

Practical 0

Aim: Introduction to Android, Introduction to Android Studio IDE, Application Fundamentals: Creating a Project, Android Components, Activities, Services, Content Providers, Broadcast Receivers, Interface overview, Creating Android Virtual device, USB debugging mode, Android Application Overview. Simple “Hello World” program.

Code:

Main Activity.java

```
package com.example.myapplication;
import android.os.Bundle;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });
    }
}
```

activity_main.xml

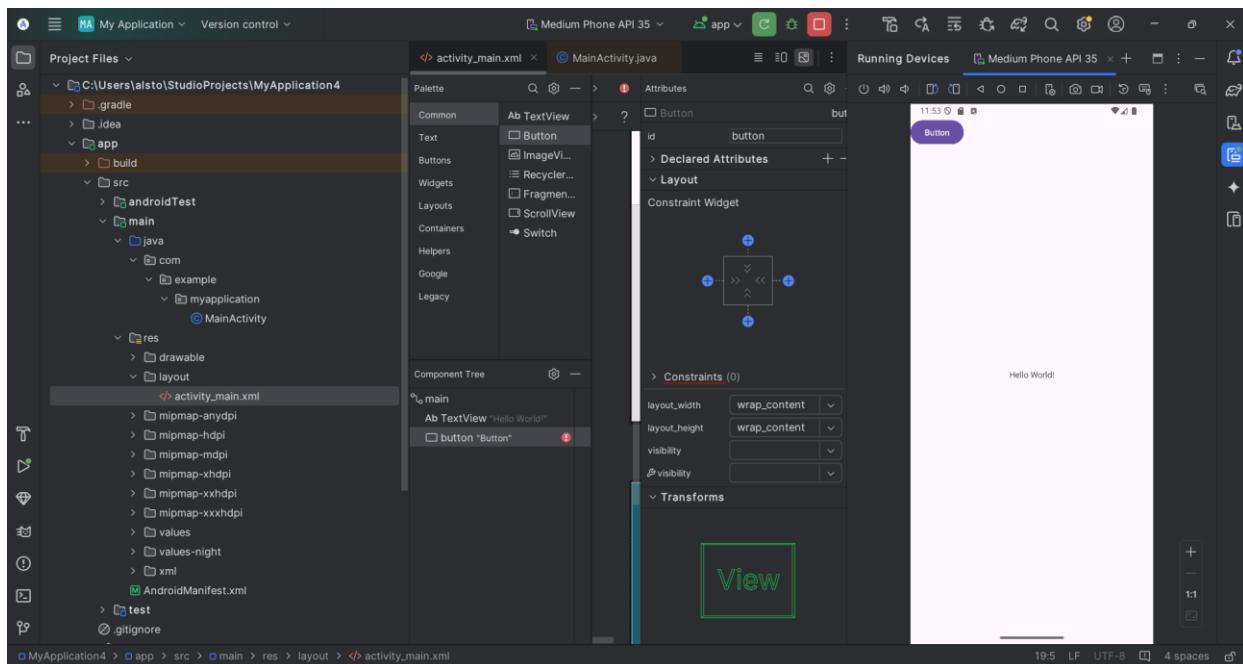
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:text="Button"
        tools:layout_editor_absoluteX="162dp"
        tools:layout_editor_absoluteY="386dp" />

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

Output:

Conclusion: Successfully performed this practical.

Practical 1

Aim: Programming Resources Android Resources: (Color, Theme, String, Drawable, Dimension, Image),

Code:

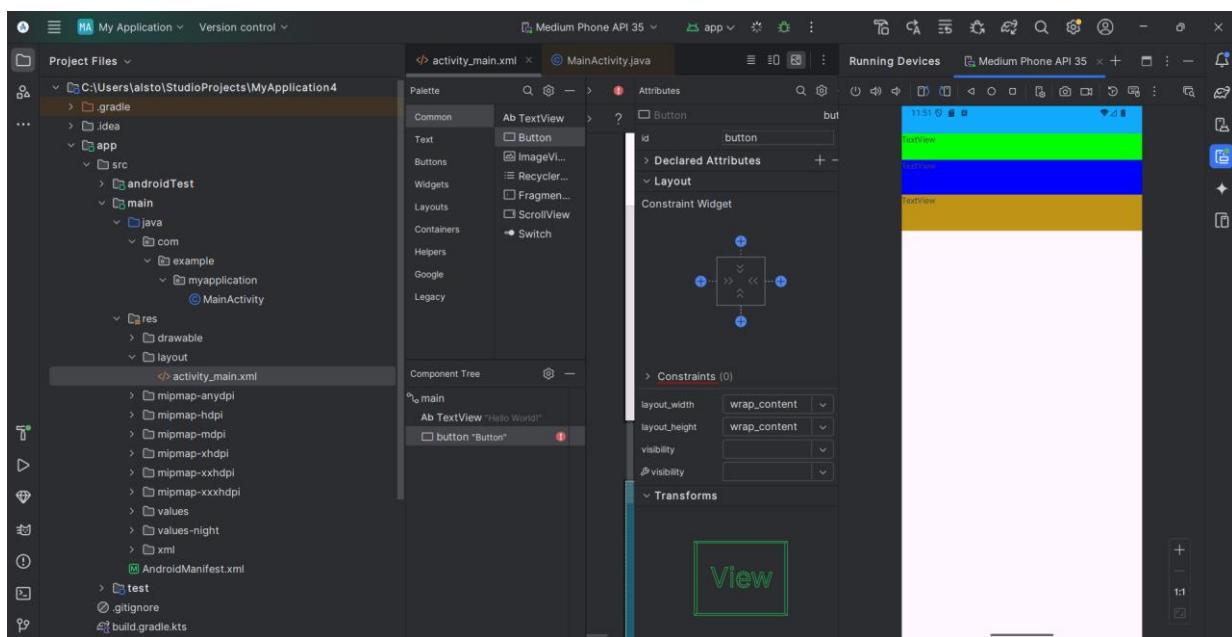
colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFF</color>
    <color name="r">#FF0000</color>
    <color name="g">#00FF00</color>
    <color name="b">#0000ff</color>
    <color name="c1">#11AAff</color>
    <color name="w">#ffffff</color>
    <color name="bl">#000000</color>
    <color name="c2">#BD9416</color>
</resources>
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
```

```
    android:orientation="vertical">  
    <TextView  
        android:id="@+id/textView"  
        android:layout_width="match_parent"  
        android:layout_height="47dp"  
        android:text="TextView"  
        android:textColor="@color/r"  
        android:background="@color/c1"  
    />  
  
    <TextView  
        android:id="@+id/textView2"  
        android:layout_width="match_parent"  
        android:layout_height="45dp"  
        android:text="TextView"  
        android:background="@color/g"  
    />  
  
    <TextView  
        android:id="@+id/textView3"  
        android:layout_width="match_parent"  
        android:layout_height="59dp"  
        android:text="TextView"  
        android:background="@color/b"  
    />  
  
    <TextView  
        android:id="@+id/textView4"  
        android:layout_width="match_parent"  
        android:layout_height="62dp"  
        android:text="TextView"  
        android:background="@color/c2"  
    />  
    </LinearLayout>  
  </RelativeLayout>
```

Output:

Conclusion: Successfully performed this practical.

Practical 2

Aim: Programming Activities and fragments Activity Life Cycle, Activity methods, Multiple Activities, Life Cycle of fragments and multiple fragments.

Code:

MainActivity.java

```
package com.example.practical;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.Fragment;

public class MainActivity extends AppCompatActivity {

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

        // Load FirstFragment initially
        if (savedInstanceState == null) {
            getSupportFragmentManager().beginTransaction()
                .replace(R.id.fragment_container, new FirstFragment())
                .commit();
        }
    }

    // First Fragment Class
```

```
public static class FirstFragment extends Fragment {  
    @Nullable  
    @Override  
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable  
    Bundle savedInstanceState) {  
        View view = inflater.inflate(R.layout.fragment_first, container, false);  
  
        Button button = view.findViewById(R.id.buttonFragment);  
        button.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View v) {  
                requireActivity().getSupportFragmentManager().beginTransaction()  
                    .replace(R.id.fragment_container, new SecondFragment())  
                    .addToBackStack(null)  
                    .commit();  
            }  
        });  
        return view;  
    }  
}  
  
// Second Fragment Class  
public static class SecondFragment extends Fragment {  
    @Nullable  
    @Override  
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable  
    Bundle savedInstanceState) {  
        return inflater.inflate(R.layout.fragment_second, container, false);  
    }  
}
```

activity_main.xml

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

```
    android:id="@+id/fragment_container"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"/>
```

fragment_first.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:orientation="vertical"  
    android:gravity="center">  
  
    <Button  
        android:id="@+id/buttonFragment"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:text="Go to Second Fragment"/>  
  
</LinearLayout>
```

Fragment_second.xml

```
<TextView xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:text="This is Second Fragment"  
    android:gravity="center"  
    android:textSize="20sp"/>
```

AndroidManifest.xml

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

Output:

The screenshots show the Android Studio interface with three tabs open:

- MainActivity.java:** Displays the Java code for the main activity, which includes a call to `startActivity(new Intent(this, SecondActivity.class))`.
- SecondActivity.java:** Displays the Java code for the second activity.
- fragment_main.xml:** Displays the XML layout for the fragment, which contains a button labeled "Hello from Fragment".

Conclusion: Successfully performed this practical.

Practical 3

Aim: Programs related to different Layouts Coordinate, Linear, Relative, Table, Absolute, Frame, List View, Grid View.

Code:

Main Activity.java

```
package com.example.myapplication;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.FrameLayout;
import android.widget.LinearLayout;
import android.widget.RelativeLayout;
import android.widget.TableLayout;
import android.widget.GridLayout;
import android.widget.ListView;
import android.widget.ArrayAdapter;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
import androidx.constraintlayout.widget.ConstraintLayout;

public class MainActivity extends AppCompatActivity {
    private FrameLayout container;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        container = findViewById(R.id.layout_container);
        setupButton(R.id.btn_constraint, createConstraintLayout());
        setupButton(R.id.btn_linear, createLinearLayout());
        setupButton(R.id.btn_relative, createRelativeLayout());
        setupButton(R.id.btn_table, createTableLayout());
        setupButton(R.id.btn_frame, createFrameLayout());
        setupButton(R.id.btn_list, createListView());
        setupButton(R.id.btn_grid, createGridLayout());
    }
    private void setupButton(int buttonId, final View layout) {
        Button button = findViewById(buttonId);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                container.removeAllViews();
                container.addView(layout);
            }
        });
    }
    private View createConstraintLayout() {
        ConstraintLayout layout = new ConstraintLayout(this);
        TextView textView = new TextView(this);
    }
}
```

```
textView.setText("Constraint Layout Example");
textView.setTextSize(20);
layout.addView(textView);
return layout;
}

private View createLinearLayout() {
    LinearLayout layout = new LinearLayout(this);
    layout.setOrientation(LinearLayout.VERTICAL);
    TextView textView = new TextView(this);
    textView.setText("Linear Layout Example");
    layout.addView(textView);
    return layout;
}

private View createRelativeLayout() {
    RelativeLayout layout = new RelativeLayout(this);
    TextView textView = new TextView(this);
    textView.setText("Relative Layout Example");
    layout.addView(textView);
    return layout;
}

private View createTableLayout() {
    TableLayout layout = new TableLayout(this);
    TextView textView = new TextView(this);
    textView.setText("Table Layout Example");
    layout.addView(textView);
    return layout;
}

private View createFrameLayout() {
    FrameLayout layout = new FrameLayout(this);
    TextView textView = new TextView(this);
    textView.setText("Frame Layout Example");
    layout.addView(textView);
    return layout;
}

private View createListView() {
    ListView listView = new ListView(this);
    String[] items = {"Item 1", "Item 2", "Item 3"};
    ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1, items);
    listView.setAdapter(adapter);
    return listView;
}

private View createGridLayout() {
    GridLayout layout = new GridLayout(this);
    layout.setColumnCount(2);
    for (int i = 1; i <= 4; i++) {
```

```
    TextView textView = new TextView(this);
    textView.setText("Grid " + i);
    layout.addView(textView);
}
return layout;
}
```

activity_main.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">

    <Button android:id="@+id	btn_constraint" android:text="Constraint Layout"
        android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <Button android:id="@+id	btn_linear" android:text="Linear Layout"
        android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <Button android:id="@+id	btn_relative" android:text="Relative Layout"
        android:layout_width="match_parent" android:layout_height="wrap_content"/>

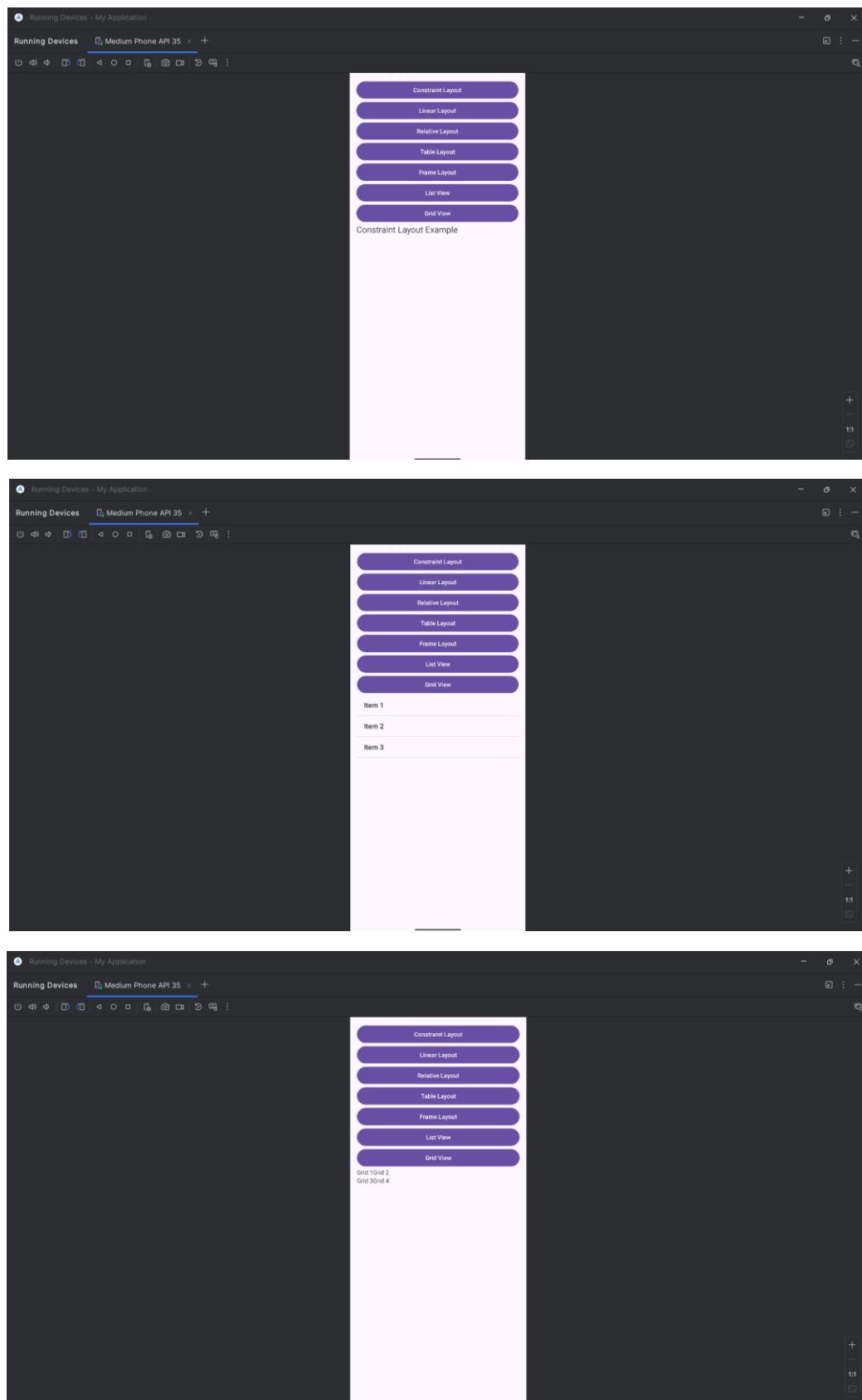
    <Button android:id="@+id	btn_table" android:text="Table Layout"
        android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <Button android:id="@+id	btn_frame" android:text="Frame Layout"
        android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <Button android:id="@+id	btn_list" android:text="List View"
        android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <Button android:id="@+id	btn_grid" android:text="Grid View"
        android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <FrameLayout android:id="@+id/layout_container"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</LinearLayout>
```

Output:

Conclusion: Successfully performed this practical.

Practical 4

Aim: Programming UI elements AppBar, Fragments, UI Components.

Code:

Main Activity.java

```
package com.example.myapplication;

import android.os.Bundle;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.FrameLayout;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import androidx.cardview.widget.CardView;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import java.util.ArrayList;
import java.util.List;

public class MainActivity extends AppCompatActivity {

    private FrameLayout container;

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

        // Setup AppBar (Toolbar)
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        getSupportActionBar().setTitle("UI Elements Demo");

        container = findViewById(R.id.layout_container);
        setupButton(R.id.btn_appbar, createAppBar());
        setupButton(R.id.btn_cardview, createCardView());
        setupButton(R.id.btn_recyclerview, createRecyclerView());
    }
}
```

```
private void setupButton(int buttonId, final View layout) {  
    Button button = findViewById(buttonId);  
    button.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            container.removeAllViews();  
            container.addView(layout);  
        } });}  
  
private View createAppBar() {  
    TextView textView = new TextView(this);  
    textView.setText("This is an AppBar Example");  
    textView.setTextSize(20);  
    return textView;}  
  
private View createCardView() {  
    CardView cardView = new CardView(this);  
    cardView.setCardElevation(8);  
    cardView.setRadius(12);  
    TextView textView = new TextView(this);  
    textView.setText("This is a CardView");  
    cardView.addView(textView);  
    return cardView;}  
  
private View createRecyclerView() {  
    RecyclerView recyclerView = new RecyclerView(this);  
    recyclerView.setLayoutManager(new LinearLayoutManager(this));  
    List<String> items = new ArrayList<>();  
    items.add("Item 1");  
    items.add("Item 2");  
    items.add("Item 3");  
    // Define and set adapter inline  
    RecyclerView.Adapter adapter = new RecyclerView.Adapter<RecyclerView.ViewHolder>() {  
        @Override  
        public RecyclerView.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {  
            TextView textView = new TextView(parent.getContext());  
            return new MyViewHolder(textView);  
        }  
        @Override  
        public void onBindViewHolder(RecyclerView.ViewHolder holder, int position) {  
            ((MyViewHolder) holder).textView.setText(items.get(position));  
        }  
        @Override  
        public int getItemCount() {  
            return items.size();  
        }  
    };  
    recyclerView.setAdapter(adapter);  
    return recyclerView;}
```

```
    textView.setPadding(20, 20, 20, 20);

    return new RecyclerView.ViewHolder(textView) {}; }

@Override

public void onBindViewHolder(RecyclerView.ViewHolder holder, int position) {

    ((TextView) holder.itemView).setText(items.get(position)); }

@Override

public int getItemCount() {

    return items.size(); } };

recyclerView.setAdapter(adapter);

return recyclerView; } }
```

activity_main.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">

    <androidx.appcompat.widget.Toolbar

        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="?attr/colorPrimary"
        android:title="UI Elements Demo"
        android:titleTextColor="@android:color/white"/>

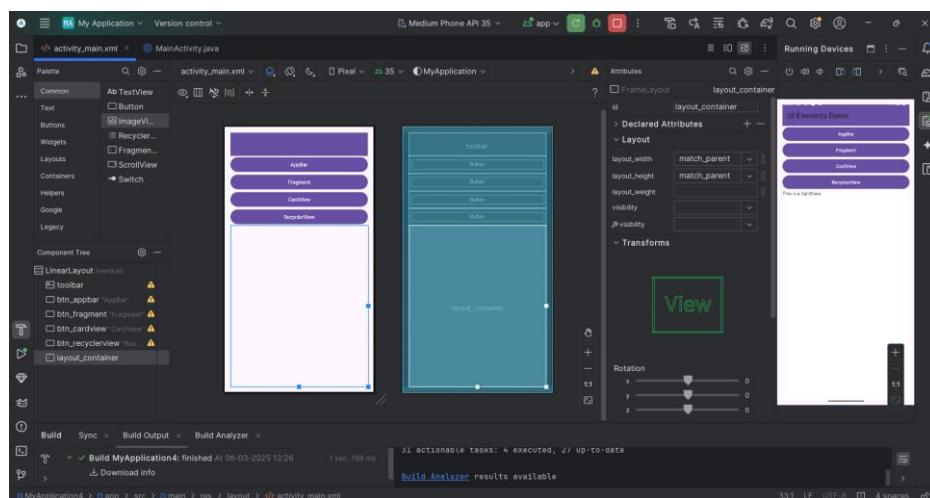
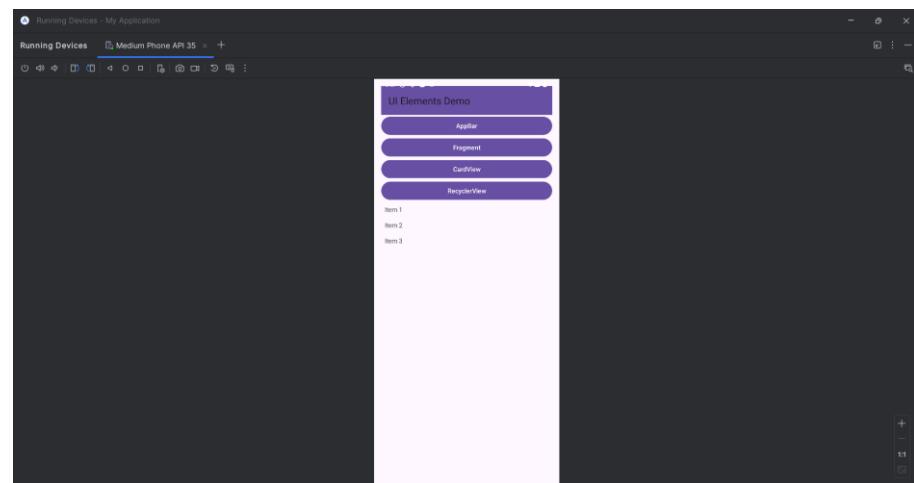
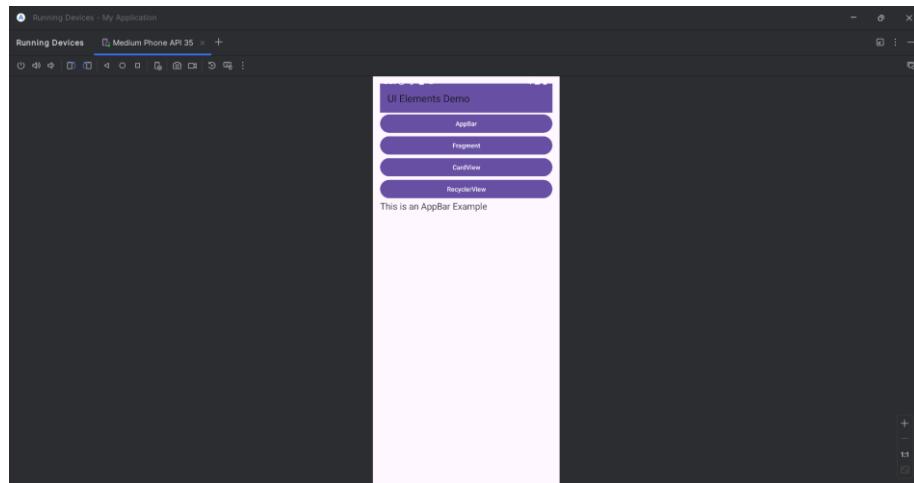
    <Button android:id="@+id/btn_appbar" android:text="AppBar"
        android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <Button android:id="@+id/btn_cardview" android:text="CardView"
        android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <Button android:id="@+id/btn_recyclerview" android:text="RecyclerView"
        android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <FrameLayout android:id="@+id/layout_container"
        android:layout_width="match_parent"
```

```
        android:layout_height="match_parent"/>  
</LinearLayout>
```

Output:

Conclusion: Successfully performed this practical.

Practical 5**Aim:** Programming menus, dialog, dialog fragments**Code:**

Main Activity.java

```
package com.example.myapplication;
import android.app.AlertDialog;
import android.app.Dialog;
import android.content.DialogInterface;
import android.os.Bundle;
import android.view.ContextMenu;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.ListView;
import android.widget.PopupMenu;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.DialogFragment;

public class MainActivity extends AppCompatActivity {
    private TextView textView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button btnShowDialog = findViewById(R.id.btn_show_dialog);
        Button btnShowFragment = findViewById(R.id.btn_show_fragment);
        Button btnShowPopup = findViewById(R.id.btn_show_popup);
        textView = findViewById(R.id.text_view);
        registerForContextMenu(textView); // Enable Context Menu
        btnShowDialog.setOnClickListener(v -> showAlertDialog());
        btnShowFragment.setOnClickListener(v -> new
CustomDialogFragment().show(getSupportFragmentManager(), "dialog"));
        btnShowPopup.setOnClickListener(this::showPopupMenu); }
    // 🔴 Create Options Menu
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.main_menu, menu);
        return true; }
    // 🔴 Handle Menu Clicks
    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        int id = item.getItemId(); // Store in a variable
```

```
if (id == R.id.menu_settings) {
    Toast.makeText(this, "Settings Clicked", Toast.LENGTH_SHORT).show();
    return true;
} else if (id == R.id.menu_about) {
    Toast.makeText(this, "About Clicked", Toast.LENGTH_SHORT).show();
    return true;
}
return super.onOptionsItemSelected(item); }

// 🔴 Show AlertDialog
private void showAlertDialog() {
    new AlertDialog.Builder(this)
        .setTitle("Confirmation")
        .setMessage("Are you sure you want to proceed?")
        .setPositiveButton("Yes", (dialog, which) -> Toast.makeText(MainActivity.this, "Confirmed!", Toast.LENGTH_SHORT).show())
        .setNegativeButton("No", (dialog, which) -> dialog.dismiss())
        .create().show(); }

// 🔴 Context Menu for TextView
@Override
public void onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuItemInfo menuInfo) {
    super.onCreateContextMenu(menu, v, menuInfo);
    getMenuInflater().inflate(R.menu.context_menu, menu); }

@Override
public boolean onContextItemSelected(@NonNull MenuItem item) {
    if (item.getItemId() == R.id.menu_edit) {
        Toast.makeText(this, "Edit Selected", Toast.LENGTH_SHORT).show();
    } else if (item.getItemId() == R.id.menu_delete) {
        Toast.makeText(this, "Delete Selected", Toast.LENGTH_SHORT).show();
    }
    return super.onContextItemSelected(item); }

// 🔴 Show Popup Menu
private void showPopupMenu(View v) {
    PopupMenu popup = new PopupMenu(this, v);
    popup.getMenuInflater().inflate(R.menu.popup_menu, popup.getMenu());
    popup.setOnMenuItemClickListener(item -> {
        Toast.makeText(this, item.getTitle() + " Clicked", Toast.LENGTH_SHORT).show();
        return true;
    });
    popup.show(); }

// 🔴 Custom Dialog Fragment
public static class CustomDialogFragment extends DialogFragment {
    @NonNull
    @Override
    public Dialog onCreateDialog(@Nullable Bundle savedInstanceState) {
        return new AlertDialog.Builder(getActivity())
            .setTitle("Fragment Dialog")
            .setMessage("This is a Dialog Fragment!")
            .setPositiveButton("OK", (dialog, which) -> dialog.dismiss())
            .create(); }}}
```

activity_main.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"
    android:background="#FAFAFA">

    <Button
        android:id="@+id	btn_show_dialog"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Show Alert Dialog" />

    <Button
        android:id="@+id	btn_show_fragment"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Show Dialog Fragment" />

    <Button
        android:id="@+id	btn_show_popup"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Show Popup Menu" />

    <TextView
        android:id="@+id/text_view"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Long press for Context Menu"
        android:gravity="center"
        android:textSize="18sp"
        android:padding="20dp"
        android:background="#DDDDDD" />
</LinearLayout>
```

main_menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/menu_settings" android:title="Settings"/>
    <item android:id="@+id/menu_about" android:title="About"/>
</menu>
```

context_menu.xml

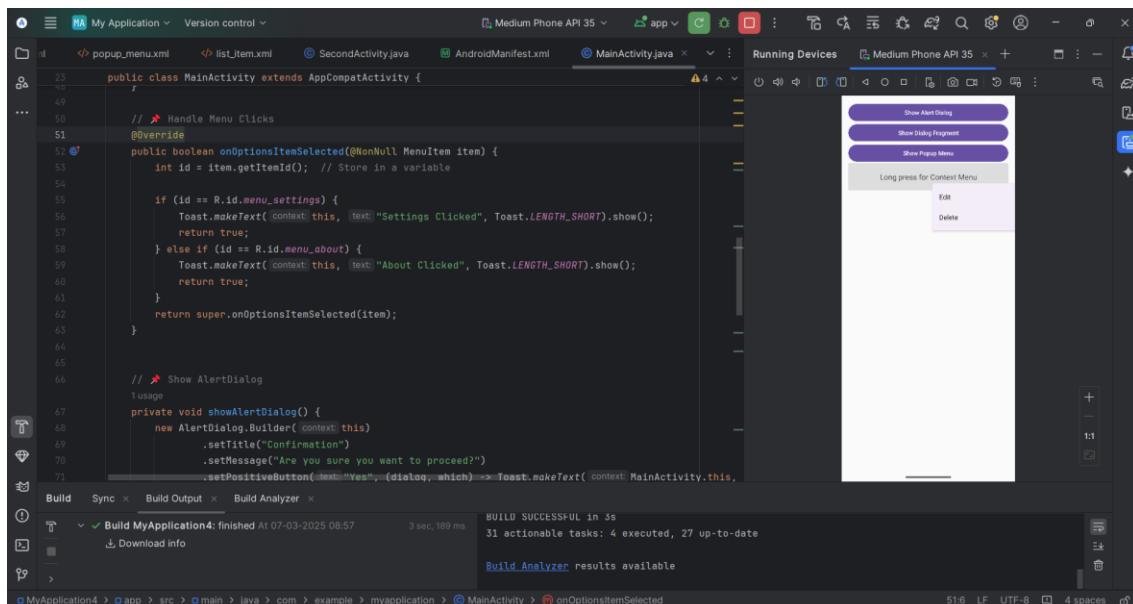
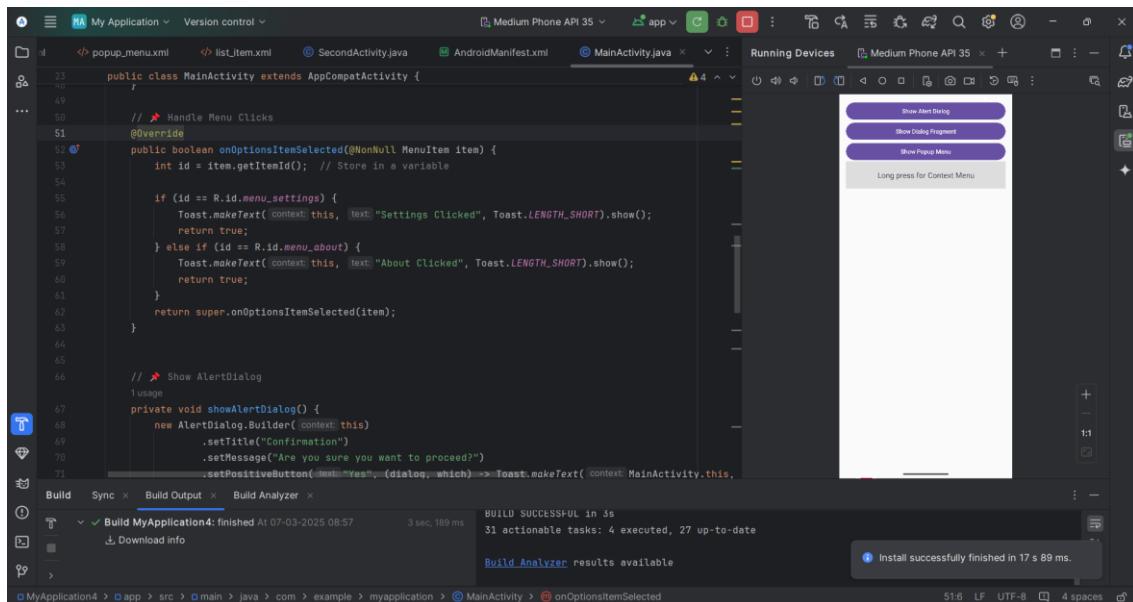
```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/menu_edit" android:title="Edit"/>
```

```
<item android:id="@+id/menu_delete" android:title="Delete"/>
</menu>
```

popup_menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/menu_option1" android:title="Option 1"/>
    <item android:id="@+id/menu_option2" android:title="Option 2"/>
</menu>
```

Output:





```

23    public class MainActivity extends AppCompatActivity {
...
50        // Handle Menu Clicks
51    @Override
52    public boolean onOptionsItemSelected(@NotNull MenuItem item) {
53        int id = item.getItemId(); // Store in a variable
54
55        if (id == R.id.menu_settings) {
56            Toast.makeText(context, "Settings Clicked", Toast.LENGTH_SHORT).show();
57            return true;
58        } else if (id == R.id.menu_about) {
59            Toast.makeText(context, "About Clicked", Toast.LENGTH_SHORT).show();
60            return true;
61        }
62        return super.onOptionsItemSelected(item);
63    }
64
65    // Show AlertDialog
66    private void showAlertDialog() {
67        new AlertDialog.Builder(this)
68            .setTitle("Confirmation")
69            .setMessage("Are you sure you want to proceed?")
70            .setPositiveButton("Yes", (dialog, which) -> Toast.makeText(context, MainActivity.this,
71

```

Build Sync Build Output Build Analyzer

Build MyApplication4: finished At 07-03-2025 08:57 3 sec, 189 ms BUILD SUCCESSFUL in 3s 31 actionable tasks: 4 executed, 27 up-to-date

Download info Build Analyzer results available



```

23    public class MainActivity extends AppCompatActivity {
...
50        // Handle Menu Clicks
51    @Override
52    public boolean onOptionsItemSelected(@NotNull MenuItem item) {
53        int id = item.getItemId(); // Store in a variable
54
55        if (id == R.id.menu_settings) {
56            Toast.makeText(context, "Settings Clicked", Toast.LENGTH_SHORT).show();
57            return true;
58        } else if (id == R.id.menu_about) {
59            Toast.makeText(context, "About Clicked", Toast.LENGTH_SHORT).show();
60            return true;
61        }
62        return super.onOptionsItemSelected(item);
63    }
64
65    // Show AlertDialog
66    private void showAlertDialog() {
67        new AlertDialog.Builder(this)
68            .setTitle("Confirmation")
69            .setMessage("Are you sure you want to proceed?")
70            .setPositiveButton("Yes", (dialog, which) -> Toast.makeText(context, MainActivity.this,
71

```

Build Sync Build Output Build Analyzer

Build MyApplication4: finished At 07-03-2025 08:57 3 sec, 189 ms BUILD SUCCESSFUL in 3s 31 actionable tasks: 4 executed, 27 up-to-date

Download info Build Analyzer results available



```

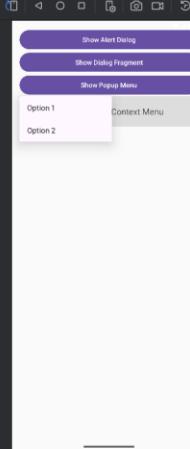
23    public class MainActivity extends AppCompatActivity {
...
50        // Handle Menu Clicks
51    @Override
52    public boolean onOptionsItemSelected(@NotNull MenuItem item) {
53        int id = item.getItemId(); // Store in a variable
54
55        if (id == R.id.menu_settings) {
56            Toast.makeText(context, "Settings Clicked", Toast.LENGTH_SHORT).show();
57            return true;
58        } else if (id == R.id.menu_about) {
59            Toast.makeText(context, "About Clicked", Toast.LENGTH_SHORT).show();
60            return true;
61        }
62        return super.onOptionsItemSelected(item);
63    }
64
65    // Show AlertDialog
66    private void showAlertDialog() {
67        new AlertDialog.Builder(this)
68            .setTitle("Confirmation")
69            .setMessage("Are you sure you want to proceed?")
70            .setPositiveButton("Yes", (dialog, which) -> Toast.makeText(context, MainActivity.this,
71

```

Build Sync Build Output Build Analyzer

Build MyApplication4: finished At 07-03-2025 08:57 3 sec, 189 ms BUILD SUCCESSFUL in 3s 31 actionable tasks: 4 executed, 27 up-to-date

Download info Build Analyzer results available



```

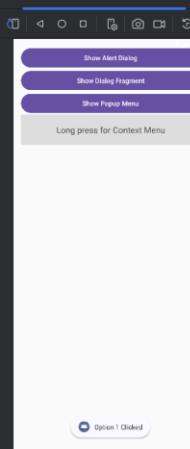
public class MainActivity extends AppCompatActivity {
    ...
    // Handle Menu Clicks
    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        int id = item.getItemId(); // Store in a variable

        if (id == R.id.menu_settings) {
            Toast.makeText(context, "Settings Clicked", Toast.LENGTH_SHORT).show();
            return true;
        } else if (id == R.id.menu_about) {
            Toast.makeText(context, "About Clicked", Toast.LENGTH_SHORT).show();
            return true;
        }
        return super.onOptionsItemSelected(item);
    }

    // Show AlertDialog
    private void showAlertDialog() {
        new AlertDialog.Builder(context)
            .setTitle("Confirmation")
            .setMessage("Are you sure you want to proceed?")
            .setPositiveButton("Yes", (dialog, which) -> Toast.makeText(context, MainActivity.this,
    
```

Build Sync Build Output Build Analyzer

Build MyApplication4: finished At 07-03-2025 08:57 3 sec, 189 ms BUILD SUCCESSFUL in 3s
31 actionable tasks: 4 executed, 27 up-to-date
Build Analyzer results available



```

public class MainActivity extends AppCompatActivity {
    ...
    // Handle Menu Clicks
    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        int id = item.getItemId(); // Store in a variable

        if (id == R.id.menu_settings) {
            Toast.makeText(context, "Settings Clicked", Toast.LENGTH_SHORT).show();
            return true;
        } else if (id == R.id.menu_about) {
            Toast.makeText(context, "About Clicked", Toast.LENGTH_SHORT).show();
            return true;
        }
        return super.onOptionsItemSelected(item);
    }

    // Show AlertDialog
    private void showAlertDialog() {
        new AlertDialog.Builder(context)
            .setTitle("Confirmation")
            .setMessage("Are you sure you want to proceed?")
            .setPositiveButton("Yes", (dialog, which) -> Toast.makeText(context, MainActivity.this,
    
```

Build Sync Build Output Build Analyzer

Build MyApplication4: finished At 07-03-2025 08:57 3 sec, 189 ms BUILD SUCCESSFUL in 3s
31 actionable tasks: 4 executed, 27 up-to-date
Build Analyzer results available

Conclusion: Successfully performed this practical.

Practical 6

Aim: Programs on Intents, Events, Listeners and Adapters. The Android Intent Class, Using Events and Event Listeners

Code:

Main Activity.java

```
package com.example.myapplication;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.LinearLayout;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import java.util.ArrayList;
import java.util.List;

public class MainActivity extends AppCompatActivity {

    private LinearLayout container;

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

        // Set up AppBar (Toolbar)
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        getSupportActionBar().setTitle("Intents, Events, Listeners");
        container = findViewById(R.id.layout_container);
    }
}
```

```
setupButton(R.id.btn_explicit_intent, createExplicitIntentButton());
setupButton(R.id.btn_implicit_intent, createImplicitIntentButton());
setupButton(R.id.btn_long_press, createLongPressButton());
setupButton(R.id.btn_listview, createListView()); }

private void setupButton(int buttonId, final View layout) {
    Button button = findViewById(buttonId);
    button.setOnClickListener(v -> {
        container.removeAllViews();
        container.addView(layout);
    });
}

private View createExplicitIntentButton() {
    Button button = new Button(this);
    button.setText("Open New Activity");
    button.setOnClickListener(v -> {
        Intent intent = new Intent(MainActivity.this, SecondActivity.class);
        startActivity(intent);
    });
    return button;
}

private View createImplicitIntentButton() {
    Button button = new Button(this);
    button.setText("Open Website");
    button.setOnClickListener(v -> {
        Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("https://www.google.com"));
        startActivity(intent);
    });
    return button;
}

private View createLongPressButton() {
    Button button = new Button(this);
    button.setText("Long Press Me");
    button.setOnLongClickListener(v -> {
        Toast.makeText(MainActivity.this, "Long Press Detected!", Toast.LENGTH_SHORT).show();
        return true;
    });
    return button;
}

private View createListView() {
```

```
ListView listView = new ListView(this);
List<String> items = new ArrayList<>();
items.add("Apple");
items.add("Banana");
items.add("Cherry");
ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1, items);
listView.setAdapter(adapter);
listView.setOnItemClickListener((parent, view, position, id) -> {
    String selectedItem = ((TextView) view).getText().toString();
    Toast.makeText(MainActivity.this, "Clicked: " + selectedItem, Toast.LENGTH_SHORT).show(); });
return listView; }}
```

activity_main.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"
    android:background="#FAFAFA">
    <androidx.appcompat.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="?attr/colorPrimary"
        android:title="Intent & Event Demo"
        android:titleTextColor="@android:color/white"/>
    <Button
        android:id="@+id(btn_explicit_intent"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Explicit Intent" />
    <Button
```

```
    android:id="@+id	btn_implicit_intent"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Implicit Intent" />

<Button
    android:id="@+id	btn_long_press"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Long Press Event" />

<Button
    android:id="@+id	btn_listview"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="ListView with Adapter" />

<LinearLayout
    android:id="@+id	layout_container"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"/>

</LinearLayout>
```

SecondActivity.java

```
package com.example.myapplication;
import android.os.Bundle;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
public class SecondActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        TextView textView = new TextView(this);
        textView.setText("Welcome to the Second Activity!");
    }
}
```

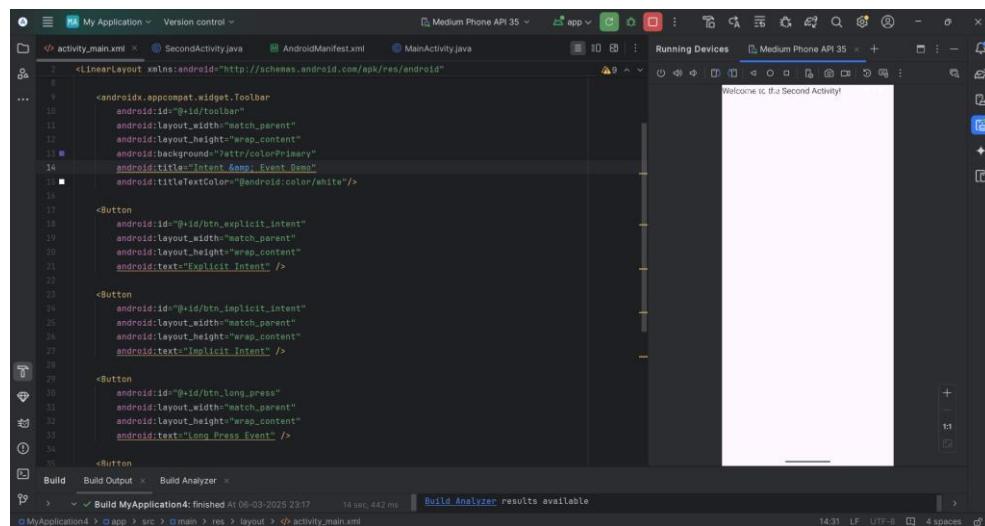
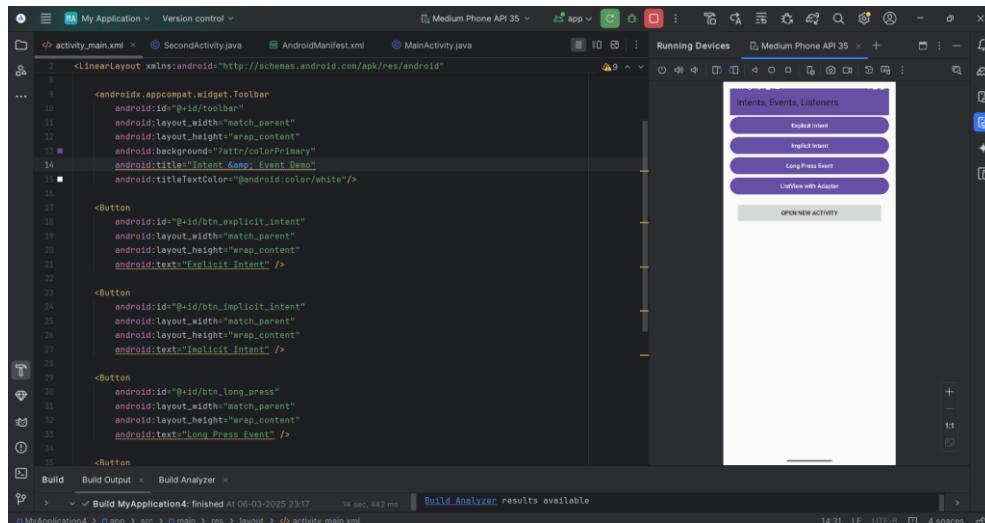
```
        textView.setTextSize(20);

        setContentView(textView);

    }

}
```

Output:



```

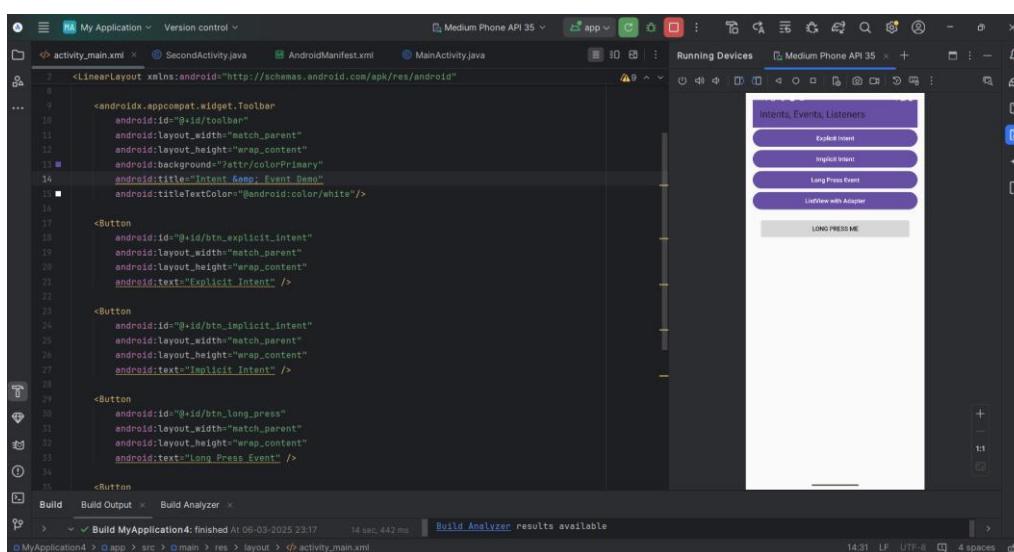
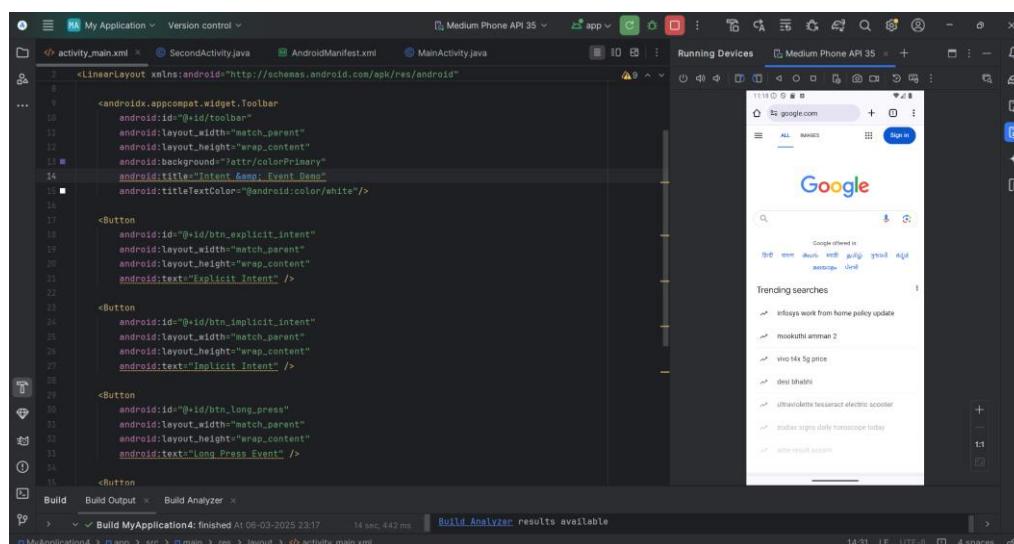
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="?attr/colorPrimary"
    android:title="Intent & Event Demo"
    android:titleTextColor="@android:color/white"/>

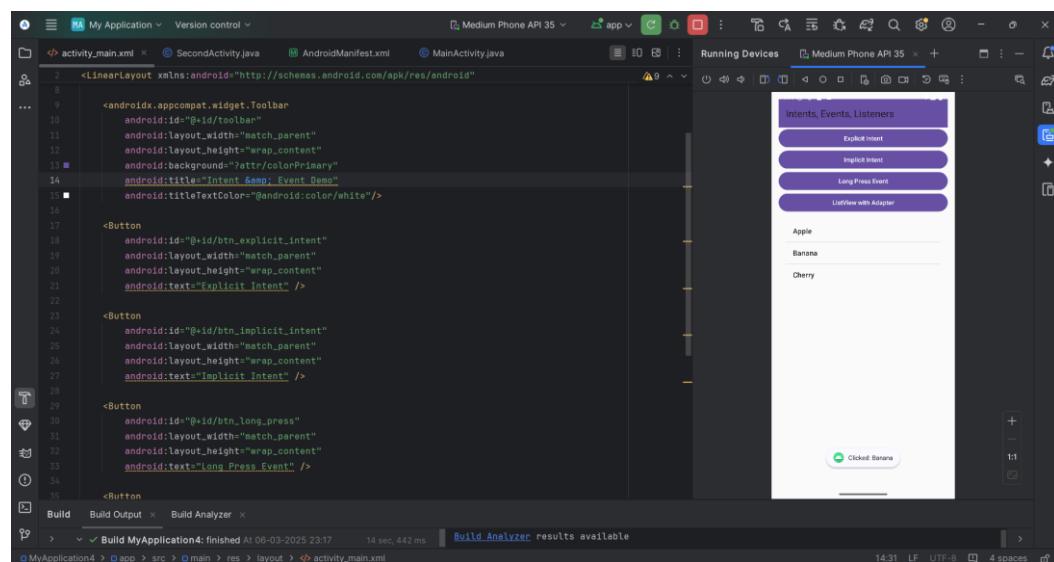
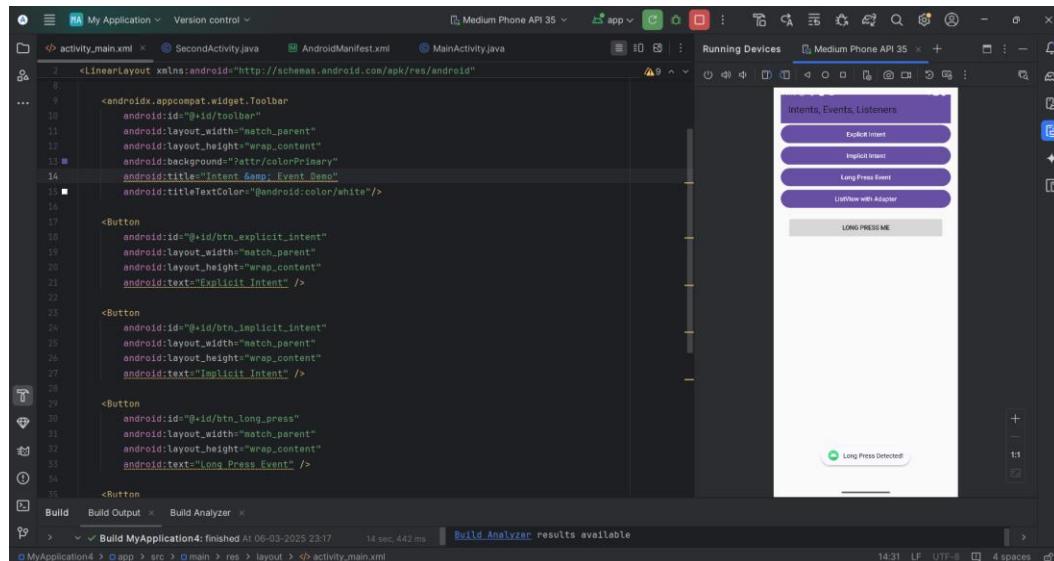
<Button
    android:id="@+id/btn_explicit_intent"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Explicit Intent" />

<Button
    android:id="@+id/btn_implicit_intent"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Implicit Intent" />

<Button
    android:id="@+id/btn_long_press"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Long Press Event" />

```





Conclusion: Successfully performed this practical.

Practical 7**Aim:** Programs on Services, notification and broadcast receivers**Code:**

Main Activity.java

```
package com.example.myapplication;
import android.Manifest;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.app.Service;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.pm.PackageManager;
import android.os.Build;
import android.os.Bundle;
import android.os.IBinder;
import android.widget.Button;
import android.widget.LinearLayout;
import android.widget.Toast;
import androidx.annotation.Nullable;
import androidx.annotation.RequiresApi;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.app.NotificationCompat;

public class MainActivity extends AppCompatActivity {
    private LinearLayout container;
    private AirplaneModeReceiver receiver;
    private static final String CHANNEL_ID = "my_channel";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        container = findViewById(R.id.layout_container);
        // Request Notification Permission (Android 13+)
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.TIRAMISU) {
            if (ActivityCompat.checkSelfPermission(this, Manifest.permission.POST_NOTIFICATIONS) != PackageManager.PERMISSION_GRANTED) {
                ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.POST_NOTIFICATIONS}, 101);
            }
        }
        setupButton(R.id.btn_service, () -> startService(new Intent(this, MyService.class)));
        setupButton(R.id.btn_notification, this::showNotification);
        setupButton(R.id.btn_broadcast, this::registerReceiver);
    }
    private void setupButton(int buttonId, Runnable action) {
        Button button = findViewById(buttonId);
        button.setOnClickListener(v -> action.run());
    }
}
```

```
// Foreground Service
public static class MyService extends Service {
    @Override
    public int onStartCommand(Intent intent, int flags, int startId) {
        Toast.makeText(this, "Service Started", Toast.LENGTH_SHORT).show();
        return START_STICKY;
    }
    @Nullable
    @Override
    public IBinder onBind(Intent intent) {
        return null;
    }
}
// Show Notification
private void showNotification() {
    NotificationManager manager = (NotificationManager)
getSystemService(Context.NOTIFICATION_SERVICE);
    // Create Notification Channel (For Android 8.0+)
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        NotificationChannel channel = new NotificationChannel(
            CHANNEL_ID, "My Notifications", NotificationManager.IMPORTANCE_HIGH);
        manager.createNotificationChannel(channel);
    }
    // Create Notification Intent
    Intent intent = new Intent(this, MainActivity.class);
    intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_CLEAR_TASK);
    PendingIntent pendingIntent = PendingIntent.getActivity(
        this, 0, intent, PendingIntent.FLAG_UPDATE_CURRENT | PendingIntent.FLAG_IMMUTABLE );
    // Build Notification
    NotificationCompat.Builder builder = new NotificationCompat.Builder(this, CHANNEL_ID)
        .setSmallIcon(R.drawable.ic_launcher_foreground) // Ensure this drawable exists
        .setContentTitle("New Notification")
        .setContentText("This is a sample notification.")
        .setAutoCancel(true) // Dismiss when clicked
        .setPriority(NotificationCompat.PRIORITY_HIGH) // Ensure it's visible
        .setContentIntent(pendingIntent); // Opens app when clicked
    // Show Notification
    manager.notify(1, builder.build());
}
// Broadcast Receiver for Airplane Mode
private void registerReceiver() {
    receiver = new AirplaneModeReceiver();
    IntentFilter filter = new IntentFilter(Intent.ACTION_AIRPLANE_MODE_CHANGED);
    registerReceiver(receiver, filter);
    Toast.makeText(this, "Broadcast Receiver Registered", Toast.LENGTH_SHORT).show();
}
public static class AirplaneModeReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        if (Intent.ACTION_AIRPLANE_MODE_CHANGED.equals(intent.getAction())) {
            boolean isOn = intent.getBooleanExtra("state", false);
            Toast.makeText(context, "Airplane Mode: " + (isOn ? "ON" : "OFF"), Toast.LENGTH_SHORT).show();
        }
    }
    @Override
    protected void onDestroy() {

```

```
super.onDestroy();
if(receiver != null) unregisterReceiver(receiver); }}
```

activity_main.xml

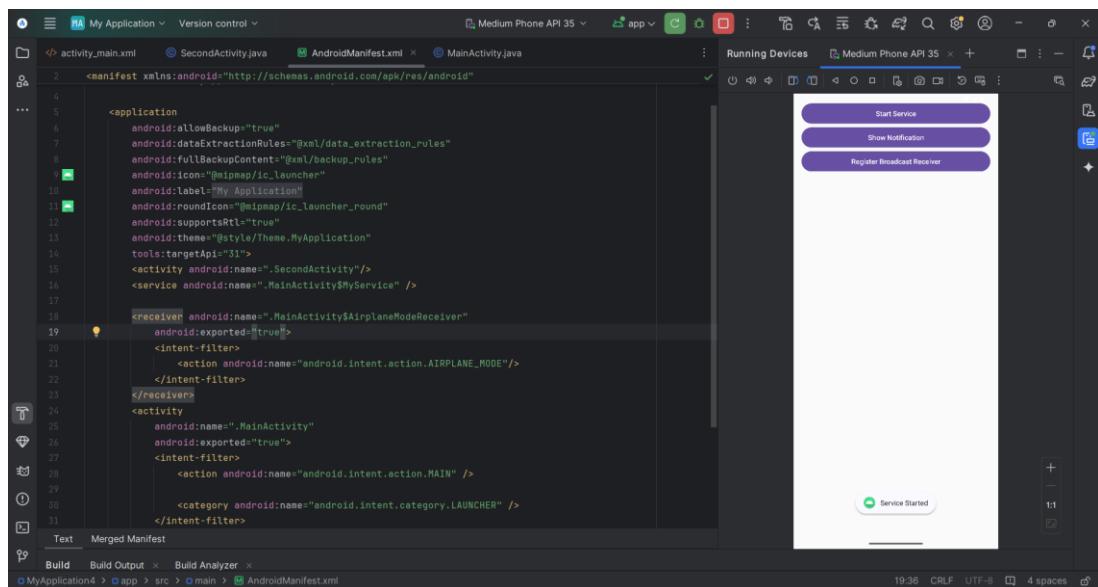
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    android:background="#FAFAFA">
    <Button
        android:id="@+id/btn_service"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Start Service" />
    <Button
        android:id="@+id/btn_notification"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Show Notification" />
    <Button
        android:id="@+id/btn_broadcast"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Register Broadcast Receiver" />
    <LinearLayout
        android:id="@+id/layout_container"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:padding="16dp"/>
</LinearLayout>
```

activity_main.xml

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.myapplication">
    <!-- Permissions -->
    <uses-permission android:name="android.permission.POST_NOTIFICATIONS"/>
    <application
        android:allowBackup="true"
        android:theme="@style/Theme.MyApplication"
        android:label="@string/app_name">
        <!--  Add android:exported="true" -->
        <activity
```

```
        android:name=".MainActivity"  
        android:exported="true">  
  
        <intent-filter>  
            <action android:name="android.intent.action.MAIN"/>  
            <category android:name="android.intent.category.LAUNCHER"/>  
        </intent-filter>  
  
    </activity>  
  
<!-- ✓ Foreground Service -->  
  
<service  
        android:name=".MainActivity$MyService"  
        android:exported="false"/>  
  
<!-- ✓ Broadcast Receiver -->  
  
<receiver  
        android:name=".MainActivity$AirplaneModeReceiver"  
        android:exported="true">  
  
        <intent-filter>  
            <action android:name="android.intent.action.AIRPLANE_MODE"/>  
        </intent-filter>  
  
    </receiver>  
  
</application>  
  
</manifest>
```

Output:



The screenshot shows the Android Studio interface with the manifest file open. The manifest includes declarations for a foreground service, a broadcast receiver for airplane mode changes, and an activity. A notification permission dialog is displayed on the emulator screen, asking if the application can send notifications with options to 'Allow' or 'Don't allow'. The status bar on the emulator shows signal strength, battery level, and connectivity.

```

<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    <application>
        <activity>
        </activity>
        <!-- Foreground Service -->
        <service
            android:name=".MainActivity$MyService"
            android:exported="false"/>
        <!-- Broadcast Receiver -->
        <receiver
            android:name=".MainActivity$AirplaneModeReceiver"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.AIRPLANE_MODE"/>
            </intent-filter>
        </receiver>
    </application>
</manifest>

```

The screenshot shows the Android Studio interface with the manifest file open. The manifest includes declarations for a foreground service, a broadcast receiver for airplane mode changes, and an activity. A notification is displayed on the emulator screen from the application, titled 'New Notification', stating 'This is a sample notification'. The status bar on the emulator shows signal strength, battery level, and connectivity.

```

<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    <application>
        <activity>
        </activity>
        <!-- Foreground Service -->
        <service
            android:name=".MainActivity$MyService"
            android:exported="false"/>
        <!-- Broadcast Receiver -->
        <receiver
            android:name=".MainActivity$AirplaneModeReceiver"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.AIRPLANE_MODE"/>
            </intent-filter>
        </receiver>
    </application>
</manifest>

```

The screenshot shows the Android Studio interface with the manifest file open. The manifest includes declarations for a foreground service, a broadcast receiver for airplane mode changes, and an activity. A broadcast receiver registration confirmation dialog is displayed on the emulator screen, stating 'Broadcast Receiver Registered'. The status bar on the emulator shows signal strength, battery level, and connectivity.

```

<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="My Application"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.MyApplication"
        tools:targetApi="31">
        <activity android:name=".SecondActivity" />
        <service android:name=".MainActivity$MyService" />
        <!-- Broadcast Receiver -->
        <receiver
            android:name=".MainActivity$AirplaneModeReceiver"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.AIRPLANE_MODE"/>
            </intent-filter>
        </receiver>
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>

```

Conclusion: Successfully performed this practical.

Practical 8(a)**Aim:** Database Programming with SQLite**Code:**

Main Activity.java

```
package com.example.myapplication;
import android.content.ContentValues;
import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.ListView;
import android.widget.SimpleCursorAdapter;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    private SQLiteDatabase db;
    private EditText etName, etAge;
    private ListView listView;
    private SimpleCursorAdapter adapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // UI Elements
        etName = findViewById(R.id.et_name);
        etAge = findViewById(R.id.et_age);
        listView = findViewById(R.id.list_view);
        setupDatabase(); // Create and open DB
        setupButton(R.id.btn_insert, this::insertData);
        setupButton(R.id.btn_view, this::viewData);
        setupButton(R.id.btn_delete, this::deleteAll);
    }
    private void setupDatabase() {
        try {
            db = new DatabaseHelper(this).getWritableDatabase();
            Toast.makeText(this, "Database Ready", Toast.LENGTH_SHORT).show();
        } catch (SQLException e) {
            Toast.makeText(this, "DB Error: " + e.getMessage(), Toast.LENGTH_LONG).show();
        }
    }
    private void setupButton(int buttonId, Runnable action) {
        Button button = findViewById(buttonId);
```

```
button.setOnClickListener(v -> action.run());  
}  
private void insertData() {  
    String name = etName.getText().toString();  
    String age = etAge.getText().toString();  
    if (name.isEmpty() || age.isEmpty()) {  
        Toast.makeText(this, "Enter name & age", Toast.LENGTH_SHORT).show();  
        return;  
    }  
    ContentValues values = new ContentValues();  
    values.put("name", name);  
    values.put("age", age);  
    long id = db.insert("users", null, values);  
  
    if (id > 0) {  
        Toast.makeText(this, "Inserted!", Toast.LENGTH_SHORT).show();  
        etName.setText("");  
        etAge.setText("");  
    } else {  
        Toast.makeText(this, "Insert Failed", Toast.LENGTH_SHORT).show();  
    }  
}  
private void ViewData() {  
    Cursor cursor = db.rawQuery("SELECT * FROM users", null);  
    if (cursor.getCount() == 0) {  
        Toast.makeText(this, "No data found", Toast.LENGTH_SHORT).show();  
        return;  
    }  
    String[] from = {"_id", "name", "age"};  
    int[] to = {R.id.text_id, R.id.text_name, R.id.text_age};  
    adapter = new SimpleCursorAdapter(this, R.layout.list_item, cursor, from, to, 0);  
    listView.setAdapter(adapter);  
}  
private void deleteAll() {  
    db.execSQL("DELETE FROM users");  
    Toast.makeText(this, "All Data Deleted", Toast.LENGTH_SHORT).show();  
    ViewData();  
}  
// Database Helper Class (Inside MainActivity)  
private static class DatabaseHelper extends SQLiteOpenHelper {  
    private static final String DB_NAME = "users.db";  
    private static final int DB_VERSION = 1;  
    public DatabaseHelper(MainActivity context) {  
        super(context, DB_NAME, null, DB_VERSION);  
    }  
    @Override  
    public void onCreate(SQLiteDatabase db) {  
        db.execSQL("CREATE TABLE users (_id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT,  
        age TEXT)");  
    }  
    @Override
```

```
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
    db.execSQL("DROP TABLE IF EXISTS users");  
    onCreate(db);  
}  
}  
}
```

activity_main.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"  
    android:background="#FAFAFA">  
  
<EditText  
    android:id="@+id/et_name"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:hint="Enter Name" />  
  
<EditText  
    android:id="@+id/et_age"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:hint="Enter Age"  
    android:inputType="number" />  
  
<Button  
    android:id="@+id/btn_insert"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="Insert Data" />  
  
<Button  
    android:id="@+id/btn_view"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="View Data" />  
  
<Button  
    android:id="@+id/btn_delete"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="Delete All" />  
  
<ListView  
    android:id="@+id/list_view"  
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent" />
</LinearLayout>
```

list_item.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"
    android:background="#FAFAFA">

    <EditText
        android:id="@+id/et_name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Name" />

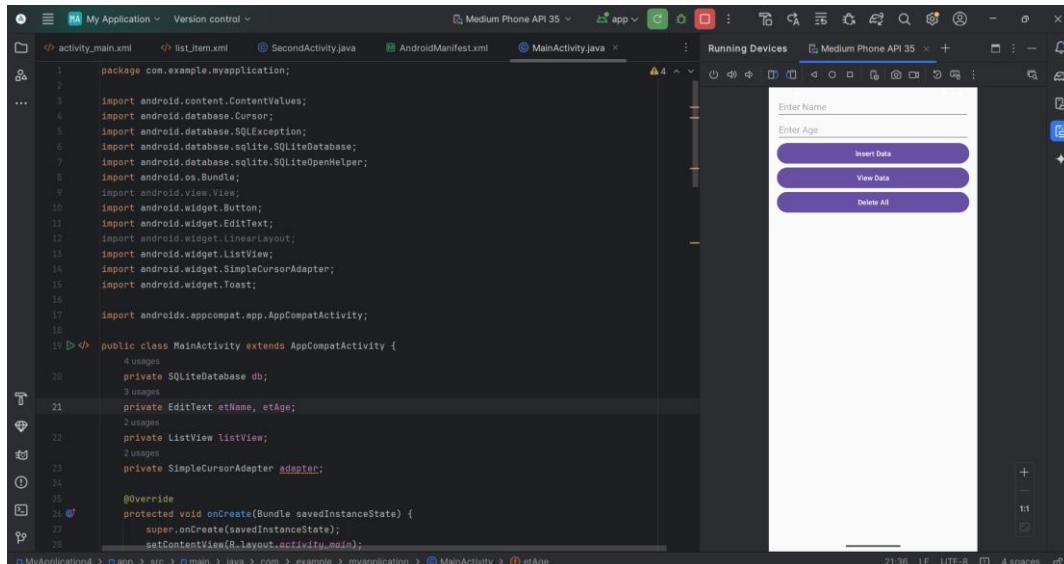
    <EditText
        android:id="@+id/et_age"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Age"
        android:inputType="number" />

    <Button
        android:id="@+id/btn_insert"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Insert Data" />

    <Button
        android:id="@+id/btn_view"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="View Data" />

    <Button
        android:id="@+id/btn_delete"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Delete All" />

    <ListView
        android:id="@+id/list_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
</LinearLayout>
```

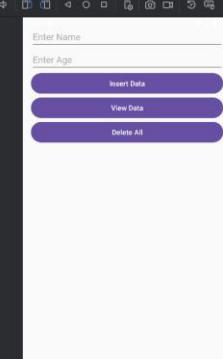
Output:


MainActivity.java code:

```

1 package com.example.myapplication;
2
3 import android.content.ContentValues;
4 import android.database.Cursor;
5 import android.database.SQLException;
6 import android.database.sqlite.SQLiteDatabase;
7 import android.database.sqlite.SQLiteOpenHelper;
8 import android.os.Bundle;
9 import android.view.View;
10 import android.widget.Button;
11 import android.widget.EditText;
12 import android.widget.LinearLayout;
13 import android.widget.ListView;
14 import android.widget.SimpleCursorAdapter;
15 import android.widget.Toast;
16
17 import androidx.appcompat.app.AppCompatActivity;
18
19 public class MainActivity extends AppCompatActivity {
20     private SQLiteDatabase db;
21     private EditText etName, etAge;
22     private ListView listView;
23     private SimpleCursorAdapter adapter;
24
25     @Override
26     protected void onCreate(Bundle savedInstanceState) {
27         super.onCreate(savedInstanceState);
28         setContentView(R.layout.activity_main);
    }
  
```

Running Device Screenshot:



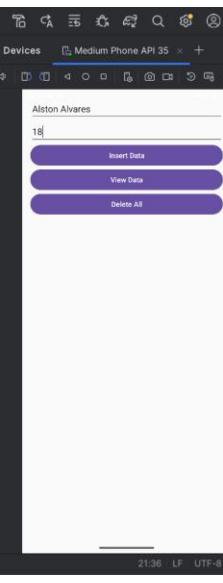
Alston Alvares

18

Insert Data

View Data

Delete All



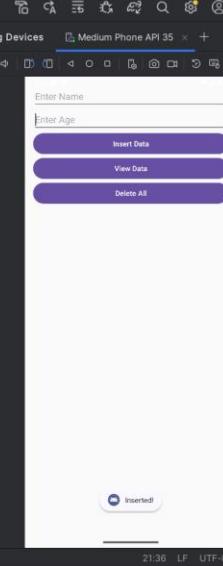
Alston Alvares

18

Insert Data

View Data

Delete All



Inserted!

The screenshot shows the Android Studio interface. The left pane displays the Java code for `MainActivity.java`. The right pane shows a preview of the application running on a "Medium Phone API 35" device. The application interface includes an "Enter Name" input field, an "Enter Age" input field, and three buttons: "Insert Data", "View Data", and "Delete All". Below these buttons is a list view showing a single item: "1 Alston Alvares 18".

```
1 package com.example.myapplication;
2
3 import android.content.ContentValues;
4 import android.database.Cursor;
5 import android.database.SQLException;
6 import android.database.sqlite.SQLiteDatabase;
7 import android.database.sqlite.SQLiteOpenHelper;
8 import android.os.Bundle;
9 import android.view.View;
10 import android.widget.Button;
11 import android.widget.EditText;
12 import android.widget.LinearLayout;
13 import android.widget.ListView;
14 import android.widget.SimpleCursorAdapter;
15 import android.widget.Toast;
16
17 import androidx.appcompat.app.AppCompatActivity;
18
19 public class MainActivity extends AppCompatActivity {
20     private SQLiteDatabase db;
21     private EditText etName, etAge;
22     private ListView listView;
23     private SimpleCursorAdapter adapter;
24
25     @Override
26     protected void onCreate(Bundle savedInstanceState) {
27         super.onCreate(savedInstanceState);
28         setContentView(R.layout.activity_main);
```

This screenshot is similar to the one above, showing the same Java code in `MainActivity.java`. However, the preview on the right shows a list view with a message "No data found" instead of the previous data entry.

```
1 package com.example.myapplication;
2
3 import android.content.ContentValues;
4 import android.database.Cursor;
5 import android.database.SQLException;
6 import android.database.sqlite.SQLiteDatabase;
7 import android.database.sqlite.SQLiteOpenHelper;
8 import android.os.Bundle;
9 import android.view.View;
10 import android.widget.Button;
11 import android.widget.EditText;
12 import android.widget.LinearLayout;
13 import android.widget.ListView;
14 import android.widget.SimpleCursorAdapter;
15 import android.widget.Toast;
16
17 import androidx.appcompat.app.AppCompatActivity;
18
19 public class MainActivity extends AppCompatActivity {
20     private SQLiteDatabase db;
21     private EditText etName, etAge;
22     private ListView listView;
23     private SimpleCursorAdapter adapter;
24
25     @Override
26     protected void onCreate(Bundle savedInstanceState) {
27         super.onCreate(savedInstanceState);
28         setContentView(R.layout.activity_main);
```

Conclusion: Successfully performed this practical.

Practical 8(b)

Aim: Programming Network Communications and Services (JSON).

Code:

Main Activity.java

```
package com.example.myapplication;

import android.os.AsyncTask;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ListView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

    private static final String API_URL = "https://jsonplaceholder.typicode.com/users"; // Sample API
    private ListView listView;
    private ArrayList<String> userList;
    private ArrayAdapter<String> adapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
```

```
listView = findViewById(R.id.list_view);
Button fetchButton = findViewById(R.id.btn_fetch);
userList = new ArrayList<>();
adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1, userList);
listView.setAdapter(adapter);
fetchButton.setOnClickListener(v -> new FetchDataTask().execute(API_URL));
}

// AsyncTask to Fetch JSON Data

private class FetchDataTask extends AsyncTask<String, Void, String> {
    @Override
    protected String doInBackground(String... urls) {
        StringBuilder result = new StringBuilder();
        try {
            URL url = new URL(urls[0]);
            HttpURLConnection connection = (HttpURLConnection) url.openConnection();
            connection.setRequestMethod("GET");
            BufferedReader reader = new BufferedReader(new InputStreamReader(connection.getInputStream()));
            String line;
            while ((line = reader.readLine()) != null) {
                result.append(line);
            }
            reader.close();
        } catch (Exception e) {
            Log.e("NetworkError", "Error fetching data", e);
            return null;
        }
        return result.toString();
    }
    @Override
    protected void onPostExecute(String json) {
        if (json != null) {
            parseJSON(json);
        } else {
    
```

```
        Toast.makeText(MainActivity.this, "Failed to fetch data", Toast.LENGTH_SHORT).show();
    }
}

private void parseJSON(String json) {
    userList.clear();
    try {
        JSONArray jsonArray = new JSONArray(json);
        for (int i = 0; i < jsonArray.length(); i++) {
            JSONObject user = jsonArray.getJSONObject(i);
            String name = user.getString("name");
            String email = user.getString("email");
            userList.add(name + " - " + email);
        }
        adapter.notifyDataSetChanged();
    } catch (JSONException e) {
        Log.e("JSONException", "Error parsing JSON", e);
        Toast.makeText(this, "Error parsing JSON", Toast.LENGTH_SHORT).show();
    }
}
```

activity_main.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"
    android:background="#FAFAFA">
    <Button
        android:id="@+id	btn_fetch"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Fetch Data" />
    <ListView
```

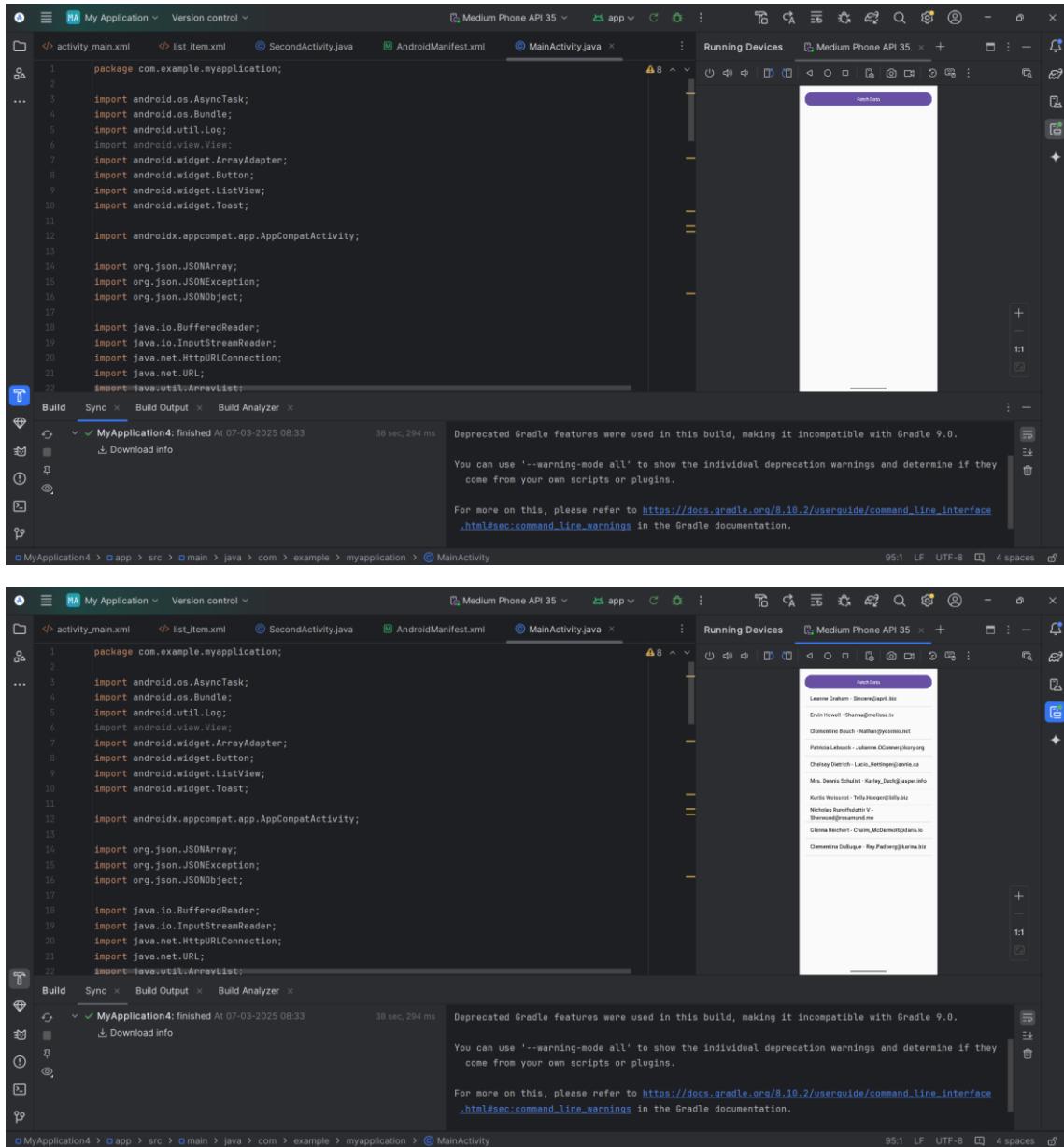
```

    android:id="@+id/list_view"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />
</LinearLayout>

```

AndroidManifest.xml

```
<uses-permission android:name="android.permission.INTERNET"/>
```

Output:

Conclusion: Successfully performed this practical.