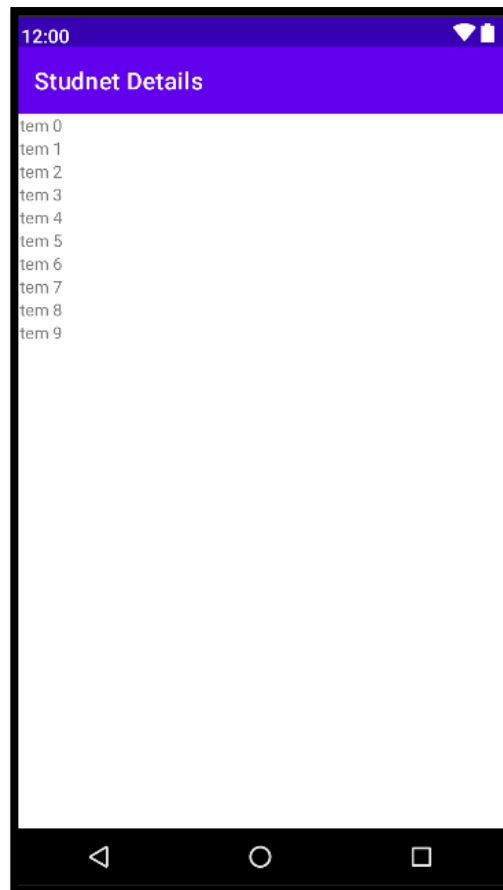


PRACTICAL 6

AIM: Create a Recycler View and list the details of student using following fields: 1. Name
2. Address 3. Photo (Image) 4. Delete (Button Operation)

- **activity_main.xml**

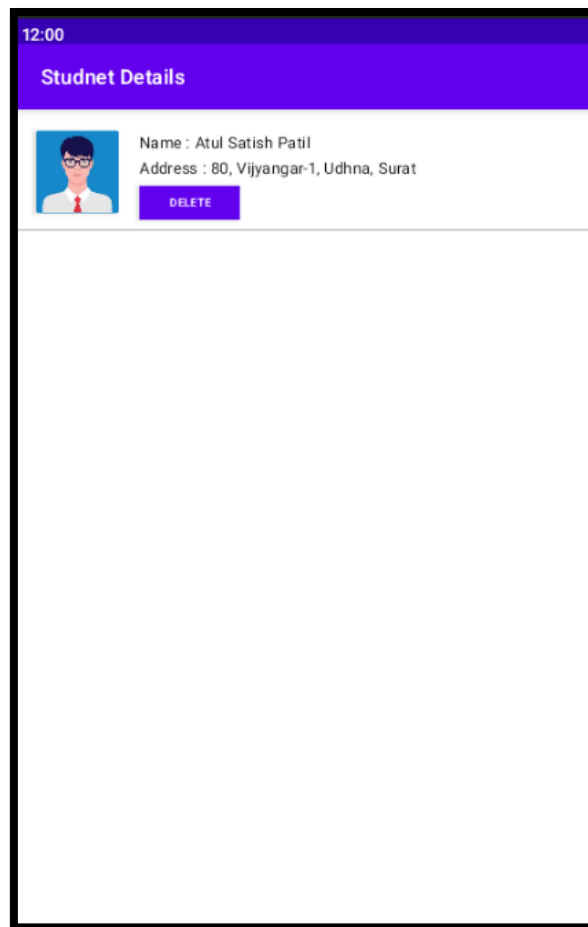


Layout: Linear Layout

Widgets:

RecyclerView(Id: "@+id/recycleView")

- student_row.xml



Layout: Relative Layout

Widgets:

```

CardView(Id: "@+id/imgview")
ImageView(Id: "@+id/imgview1")
TextView(Id: "@+id/textview1", text: "Name : ")
TextView(Id: "@+id/textview2", text: "Atul Satish Patil ")
TextView(Id: "@+id/textview3", text: "Address : ")
TextView(Id: "@+id/textview4", text: "80, Vijyangar-1, Udhna, Surat")
Button (Id: "@+id/deleteBtnView", text: "DELETE" )
TextView(Id: "@+id/divider", text: "_____")

```

- **MainActivity.java**

```
package com.example.recyclerview;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import android.os.Bundle;
import java.util.ArrayList;
import java.util.List;

public class MainActivity extends AppCompatActivity {

    RecyclerView recyclerView;
    LinearLayoutManager layoutManager;
    List<ModelClass> studentList;
    Adapter adapter;

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

        initData();
        initRecyclerView();
    }

    private void initData() {
        studentList = new ArrayList<>();
        studentList.add(new ModelClass(R.drawable._boy1, "Name : ", "Atul Patil", "Address",
"80, Vijyangar-1, Udhna, Surat", "_____",
"DELETE"));
        studentList.add(new ModelClass(R.drawable._boy2, "Name : ", "Joseph Williams",
"Address", "81, Vijyangar-1, Udhna, Surat",
"_____ ", "DELETE"));
        studentList.add(new ModelClass(R.drawable._boy3, "Name : ", "Caleb Hernandez",
"Address", "82, Vijyangar-1, Udhna, Surat",
"_____ ", "DELETE"));
        studentList.add(new ModelClass(R.drawable._boy4, "Name : ", "Edward Hughes",
"Address", "83, Vijyangar-1, Udhna, Surat",
"_____ ", "DELETE"));
        studentList.add(new ModelClass(R.drawable._boy5, "Name : ", "Lawrence Waters",
"Address", "84, Vijyangar-1, Udhna, Surat",
"_____ ", "DELETE"));
        studentList.add(new ModelClass(R.drawable._boy6, "Name : ", "Michael Rodgers",
"Address", "85, Vijyangar-1, Udhna, Surat",
"_____ ", "DELETE"));
        studentList.add(new ModelClass(R.drawable._boy7, "Name : ", "Alan Clay", "Address",
"86, Vijyangar-1, Udhna, Surat", "_____ ",
"DELETE"));
    }
}
```

```

        studentList.add(new ModelClass(R.drawable._girl1, "Name : ", "Deanna Watkins",
"Address", "87, Vijyangar-1, Udhna, Surat",
"_____", "DELETE"));
        studentList.add(new ModelClass(R.drawable._girl2, "Name : ", "Melissa Olsen",
"Address", "88, Vijyangar-1, Udhna, Surat",
"_____", "DELETE"));
    }

    private void initRecylcerView() {
        recyclerView = findViewById(R.id.recycleView);
        layoutManager = new LinearLayoutManager(this);
        layoutManager.setOrientation(RecyclerView.VERTICAL);
        recyclerView.setLayoutManager(layoutManager);
        adapter = new Adapter(studentList);
        recyclerView.setAdapter(adapter);
        adapter.notifyDataSetChanged();
    }
}

```

- **ModelClass.java**

```
package com.example.recyclerview;
```

```

public class ModelClass {
    private int imgview1;
    private String textview1;
    private String textview2;
    private String textview3;
    private String textview4;
    private String divider;
    private String deleteBtnView;

    ModelClass(int imageview1, String textview1, String textview2, String textview3,String
textview4,String divider, String deleteBtnView){
        this.imgview1 = imageview1;
        this.textview1 = textview1;
        this.textview2 = textview2;
        this.textview3 = textview3;
        this.textview4 = textview4;
        this.divider = divider;
        this.deleteBtnView = deleteBtnView;
    }

    public int getImageview1() {
        return imgview1;
    }

    public String getTextview1() {
        return textview1;
    }
}

```

```

public String getTextView2() {
    return textview2;
}

public String getTextView3() {
    return textview3;
}

public String getTextView4() {
    return textview4;
}

public String getDivider() {
    return divider;
}

public String getDeleteBtnView() {
    return deleteBtnView;
}
}

```

- **Adapter.java**

```

package com.example.recyclerview;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import java.util.List;

public class Adapter extends RecyclerView.Adapter<Adapter.ViewHolder> {

    private List<ModelClass> studentList;
    public Adapter(List<ModelClass>studentList){ this.studentList = studentList;}

    @NonNull
    @Override
    public Adapter.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.student_row, parent, false);
        return new ViewHolder(view);
    }

    @Override

```

```

public void onBindViewHolder(@NonNull Adapter.ViewHolder holder, int position) {
    int img_resource = studentList.get(position).getImageview1();
    String name = studentList.get(position).getTextview1();
    String sname = studentList.get(position).getTextview2();
    String add = studentList.get(position).getTextview3();
    String sadd = studentList.get(position).getTextview4();
    String btn = studentList.get(position).getDeleteBtnView();
    String line = studentList.get(position).getDivider();

    holder.setData(img_resource, name, sname, add, sadd, btn, line);
    holder.delBtnView.setOnClickListener(new View.OnClickListener(){
        @Override
        public void onClick(View view) {
            AlertDialog.Builder builder = new AlertDialog.Builder(view.getContext())
                .setTitle("Delete Student")
                .setMessage("Are you sure want to delete?")
                .setIcon(R.drawable.ic_baseline_delete_24)
                .setPositiveButton("Yes", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {
                        studentList.remove(holder.getAdapterPosition());
                        notifyItemRemoved(holder.getAdapterPosition());
                    }
                })
                .setNegativeButton("No", new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i) {

                    }
                });
            builder.show();
        }
    });
}

@Override
public int getItemCount() {
    return studentList.size();
}

public class ViewHolder extends RecyclerView.ViewHolder{
    private ImageView imgview1;
    private TextView textview1;
    private TextView textview2;
    private TextView textview3;
    private TextView textview4;
    private Button delBtnView;
    private TextView dividerLine;

    public ViewHolder(@NonNull View itemView) {

```

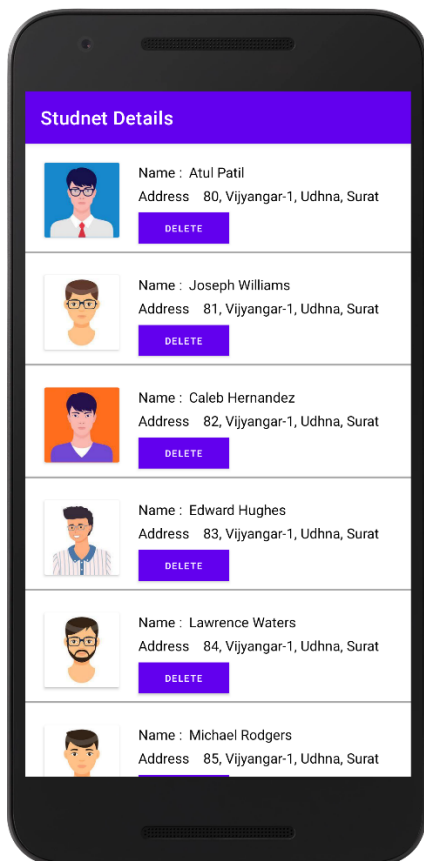
```

super(itemView);
imgview1 = itemView.findViewById(R.id.imgview1);
textview1 = itemView.findViewById(R.id.textview1);
textview2 = itemView.findViewById(R.id.textview2);
textview3 = itemView.findViewById(R.id.textview3);
textview4 = itemView.findViewById(R.id.textview4);
delBtnView = itemView.findViewById(R.id.deleteBtnView);
dividerLine = itemView.findViewById(R.id.divider);
}

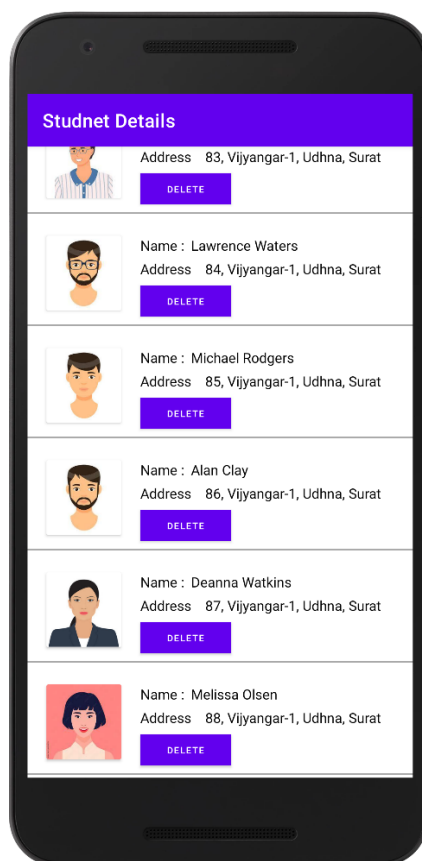
public void setData(int img_resource, String name, String sname, String add, String sadd,
String btn, String line) {
    imgview1.setImageResource(img_resource);
    textview1.setText(name);
    textview2.setText(sname);
    textview3.setText(add);
    textview4.setText(sadd);
    delBtnView.setText(btn);
    dividerLine.setText(line);
}
}
}
}

```

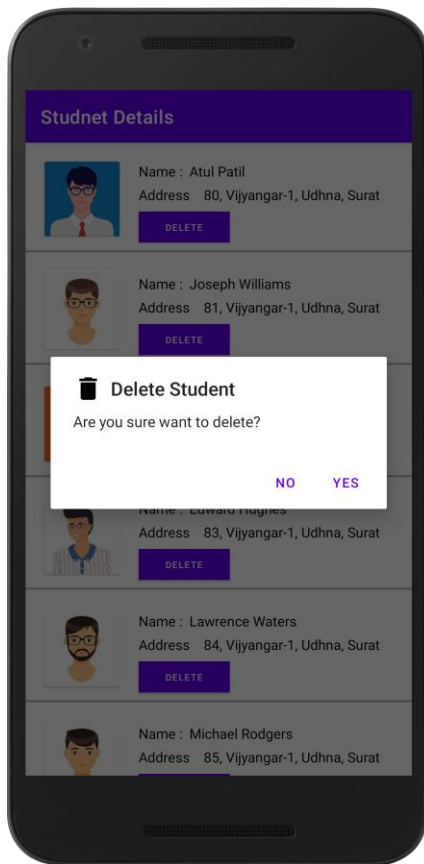
• Output



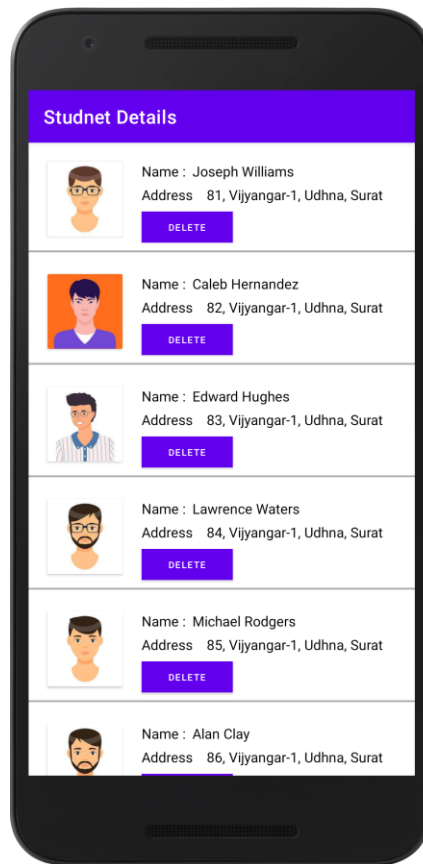
1. View Before Scrolling



2. View After Scrolling



3. Before removing first student

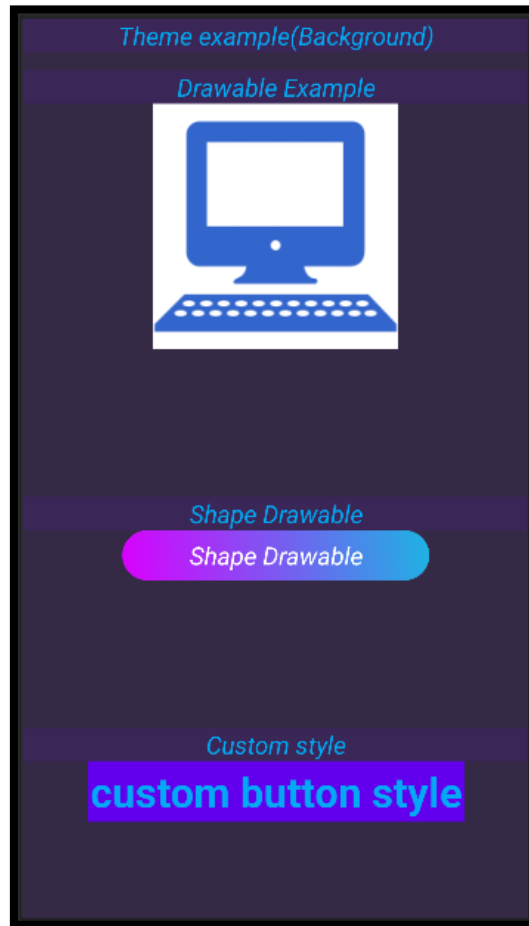


4. After removing first student

PRACTICAL 7

Aim: Theme, Custom Styles, Drawables.

- **activity_main.xml**



Layout: LinearLayout (Theme: mytheme)

Widgets:

TextView (text :Theme Example(background))

TextView (text :Drawable Example)

ImageView(src : @drawable/img)

TextView (text : Shape drawable)

Button(text: Shape Drawable, Background: @drawable/shape_drawable)

TextView (text :Custom Style)

Button(Text:custom button style, Style: @style/BoldStyle)

- **themes.xml**

```
<resources xmlns:tools="http://schemas.android.com/tools">
  <!-- Base application theme. -->
  <style name="Theme.Pr_7"
parent="Theme.MaterialComponents.DayNight.DarkActionBar">
    <!-- Primary brand color. -->
    <item name="colorPrimary">@color/teal_200</item>
    <item name="colorPrimaryVariant">#FFEB3B</item>
    <item name="colorOnPrimary">@color/white</item>
    <!-- Secondary brand color. -->
    <item name="colorSecondary">@color/teal_200</item>
    <item name="colorSecondaryVariant">@color/teal_700</item>
    <item name="colorOnSecondary">@color/black</item>
    <!-- Status bar color. -->
    <item name="android:statusBarColor"
tools:targetApi="l">?attr/colorPrimaryVariant</item>
    <!-- Customize your theme here. -->
  </style>
  <style name="MyTheme">
    <item name="android:textStyle">italic</item>
    <item name="android:textColor">#03A9F4</item>
    <item name="android:background">#197700FF</item>
    <item name="android:textSize">20sp</item>
  </style>
</resources>
```

- **MYStyle.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <style name="BoldStyle">
    <item name="android:textSize">34dp</item>
    <item name="android:textColor">#03A9F4</item>
    <item name="android:textStyle">bold</item>
  </style>
  <style name="buttonStyle">
    <item name="android:textSize">20sp</item>
    <item name="android:background">@drawable/shapedrawable</item>
    <item name="android:textColor">#fff</item>
  </style>
</resources>
```

- **ShapeDrawable.xml**

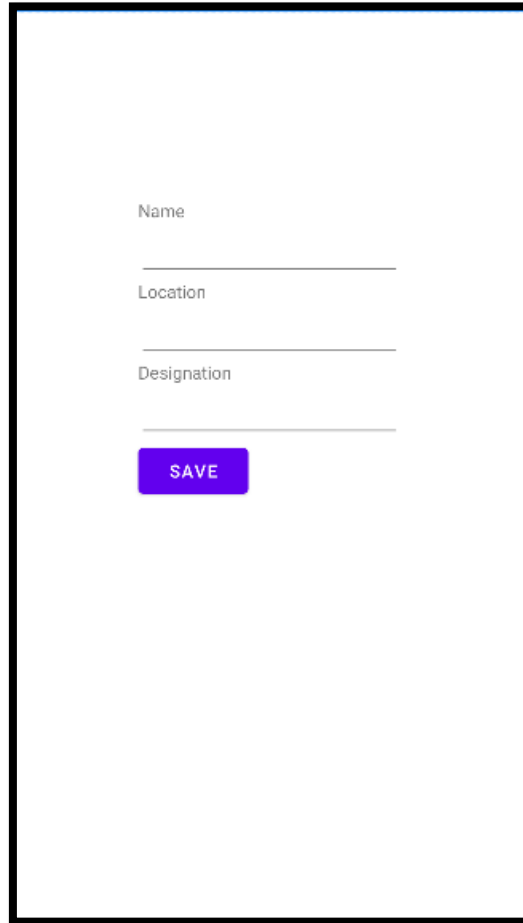
```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
  android:shape="rectangle">
  <gradient
    android:startColor="#000000"
    android:endColor="#2196F3"
```

```
        android:angle="45"/>
    <padding android:left="7dp"
        android:top="7dp"
        android:right="7dp"
        android:bottom="7dp" />
    <corners
        android:radius="50dp"/>
</shape>
```

PRACTICAL 8

Aim: Save user data in a database.

- **activity_main.xml**

A screenshot of a mobile application interface showing a form for saving user data. The form is enclosed in a black rectangular border. It contains three text input fields, each preceded by a label: 'Name', 'Location', and 'Designation'. Below the 'Designation' field is a purple button with the text 'SAVE' in white capital letters.

Layout: Linear Layout(orientation="vertical")

Widgets:

```
TextView(id="@+id/fstTxt", text: "Name")
EditText(id="@+id/txtName")
TextView(id="@+id/secTxt", text: "Location")
EditText(id="@+id/txtLocation")
TextView(id="@+id/thirdTxt", text="Designation")
EditText(id="@+id/txtDesignation")
Button (id="@+id/btnSave", text: "SAVE" )
```

- **MainActivity.java**

```
package com.example.pr_8;

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

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    EditText name, loc, desig;
    Button saveBtn;
    Intent intent;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        name = (EditText) findViewById(R.id.txtName);
        loc = (EditText) findViewById(R.id.txtLocation);
        desig = (EditText) findViewById(R.id.txtDesignation);
        saveBtn = (Button) findViewById(R.id.btnSave);
        saveBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String username = name.getText().toString() + "\n";
                String location = loc.getText().toString();
                String designation = desig.getText().toString();
                DbHandler dbHandler = new DbHandler(MainActivity.this);
                dbHandler.insertUserDetails(username, location, designation);
                Toast.makeText(getApplicationContext(), "Details Inserted Successfully",
                    Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

- **DbHandler.java**

```
package com.example.pr_8;

import android.content.ContentValues;
import android.content.Context;
import android.database.DatabaseErrorHandler;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.os.Build;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;

import java.util.ArrayList;
import java.util.HashMap;

public class DbHandler extends SQLiteOpenHelper {
    private static final String DB_NAME = "Register";
    private static final int DB_VERSION = 1;
    private static final String TABLE_NAME = "formTable";
    private static final String ID_COL = "id";
    private static final String USERNAME_COL = "username" ;
    private static final String LOCATION_COL = "location" ;
    private static final String DESIGNATION_COL = "designation" ;

    public DbHandler(@Nullable Context context) {
        super(context, DB_NAME, null, DB_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase sqLiteDatabase) {
        String query = "CREATE TABLE " + TABLE_NAME + " ("
            + ID_COL + " INTEGER PRIMARY KEY AUTOINCREMENT, "
            + USERNAME_COL + " TEXT,"
            + LOCATION_COL + " TEXT,"
            + DESIGNATION_COL + " TEXT)";
        sqLiteDatabase.execSQL(query);
    }

    @Override
    public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {
        sqLiteDatabase.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(sqLiteDatabase);
    }

    public void insertUserDetails(String username, String location, String designation) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(USERNAME_COL, username);
        values.put(LOCATION_COL, location);
    }
}
```

```

        values.put(DSIGNATION_COL, designation);
        db.insert(TABLE_NAME, null, values);
        db.close();
    }
}

```

- **Output**

PR-8

Name
Atul Patil

Location
Surat

Designation
Software Engineer

SAVE

Name	Permissions	Date	Size
> com.android.wallpaper.livepick	drwxrwx--x	2022-11-09 18:39	4 KB
> com.android.wallpaperbackup	drwxrwx--x	2022-11-09 18:39	4 KB
> com.breel.wallpapers18	drwxrwx--x	2022-11-09 18:39	4 KB
> com.example.email_app	drwxrwx--x	2022-11-09 18:39	4 KB
> com.example.pr_12	drwxrwx--x	2022-11-09 18:39	4 KB
> com.example.pr_12_broadcastre	drwxrwx--x	2022-11-09 18:39	4 KB
> com.example.pr_13_notification	drwxrwx--x	2022-11-09 18:39	4 KB
> com.example.pr_15	drwxrwx--x	2022-11-09 18:39	4 KB
> com.example.pr_16	drwxrwx--x	2022-11-09 18:39	4 KB
> com.example.pr_16_location	drwxrwx--x	2022-11-09 18:39	4 KB
> com.example.pr_3_intent	drwxrwx--x	2022-11-09 18:39	4 KB
▼ com.example.pr_8	drwxrwx--x	2022-11-09 18:39	4 KB
cache	drwxrws--x	2022-11-10 04:57	4 KB
code_cache	drwxrws--x	2022-11-10 04:57	4 KB
databases	drwxrwx--x	2022-11-10 04:58	4 KB
Register	-rw-rw----	2022-11-10 04:58	20 KB
Register-journal	-rw-rw----	2022-11-10 04:58	0 B
> com.example.practical2_linearl	drwxrwx--x	2022-11-09 18:39	4 KB
> com.example.practical14_chore	drwxrwx--x	2022-11-09 18:39	4 KB

DB Browser for SQLite - C:\Users\Hp\AppData\Local\Google\AndroidStudio2021.2\device-explorer\Nexus_5X

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Open Project Save

Database Structure Browse Data Edit Pragas Execute SQL

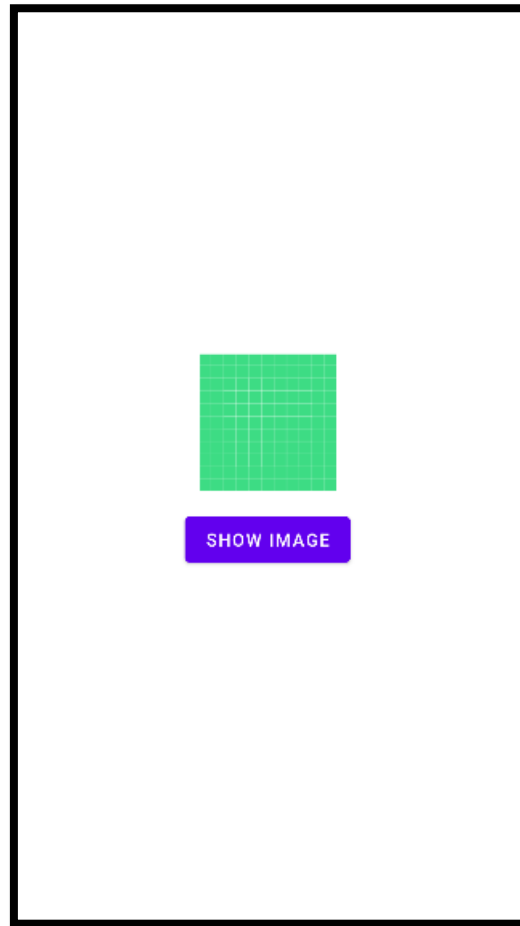
Table: formTable

	id	username	location	designation
	Filter	Filter	Filter	Filter
1	1	Atul...	Surat	Software Engineer

PRACTICAL 9

Aim: Use an AsyncTask to access remote database (make a use of simple PHP Web service).

- **activity_main.xml**



Layout: Linear Layout(orientation="vertical")

Widgets:

ImageView(Id: "@+id/ image", src="@drawable/ic_launcher_background")
Button (Id: "@+id/ showimage", text: "SHOW IMAGE")

- **MainActivity.java**

```
package com.example.pr_9;

import androidx.appcompat.app.AppCompatActivity;
import android.app.ProgressDialog;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.AsyncTask;
import android.os.Bundle;
```



```

import android.util.Log;
import android.widget.Button;
import android.widget.ImageView;
import java.io.InputStream;

public class MainActivity extends AppCompatActivity {

    private ImageView imageView;
    String apiUrl = "https://www.timesofsports.com/wp-content/uploads/2021/10/Suryakumar-
Yadav.png";
    ProgressDialog progressDialog;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button viewButton = findViewById(R.id.showimage);
        imageView = findViewById(R.id.image);
        viewButton.setOnClickListener(view -> {
            MyAsyncTask myAsyncTask = new MyAsyncTask();
            myAsyncTask.execute(apiUrl);
        });
    }
    public class MyAsyncTask extends AsyncTask<String, Void, Bitmap> {
        @Override
        protected void onPreExecute() {
            super.onPreExecute();
            progressDialog = new ProgressDialog(MainActivity.this);
            progressDialog.setMessage("Please Wait!");
            progressDialog.setCancelable(false);
            progressDialog.show();
        }
        @Override
        protected Bitmap doInBackground(String... URL) {
            String imageURL = URL[0];
            Bitmap bitmap = null;
            try {
                Log.d("", "In Try");
                InputStream inputStream = new java.net.URL(imageURL).openStream();
                bitmap = BitmapFactory.decodeStream(inputStream);
            } catch (Exception e) {
                e.printStackTrace();
                Log.d("", "In Catch ");
            }
            return bitmap;
        }
        @Override
        protected void onPostExecute(Bitmap result) {
            super.onPostExecute(result);
            imageView.setImageBitmap(result);
        }
    }
}

```

```

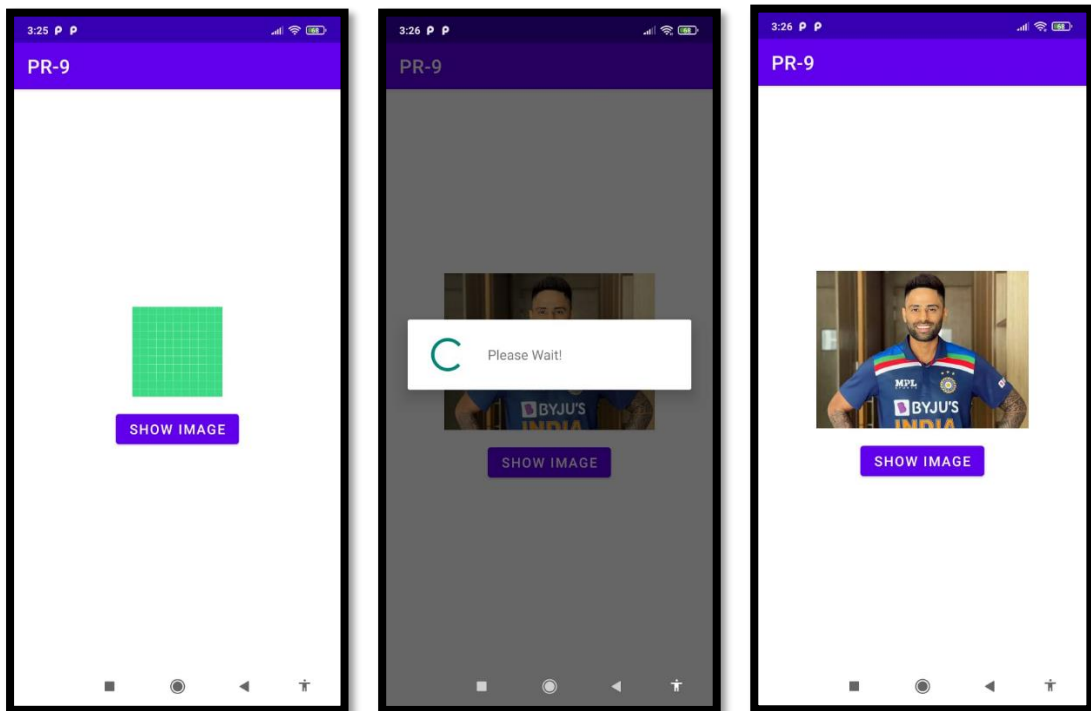
        progressDialog.dismiss();
    }
}

```

- **AndroidManifest.xml(add following permission)**

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

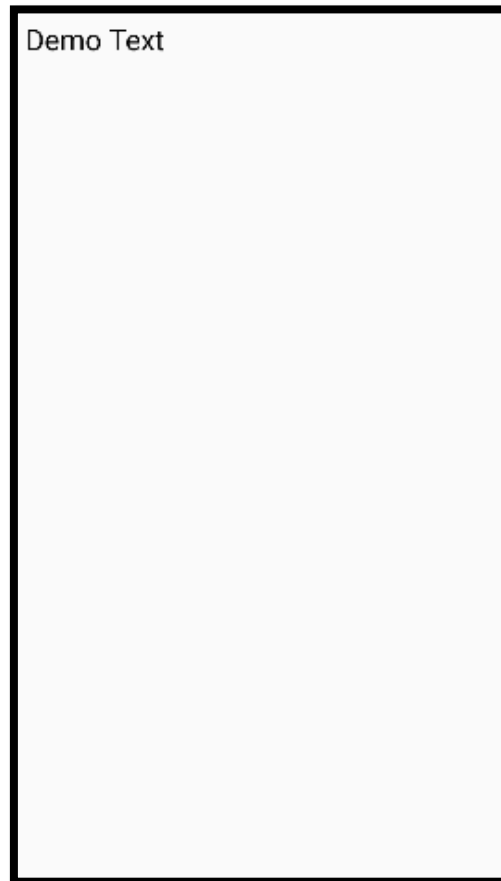
- **Output**



PRACTICAL 10

Aim: Use Retrofit to access remote database (make a use of simple PHP Web service).

- **Activity_main.xml**



Layout: LinearLayout

Widgets:

```
ScrollView(Id: @+id/ ScrollView)
TextView(Id: @+id/tv , Text:Demo Text)
```

- **Model.java**

```
package com.kazimasum.retrodemo;
public class model
{
    int userId,id;
    String title,body;

    public model(int userId, int id, String title, String body) {
        this.userId = userId;
        this.id = id;
        this.title = title;
        this.body = body;
    }
}
```

```

    }

    public int getUserId() {
        return userId;
    }

    public void setUserId(int userId) {
        this.userId = userId;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getTitle() {
        return title;
    }

    public void setTitle(String title) {
        this.title = title;
    }

    public String getBody() {
        return body;
    }

    public void setBody(String body) {
        this.body = body;
    }
}

```

- **myapi.java**

```

package com.kazimasum.retrodemo;

import java.util.List;

import retrofit2.Call;
import retrofit2.http.GET;

public interface myapi
{
    @GET("posts")
    Call<List<model>> getmodels();
}

```

- **MainActivity.java**

```

package com.kazimasum.retrodemo;
import androidx.annotation.Dimension;
import androidx.appcompat.app.AppCompatActivity;
import android.graphics.Paint;
import android.os.Bundle;
import android.widget.TextView;
import java.util.List;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
import retrofit2.Retrofit;
import retrofit2.converter.gson.GsonConverterFactory;

public class MainActivity extends AppCompatActivity
{
    TextView tv;
    String url="https://jsonplaceholder.typicode.com";
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        tv=(TextView)findViewById(R.id.tv);
        tv.setText("");

        Retrofit retrofit=new Retrofit.Builder()
            .baseUrl(url)
            .addConverterFactory(GsonConverterFactory.create())
            .build();

        myapi api=retrofit.create(myapi.class);

        Call<List<model>> call=api.getmodels();

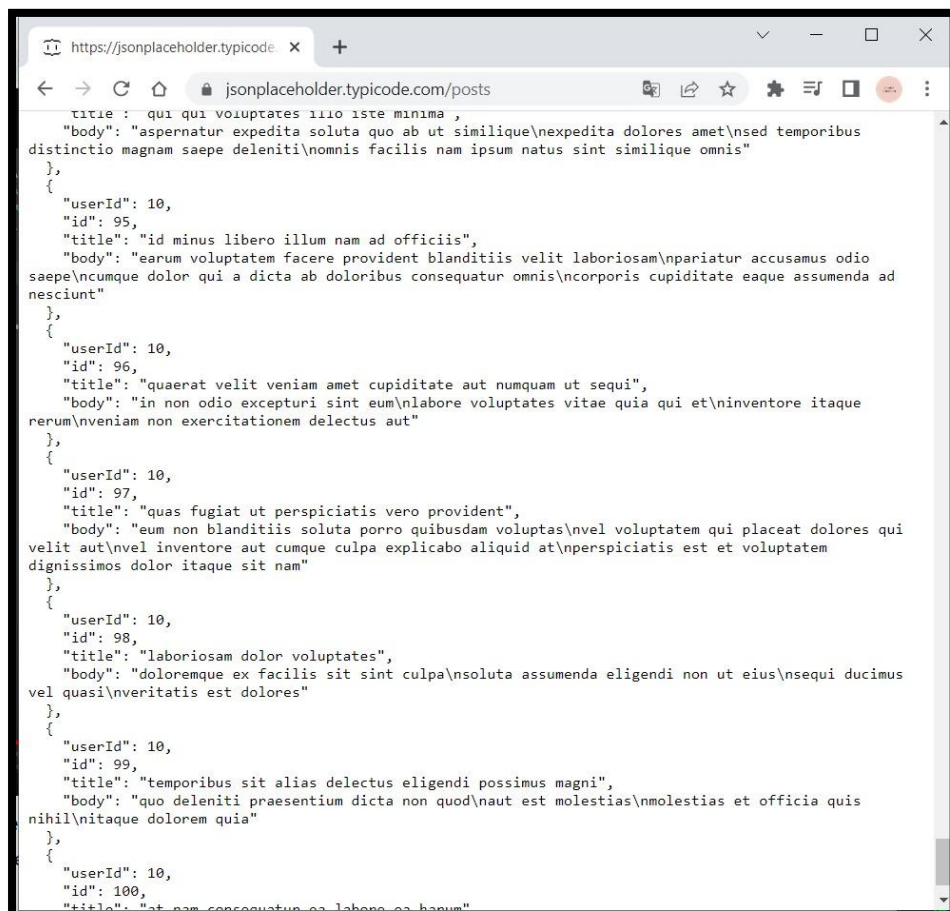
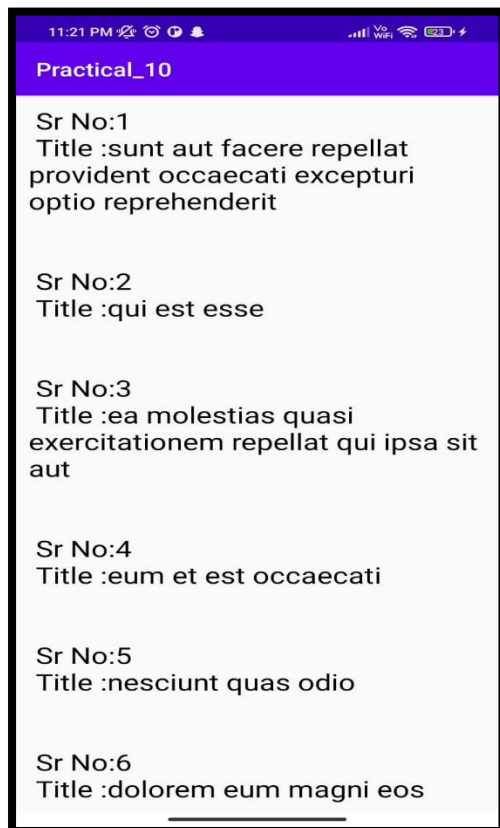
        call.enqueue(new Callback<List<model>>() {
            @Override
            public void onResponse(Call<List<model>> call, Response<List<model>> response) {
                List<model> data=response.body();
                for(int i=0; i<data.size();i++)
                    tv.append(" Sr No:"+data.get(i).getId()+" \n Title :"+data.get(i).getTitle()+"\n\n");
            }

            @Override
            public void onFailure(Call<List<model>> call, Throwable t) {

            }
        });
    }
}

```

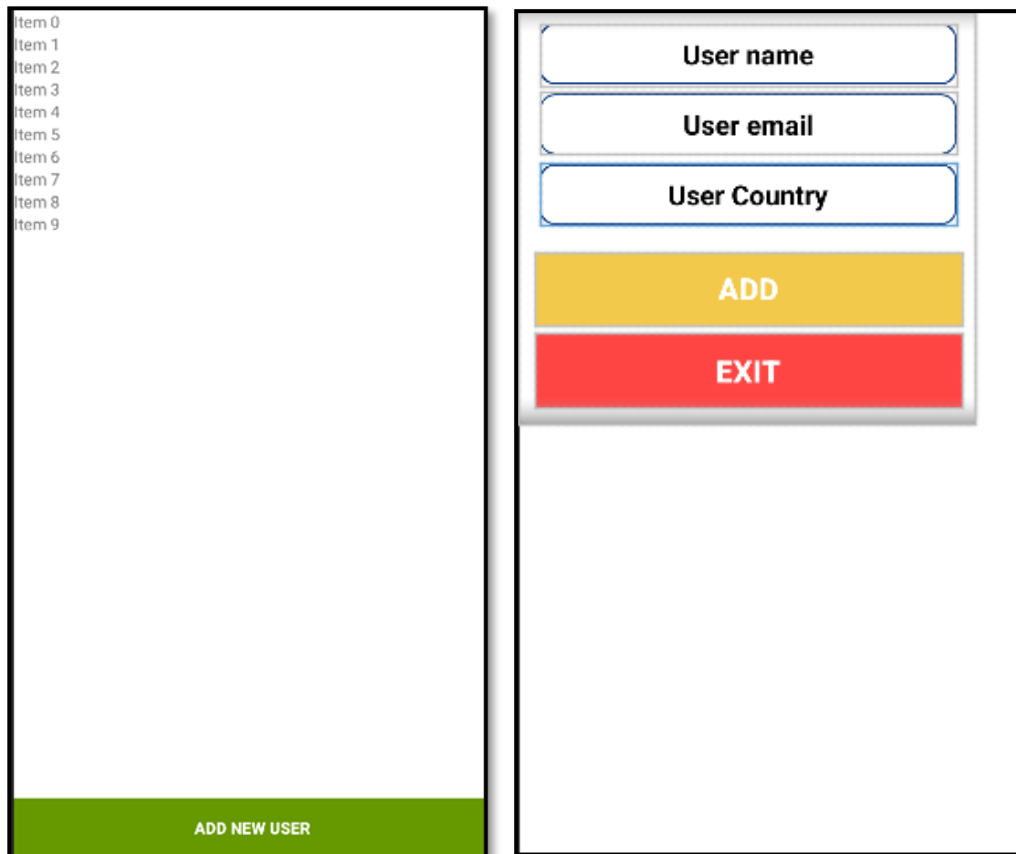
- Output



PRACTICAL 11

AIM: Use Firebase to perform CRUD operation.

- **activity_main.xml & alert_dialog_add_new_user.xml**



activity_main.xml

Layout: LinearLayout

Widgets: RecyclerView (Id: @+id/recyclerView)
Button (Id: @+id/AddBtn, Text: Add New User)

alert_dialog_add_new_user.xml

Layout: LinearLayout

Widgets: EditText (Id: @+id/username, Hint: User Name)
EditText (Id: @+id/email, Hint: User Email)
EditText (Id: @+id/country, Hint: Country)
Button (Id: @+id/AddBtn, Text: Add)
Button (Id: @+id/ExitBtn, Text: Add)

- User_Item.xml & View_dialog_confirm_Delete.xml

The image displays two side-by-side mobile application interface mockups. The left mockup represents the 'User_Item.xml' screen, featuring three text input fields labeled 'User name', 'User email', and 'User country'. Below these fields are two buttons: a red 'DELETE' button and a blue 'UPDATE' button. The right mockup represents the 'View_dialog_confirm_Delete.xml' screen, showing a confirmation dialog with the text 'Are you sure ?' at the top. Below the text are two buttons: a red 'YES, DELETE' button and a blue 'NO, EXIT' button.

- User_Item.xml

Layout: LinearLayout

Widgets: TextView (Id: @+id/usernameDisplay, Text: User Name)
 TextView (Id: @+id/emailDisplay, Text: User Email)
 TextView (Id: @+id/countryDisplay, Text: Country)
 Button (Id: @+id/Delete, Text: Delete)
 Button (Id: @+id/Update, Text: Update)

- View_dialog_confirm_Delete.xml

Layout: LinearLayout

Widgets: TextView (Text: Are You Sure?)
 Button (Id: @+id/DeleteBtn, Text: Yes, Delete)
 Button (Id: @+id/NoBtn, Text: NO, Exit)

- **UserItem.java**

```
package com.codingwithzo.realtimedbaddreadupdatedelete;
public class UsersItem {
    String userID;
    String userName;
    String userEmail;
    String userCountry;
    public UsersItem() {
    }
    public UsersItem(String userID, String userName, String userEmail, String userCountry) {
        this.userID = userID;
        this.userName = userName;
        this.userEmail = userEmail;
        this.userCountry = userCountry;
    }
    public String getUserID() {
        return userID;
    }
    public void setUserID(String userID) {
        this.userID = userID;
    }
    public String getUserName() {
        return userName;
    }
    public void setUserName(String userName) {
        this.userName = userName;
    }
    public String getUserEmail() {
        return userEmail;
    }
    public void setUserEmail(String userEmail) {
        this.userEmail = userEmail;
    }
    public String getUserCountry() {
        return userCountry;
    }
    public void setUserCountry(String userCountry) {
        this.userCountry = userCountry;
    }
}
```

- **UserRecyclerAdapter.java**

```

package com.codingwithzo.realtimedbaddreadupdatedelete;
import android.app.Dialog;
import android.content.Context;
import android.graphics.Color;
import android.graphics.drawable.ColorDrawable;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.view.Window;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import java.util.ArrayList;
import java.util.Date;
public class UsersRecyclerAdapter extends
RecyclerView.Adapter<UsersRecyclerAdapter.ViewHolder> {
    Context context;
    ArrayList<UsersItem> usersItemArrayList;
    DatabaseReference databaseReference;
    public UsersRecyclerAdapter(Context context, ArrayList<UsersItem> usersItemArrayList)
    {
        this.context = context;
        this.usersItemArrayList = usersItemArrayList;
        databaseReference = FirebaseDatabase.getInstance().getReference();
    }
    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        LayoutInflater inflater = LayoutInflater.from(context);
        View view = inflater.inflate(R.layout.user_item, parent, false);
        return new ViewHolder(view);
    }
    @Override
    public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
        UsersItem users = usersItemArrayList.get(position);
        holder.textName.setText("Name : " + users.getUserName());
        holder.textEmail.setText("Email : " + users.getUserEmail());
        holder.textCountry.setText("Country : " + users.getUserCountry());
        holder.buttonUpdate.setOnClickListener(new View.OnClickListener() {
            @Override

```

```

        public void onClick(View view) {
            ViewDialogUpdate viewDialogUpdate = new ViewDialogUpdate();
            viewDialogUpdate.showDialog(context, users.getUserID(), users.getUserName(),
users.getUserEmail(), users.getUserCountry());
        }
    });
    holder.buttonDelete.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            ViewDialogConfirmDelete viewDialogConfirmDelete = new
ViewDialogConfirmDelete();
            viewDialogConfirmDelete.showDialog(context, users.getUserID());
        }
    });
}
@Override
public int getItemCount() {
    return usersItemArrayList.size();
}
public static class ViewHolder extends RecyclerView.ViewHolder {
    TextView textName;
    TextView textEmail;
    TextView textCountry;
    Button buttonDelete;
    Button buttonUpdate;
    public ViewHolder(@NonNull View itemView) {
        super(itemView);
        textName = itemView.findViewById(R.id.textName);
        textEmail = itemView.findViewById(R.id.textEmail);
        textCountry = itemView.findViewById(R.id.textCountry);
        buttonDelete = itemView.findViewById(R.id.buttonDelete);
        buttonUpdate = itemView.findViewById(R.id.buttonUpdate);
    }
}
public class ViewDialogUpdate {
    public void showDialog(Context context, String id, String name, String email, String
country) {
        final Dialog dialog = new Dialog(context);
        dialog.requestWindowFeature(Window.FEATURE_NO_TITLE);
        dialog.setCancelable(false);
        dialog setContentView(R.layout.alert_dialog_add_new_user);
        EditText textName = dialog.findViewById(R.id.textName);
        EditText textEmail = dialog.findViewById(R.id.textEmail);
        EditText textCountry = dialog.findViewById(R.id.textCountry);
        textName.setText(name);
    }
}

```

```

textEmail.setText(email);
textCountry.setText(country);
Button buttonUpdate = dialog.findViewById(R.id.buttonAdd);
Button buttonCancel = dialog.findViewById(R.id.buttonCancel);
buttonUpdate.setText("UPDATE");
buttonCancel.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        dialog.dismiss();
    }
});
buttonUpdate.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        String newName = textName.getText().toString();
        String newEmail = textEmail.getText().toString();
        String newCountry = textCountry.getText().toString();
        if (name.isEmpty() || email.isEmpty() || country.isEmpty()) {
            Toast.makeText(context, "Please Enter All data...",
Toast.LENGTH_SHORT).show();
        } else {
            if (newName.equals(name) && newEmail.equals(email) &&
newCountry.equals(country)) {
                Toast.makeText(context, "you don't change anything",
Toast.LENGTH_SHORT).show();
            } else {
                databaseReference.child("USERS").child(id).setValue(new UsersItem(id,
newName, newEmail, newCountry));
                Toast.makeText(context, "User Updated successfully!",
Toast.LENGTH_SHORT).show();
                dialog.dismiss();
            }
        }
    }
});

dialog.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT));
dialog.show();

}
}

public class ViewDialogConfirmDelete {
    public void showDialog(Context context, String id) {
        final Dialog dialog = new Dialog(context);
        dialog.requestWindowFeature(Window.FEATURE_NO_TITLE);

```

```
        dialog.setCancelable(false);
        dialog.setContentView(R.layout.view_dialog_confirm_delete);
        Button buttonDelete = dialog.findViewById(R.id.buttonDelete);
        Button buttonCancel = dialog.findViewById(R.id.buttonCancel);
        buttonCancel.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                dialog.dismiss();
            }
        });
        buttonDelete.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                databaseReference.child("USERS").child(id).removeValue();
                Toast.makeText(context, "User Deleted successfully!",
Toast.LENGTH_SHORT).show();
                dialog.dismiss();
            }
        });
        dialog.getWindow().setBackgroundDrawable(new
ColorDrawable(Color.TRANSPARENT));
        dialog.show();
    }
}
```

- **Output (Creation)**

8:03 PM | 0.0KB/s

Atul patil

atulpatil49@gmail.com

india

ADD

EXIT

ADD NEW USER

8:03 PM | 0.9KB/s

Name : Atul patil

Email : atulpatil49@gmail.com

Country : india

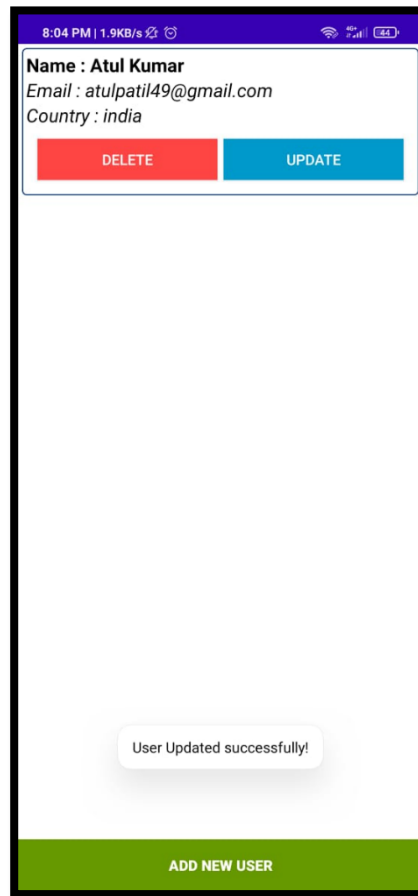
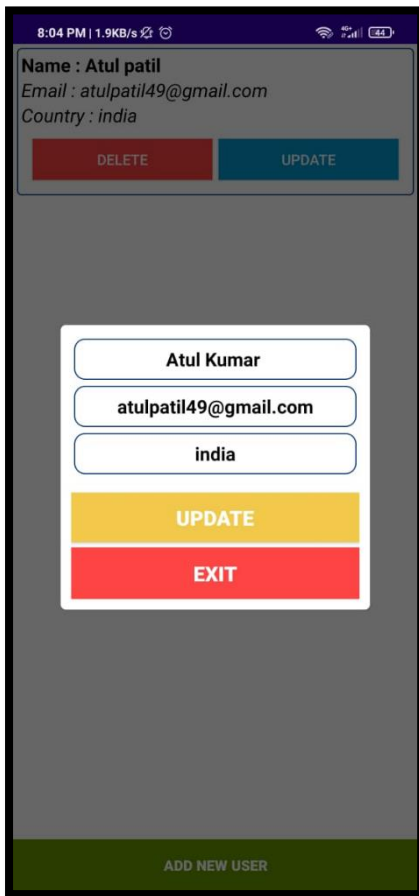
DELETE UPDATE

DONE!

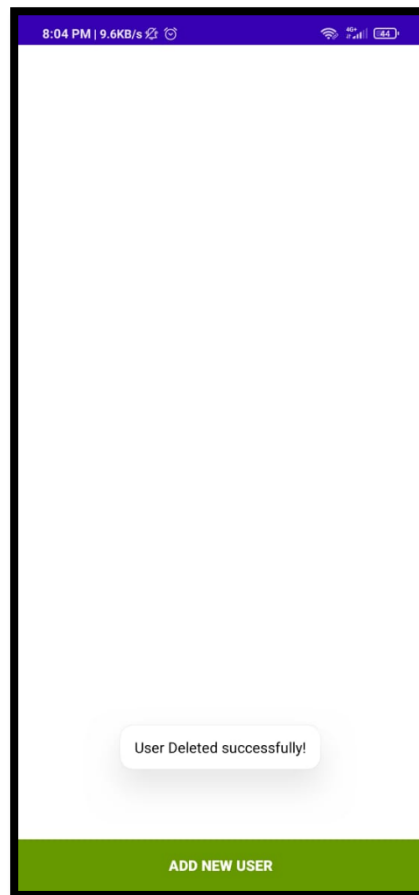
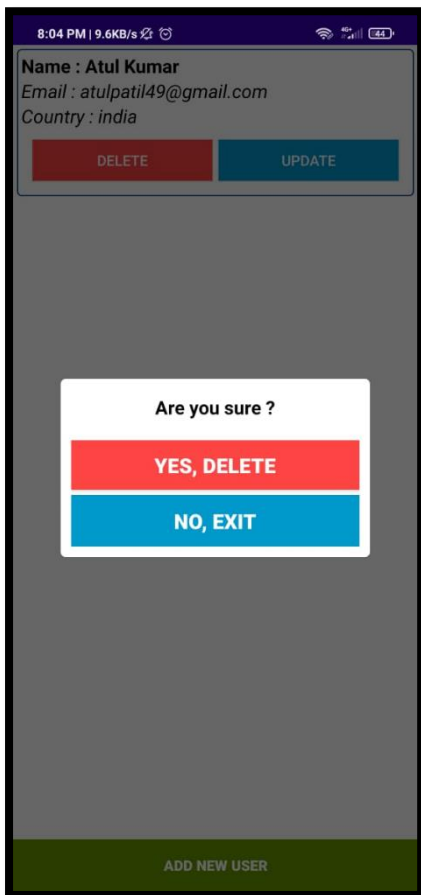
ADD NEW USER



- **Output (Updation)**



- **Output (Deletion)**



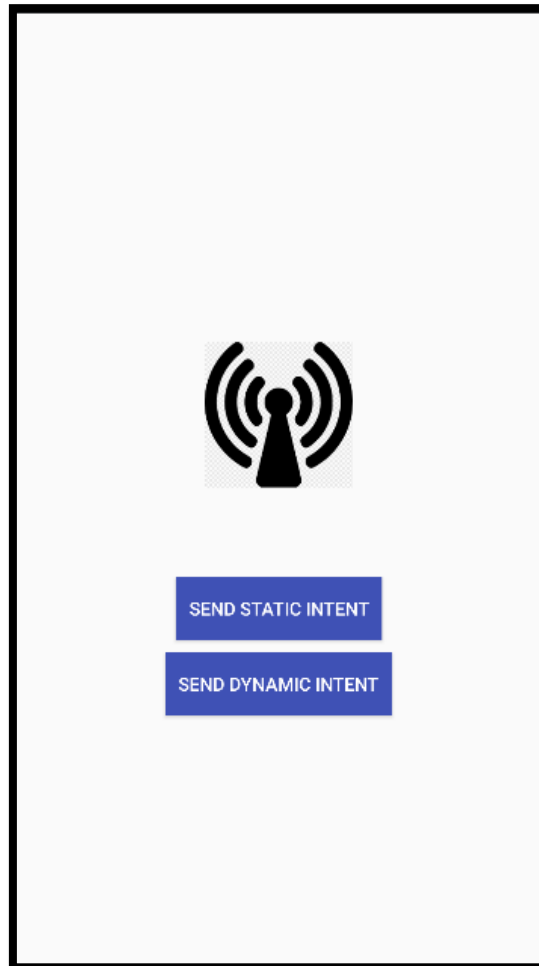
<https://practical9-1f5b9-default-rtdb.firebaseio.com>

`https://practical9-1f5b9-default-rtdb.firebaseio.com/:null`

PRACTICAL 12

AIM: Implement BroadcastReceiver.

- **activity_main.xml**



Layout: Linear Layout (Orientation: "vertical")

Widgets:

ImageView (Src: @drawable/broadcast1)

Button (Id: @+id/button1, Text: SEND STATIC INTENT)

Button (Id: @+id/Button2, Text: SEND DYNAMIC INTENT)

- **MainActivity.java**

```
package com.example.pr_12;
import android.content.Intent;
import android.content.IntentFilter;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.localbroadcastmanager.content.LocalBroadcastManager;
```

```
import android.util.Log;
import android.view.View;
```

```
public class MainActivity extends AppCompatActivity {

    public static final String ACTION1 = "edu.cs4730.bcr.mystaticevent";
    public static final String ACTION2 = "edu.cs4730.bcr.mydyanicevent";

    String TAG = "MainActivity";
    MyReceiver mReceiver;

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

        findViewById(R.id.button1).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent i = new Intent(MainActivity.ACTION1);
                i.setPackage("edu.cs4730.broadcastdemo1");
                sendBroadcast(i);
            }
        });

        findViewById(R.id.button2).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent i = new Intent(MainActivity.ACTION2);
                LocalBroadcastManager.getInstance(getApplicationContext()).sendBroadcast(i);
                Log.v(TAG, "Should have sent the broadcast.");
            }
        });
    }

    @Override
    public void onResume() {
        super.onResume();
        LocalBroadcastManager.getInstance(this).registerReceiver(mReceiver, new
IntentFilter(ACTION2));
    }

    @Override
    protected void onPause() {
        LocalBroadcastManager.getInstance(this).unregisterReceiver(mReceiver);
        Log.v(TAG, "receiver should be unregistered");
        super.onPause();
    }
}
```

```

    }
}

```

- **MyReceiver.java**

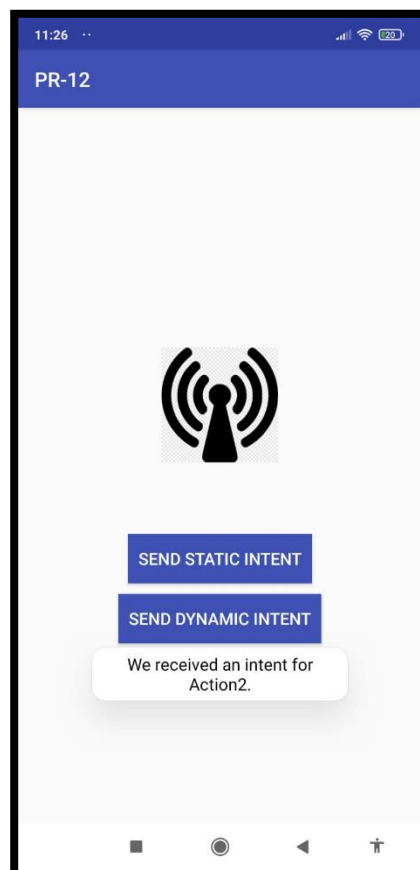
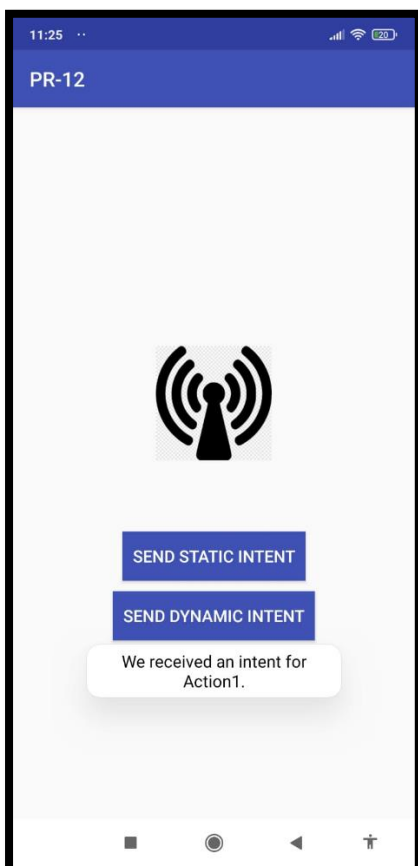
```
package com.example.pr_12;
```

```
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.widget.Toast;
```

```
public class MyReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        Toast.makeText(context, "Received an intent.", Toast.LENGTH_SHORT).show();
        if (intent.getAction().equals(MainActivity.ACTION1)) { //is it our action1?
            Toast.makeText(context, "We received an intent for Action1.",
                Toast.LENGTH_SHORT).show();
        } else if (intent.getAction().equals(MainActivity.ACTION2)) { //is it our action2?
            Toast.makeText(context, "We received an intent for Action2.",
                Toast.LENGTH_SHORT).show();
        }
    }
}

```

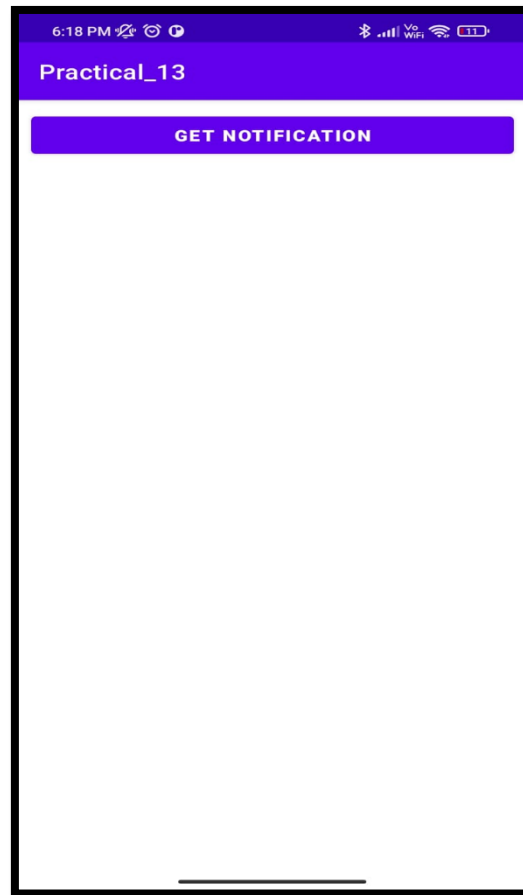
- **Output**



PRACTICAL 13

AIM: Implement Notifications system.

- **activity_main.xml**



Layout: Linear Layout (Orientation: "vertical")

Widgets:

Button (Id: "@+id/btn", text: "GET NOTIFICATION")

- **Manifest.xml**

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

- **MainActivity.java**

```
package com.example.practical_13;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.NotificationCompat;  
import androidx.core.app.NotificationManagerCompat;  
import androidx.core.content.res.ResourcesCompat;  
import android.app.AlertDialog;
```

```

import android.app.Notification;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.DialogInterface;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.drawable.BitmapDrawable;
import android.graphics.drawable.Drawable;
import android.os.Build;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    private static final String CHANNEL_ID="Message Channel";
    private final int NOTIFICATION_ID=1;
    Button notify;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button notify = findViewById(R.id.btn);
        Drawable drawable =
ResourcesCompat.getDrawable(getResources(),R.drawable.nature,null);
        BitmapDrawable bitmapDrawable=(BitmapDrawable)drawable;
        Bitmap largeIcon = bitmapDrawable.getBitmap();
        NotificationManager nm= (NotificationManager)
getSystemService(NOTIFICATION_SERVICE);

        notify.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Notification notification;
                if (android.os.Build.VERSION.SDK_INT >=
android.os.Build.VERSION_CODES.O) {
                    notification = new Notification.Builder(getApplicationContext())
                        .setLargeIcon(largeIcon)
                        .setSmallIcon(R.drawable.ic_baseline_notifications_24)
                        .setContentText("Message")
                        .setSubText("New Message from Atul")
                        .setChannelId(CHANNEL_ID)
                        .build();
                    nm.createNotificationChannel(new NotificationChannel(CHANNEL_ID,"New
Channle",NotificationManager.IMPORTANCE_HIGH));
                }
                else
                {
                    notification = new Notification.Builder(getApplicationContext())

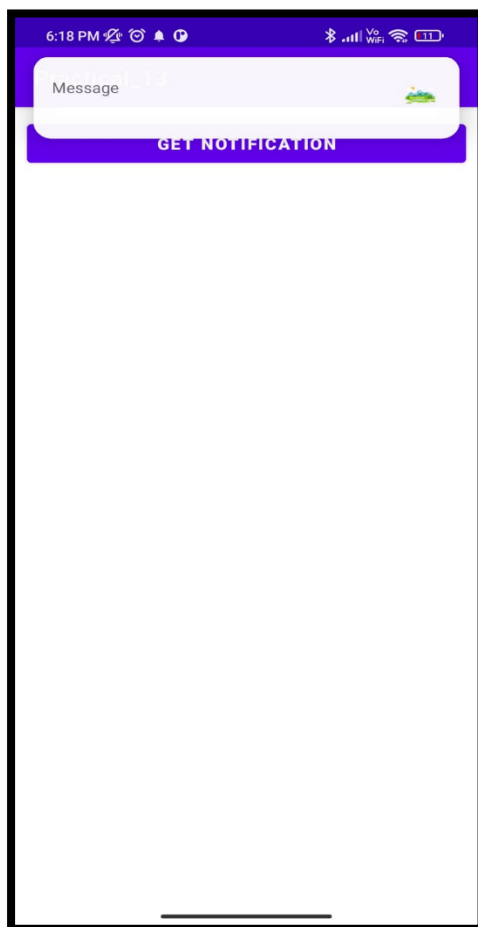
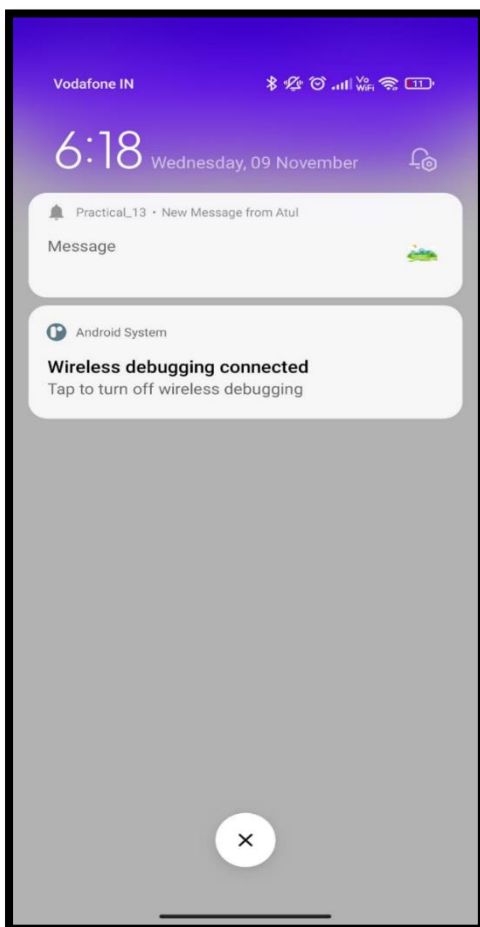
```

```

        .setLargeIcon(largeIcon)
        .setSmallIcon(R.drawable.ic_baseline_notifications_24)
        .setContentText("New Message")
        .setSubText("New Message from Atul")
        .build();
    }
    nm.notify(NOTIFICATION_ID,notification);
}
});
}
}
}

```

- **Output**

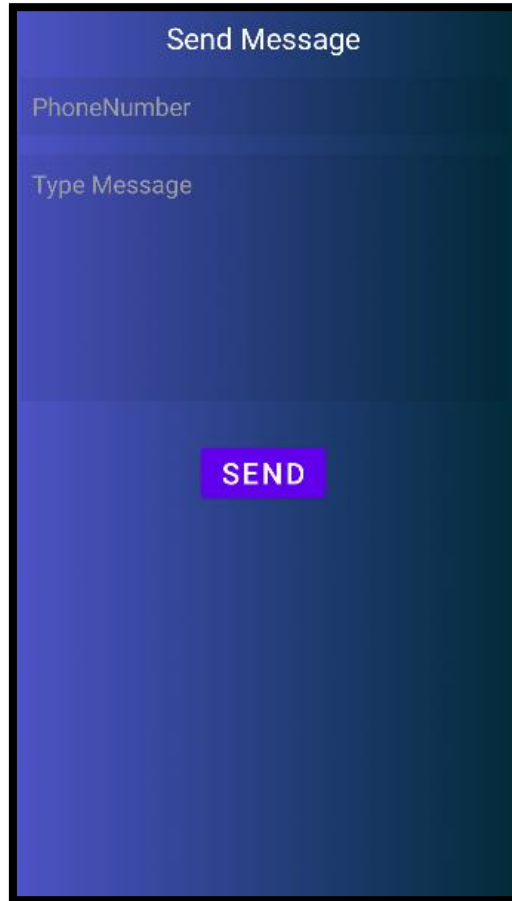


PRACTICAL 15

AIM: make a use of android system.

a) Simple SMS App

- **activity_send_message.xml**



Layout: Linear Layout (Orientation: "vertical", background="@drawable/gradient")

Widgets:

```
TextView(text="Send Message")
EditText(id="@+id/phone", hint="PhoneNumber")
EditText(id="@+id/ phone_message", hint=" Type Message")
Button(id="@+id/send_btn", text="Send")
```

- **Send_message.java**

```
package com.example.pr_15;

import android.Manifest;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.util.Log;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

public class Send_message extends AppCompatActivity {
    private EditText message, phone;
    private Button send_sms;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_send_message);
        message = findViewById(R.id.phone_message);
        phone = findViewById(R.id.phone);
        send_sms = findViewById(R.id.send_btn);
        send_sms.setOnClickListener(v -> {
            if (ContextCompat.checkSelfPermission(Send_message.this,
Manifest.permission.SEND_SMS) == PackageManager.PERMISSION_GRANTED) {
                sendMessage();
            } else {
                ActivityCompat.requestPermissions(Send_message.this, new
String[]{Manifest.permission.SEND_SMS}, 100);
            }
        });
    }

    private void sendMessage() {
        String sms_text = message.getText().toString().trim();
        String phone_no = phone.getText().toString().trim();
        if (sms_text.isEmpty() || phone_no.isEmpty()) {
            Toast.makeText(Send_message.this, "All Fields required!",
Toast.LENGTH_LONG).show();
        } else {
            try {
                SmsManager smsManager = SmsManager.getDefault();
                smsManager.sendTextMessage(phone_no, null, sms_text, null, null);
                Toast.makeText(Send_message.this, "Sent success!",
Toast.LENGTH_LONG).show();
            }
        }
    }
}
```



```

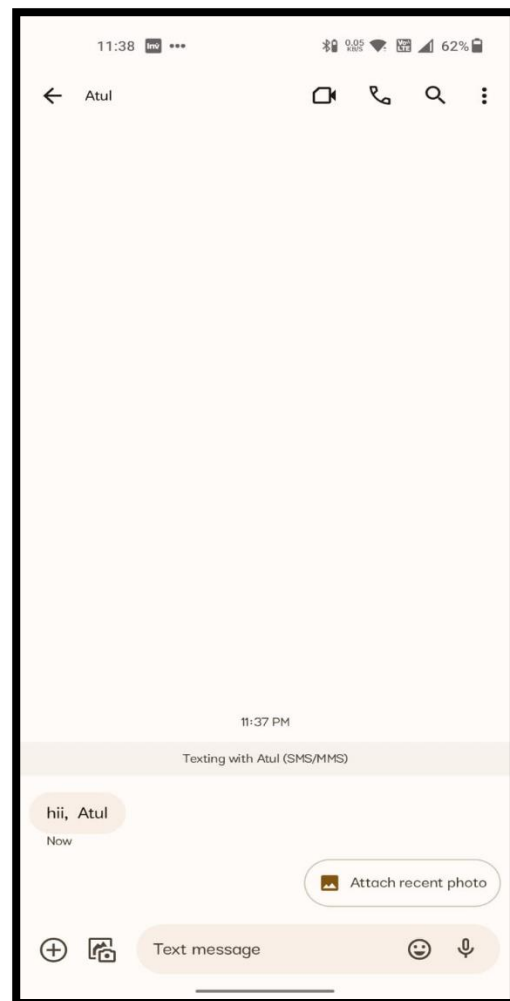
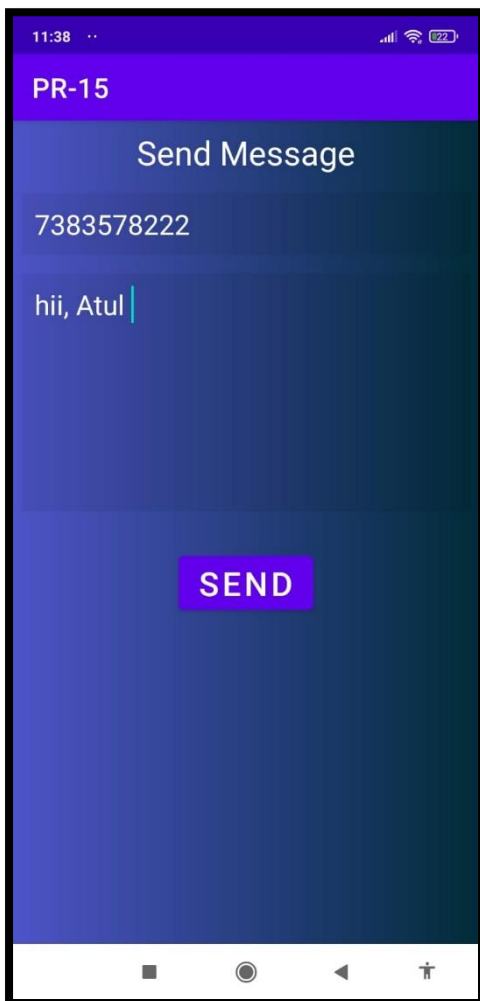
    } catch (Exception e) {
        Toast.makeText(Send_message.this, "Error!", Toast.LENGTH_LONG).show();
        Log.d("SmsError", e.toString());
    }
}
}
}

```

- **AndroidManifest.xml(add permission)**

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

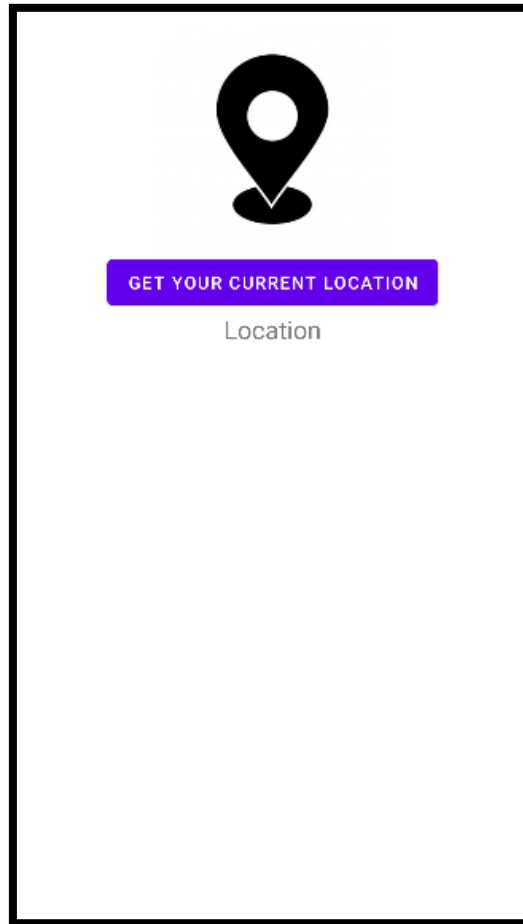
- **Output**



PRACTICAL 16

AIM: Using location service get the current location and display in TextView.

- **activity_main.xml**



Layout: Linear Layout (Orientation: "vertical", background="@drawable/gradient")

Widgets:

```
ImageView(src="@drawable/location1")
Button(id="@+id/Button_Location", text=" Get Your Current Location")
TextView(id="@+id/Current_Location", text="Location")
```

- **MainActivity.java**

```
package com.example.pr_16;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

import android.Manifest;
import android.content.pm.PackageManager;
```

```

import android.location.Address;
import android.location.Geocoder;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import java.util.List;
import java.util.Locale;

public class MainActivity extends AppCompatActivity implements LocationListener {
    Button locationButton;
    TextView locationText;
    LocationManager locationManager;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        locationButton = findViewById(R.id.Button_Location);
        locationText = findViewById(R.id.Current_Location);
        locationButton.setOnClickListener(view -> getLocation());
    }

    private void getLocation() {
        try {
            locationManager = (LocationManager)
                getApplicationContext().getSystemService(LOCATION_SERVICE);
            if (ActivityCompat.checkSelfPermission(this,
                Manifest.permission.ACCESS_FINE_LOCATION) !=
                PackageManager.PERMISSION_GRANTED &&
                ActivityCompat.checkSelfPermission(this,
                Manifest.permission.ACCESS_COARSE_LOCATION) !=
                PackageManager.PERMISSION_GRANTED) {
                ActivityCompat.requestPermissions(MainActivity.this, new
                    String[]{Manifest.permission.ACCESS_FINE_LOCATION}, 100);
            }
            locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 5000,
                5, MainActivity.this);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    @Override
    public void onLocationChanged(@NonNull Location location) {

```

```

        Toast.makeText(this, "Latitude: " + location.getLatitude() + ", Longitude: " +
location.getLongitude(), Toast.LENGTH_SHORT).show();
        try {
            Geocoder geocoder = new Geocoder(MainActivity.this, Locale.getDefault());
            List<Address> addresses = geocoder.getFromLocation(location.getLatitude(),
                location.getLongitude(), 1);
            String address = addresses.get(0).getAddressLine(0);
            locationText.setText("Longitude : " + addresses.get(0).getLongitude() + "\n" +
"Latitude : " + addresses.get(0).getLatitude());

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

    }

    @Override
    public void onLocationChanged(@NonNull List<Location> locations) {
        LocationListener.super.onLocationChanged(locations);
    }

    @Override
    public void onFlushComplete(int requestCode) {
        LocationListener.super.onFlushComplete(requestCode);
    }

    @Override
    public void onStatusChanged(String provider, int status, Bundle extras) {
        LocationListener.super.onStatusChanged(provider, status, extras);
    }

    @Override
    public void onProviderEnabled(@NonNull String provider) {
        LocationListener.super.onProviderEnabled(provider);
    }

    @Override
    public void onProviderDisabled(@NonNull String provider) {
        LocationListener.super.onProviderDisabled(provider);
    }

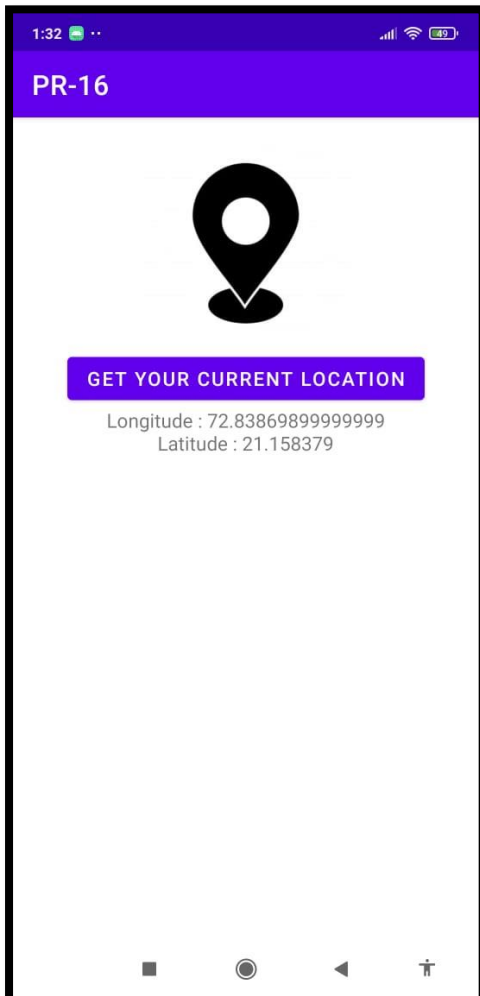
    @Override
    public void onPointerCaptureChanged(boolean hasCapture) {
        super.onPointerCaptureChanged(hasCapture);
    }
}

```

- **AndroidManifest.xml(add below permissions)**

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

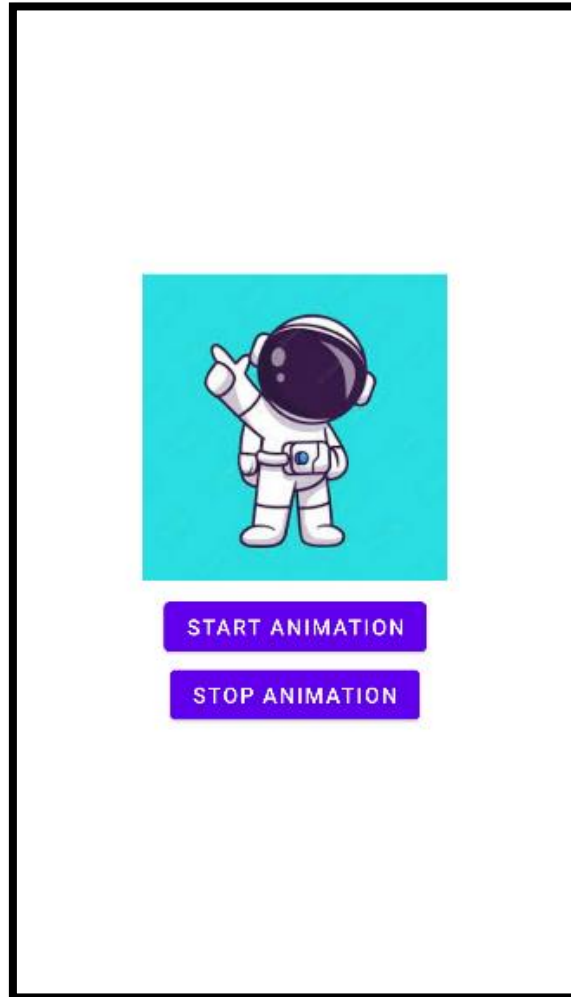
- **Output**



PRACTICAL 17

AIM: Display the use of animations.

- **activity_main.xml**



Layout: Linear Layout (Orientation: "vertical")

Widgets:

```
ImageView(src="@drawable/ cartoon", id="@+id/imageview")
Button(id="@+id/ BtnRotate", text="START ANIMATION")
Button(id="@+id/ BtnStop ", text="STOP ANIMATION")
```

- **rotate_animation.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android" >
    <rotate
        android:duration="6000"
        android:fromDegrees="0"
        android:pivotX="50%"
        android:pivotY="50%"
        android:toDegrees="360" />

    <rotate
        android:duration="6000"
        android:fromDegrees="360"
        android:pivotX="50%"
        android:pivotY="50%"
        android:startOffset="5000"
        android:toDegrees="0" />
</set>
```

- **MainActivity.java**

```
package com.example.pr_17;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;
public class MainActivity extends AppCompatActivity {
    ImageView imageView;
    Button zoomBTN, stopBTN;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        imageView = findViewById(R.id.imageview);
        zoomBTN = findViewById(R.id.BtnRotate);
        stopBTN = findViewById(R.id.BtnStop);

        zoomBTN.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // To add zoom animation
                Animation animation = AnimationUtils.loadAnimation(getApplicationContext(),
R.anim.rotate_animation);
                imageView.startAnimation(animation);
            }
        });
    }
}
```

```

stopBTN.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        // To stop the animation going on imageview
        imageView.clearAnimation();
    }
});
}
}

```

- **Output**

