            @Override

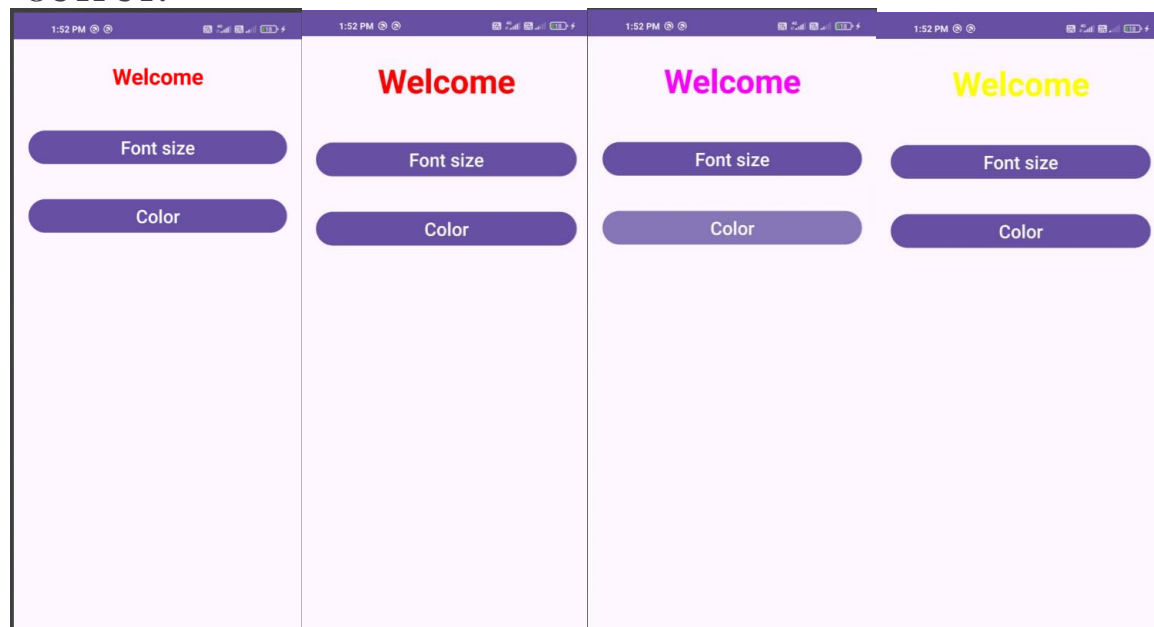
```

```

public void onClick(View v) {
    switch (ch) {
        case 1:
            t.setTextColor(Color.RED);
            break;
        case 2:
            t.setTextColor(Color.GREEN);
            break;
        case 3:
            t.setTextColor(Color.BLUE);
            break;
        case 4:
            t.setTextColor(Color.CYAN);
            break;
        case 5:
            t.setTextColor(Color.YELLOW);
            break;
        case 6:
            t.setTextColor(Color.MAGENTA);
            break;
    }
    ch++;
    if (ch == 7)
        ch = 1;
    }
});
}

```

OUTPUT:



Lab-3

AIM:

Develop an application that uses layout managers and event listeners. You have to design “STUDENT DETAIL FORM” with User Registration Number, Name and Department (can use dropdown menu for choosing department name) as input from users and a Submit button which redirects to another screen displaying all details.

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">

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="148dp"
        android:layout_marginTop="360dp"
        android:text="Submit"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/editTextText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="88dp"
        android:layout_marginTop="104dp"
        android:ems="10"
        android:inputType="text"
        android:text="Name"
```

```
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent" />
```

```
<EditText
    android:id="@+id/editTextText2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="88dp"
    android:layout_marginTop="184dp"
    android:ems="10"
    android:inputType="text"
    android:text="Registration Number"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

```
<Spinner
    android:id="@+id/spinner"
    android:layout_width="213dp"
    android:layout_height="41dp"
    android:layout_marginStart="88dp"
    android:layout_marginTop="260dp"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

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

activity_details.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <TextView
        android:id="@+id/textViewName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="18sp"
        android:textStyle="bold" />
```



```
<TextView
    android:id="@+id/textViewRegNumber"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:textSize="18sp"
    android:textStyle="bold" />
```

```
<TextView
    android:id="@+id/textViewBranch"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:textSize="18sp"
    android:textStyle="bold" />
```

```
</LinearLayout>
```

MainActivity.java

```
package com.formStudent;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.ArrayAdapter;

public class MainActivity extends AppCompatActivity {

    EditText editTextName, editTextRegistrationNumber;
    Spinner spinner;
    Button button;
```

```

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

    // Initialize views
    editTextName = findViewById(R.id.editTextText);
    editTextRegistrationNumber = findViewById(R.id.editTextText2);
    spinner = findViewById(R.id.spinner);
    button = findViewById(R.id.button);

    // Set up the Spinner
    ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this,
        R.array.spinner_items, android.R.layout.simple_spinner_item);
    adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    spinner.setAdapter(adapter);

    // Set up button click listener
    button.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            // Get values
            String name = editTextName.getText().toString().trim();
            String regNumber = editTextRegistrationNumber.getText().toString().trim();
            String selectedItem = spinner.getSelectedItem().toString();

            // Do something with the values
            // For example, log them
            System.out.println("Name: " + name);
            System.out.println("Registration Number: " + regNumber);
            System.out.println("Selected Item: " + selectedItem);
            // Create an Intent to start DetailsActivity
            Intent intent = new Intent(MainActivity.this, DetailsActivity.class);
            // Pass the entered details to DetailsActivity
            intent.putExtra("name", name);
            intent.putExtra("regNumber", regNumber);
            intent.putExtra("selectedItem", selectedItem);
            startActivity(intent);
        }
    });
}

```

```
}  
}
```

DetailsActivity.java

```
package com.formStudent;  
  
import android.os.Bundle;  
import android.widget.TextView;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
public class DetailsActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_details);  
  
        // Get the intent that started this activity  
        Bundle extras = getIntent().getExtras();  
        if (extras != null) {  
            // Retrieve the entered details from the intent  
            String name = extras.getString("name");  
            String regNumber = extras.getString("regNumber");  
            String branch = extras.getString("selectedItem");  
  
            // Display the details in TextViews  
            TextView textViewName = findViewById(R.id.textViewName);  
            TextView textViewRegNumber = findViewById(R.id.textViewRegNumber);  
            TextView textViewBranch = findViewById(R.id.textViewBranch);  
  
            textViewName.setText("Name: " + name);  
            textViewRegNumber.setText("Registration Number: " + regNumber);  
            textViewBranch.setText("Branch: " + branch);  
        }  
    }  
}
```

Output:

A screenshot of a mobile application interface. At the top, a purple status bar shows the time as 2:20 PM and various system icons. The app's background is a light pink color. The form consists of three input fields: a text field containing 'Saloni Aggarwal', a text field containing '21124095', and a dropdown menu currently showing 'IT'. Below these fields is a purple 'Submit' button. Directly under the button, the text 'Hello World!' is displayed.

2:20 PM

Saloni Aggarwal

21124095

IT

Submit

Hello World!