MAD Practical Exam Question Bank

1. Write a program to demonstrate the use of Linear Layout and and Absolute Layout

ANS:

Activity_main.xml:-

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="150dp"
        android:layout_y="50dp"
        android:text="Hello World" />

    <LinearLayout
        android:layout_x="30dp"
        android:layout_y="100dp"
        android:orientation="horizontal"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="BUTTON1"/>
        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="BUTTON2"/>
        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="BUTTON3"/>
    </LinearLayout>
</AbsoluteLayout>
```

Output:-

2. Develop a program to implement table layout and relative layout

ANS:

Activity_main:-

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

    <TextView
        android:id="@+id/txt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="DATA TABLE: " />

    <TableLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/txt"
        >
        <TableRow>
            <TextView
                android:text="DATAHEAD 1" android:padding="8dp"/>
            <TextView
```

```
                android:text="DATAHEAD 2" android:padding="8dp"/>
            <TextView
                android:text="DATAHEAD 3" android:padding="8dp"/>
        </TableRow>
        <TableRow>
            <TextView android:text="DATA 1" android:padding="8dp"/>
            <TextView android:text="DATA 2" android:padding="8dp"/>
            <TextView android:text="DATA 3" android:padding="8dp"/>
        </TableRow>
        <TableRow>
            <TextView android:text="DATA 4" android:padding="8dp"/>
            <TextView android:text="DATA 5" android:padding="8dp"/>
            <TextView android:text="DATA 6" android:padding="8dp"/>
        </TableRow>
    </TableLayout>
</RelativeLayout>
```

Output:-

3. Write a program to demonstrate the use of Autocomplete Textview

ANS:

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:gravity="center"
    android:orientation="vertical">
    <TextView

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="ENTER NAME OF FRUITS (APPLE,ORANGE,MANGO,BANANA,GRAPES)"/>

    <AutoCompleteTextView
        android:id="@+id/t1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="ENTER NAME">
        <requestFocus/>
    </AutoCompleteTextView>
</LinearLayout>
```

MainActivity.java:-

```
package com.example.demo2;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.*;

public class MainActivity extends AppCompatActivity {
    AutoCompleteTextView t1;
    String fruits[] = {"APPLE","ORANGE","MANGO","BANANA","GRAPES"};
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        ArrayAdapter<String> adapter=new
ArrayAdapter<>(this,android.R.layout.select_dialog_item,fruits);
        t1=(AutoCompleteTextView)findViewById(R.id.t1);
        t1.setThreshold(1);
        t1.setAdapter(adapter);
    }
}
```

Output:-

4. Write a program to demonstrate the use of Radio Button ?

ANS:

Activity_main.xml:-

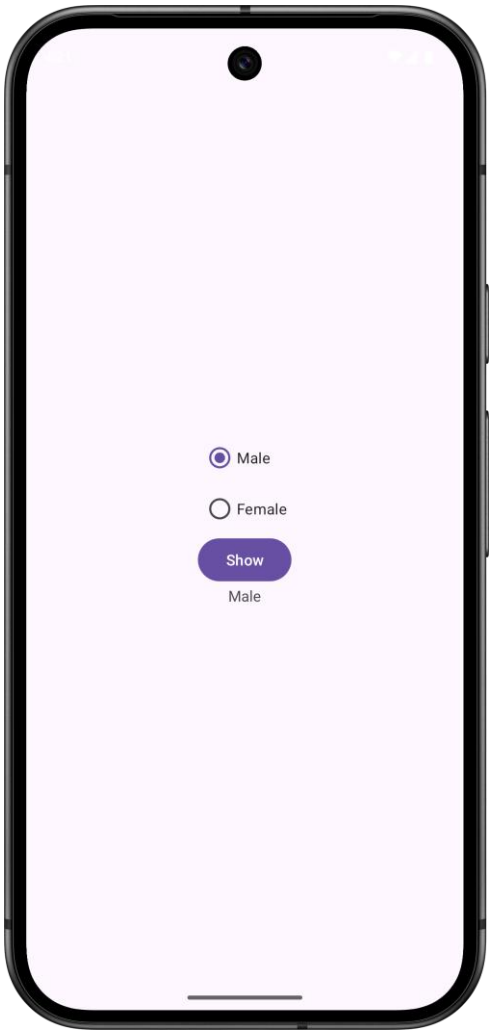
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical">

    <RadioGroup
        android:id="@+id/radioGroup"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">

        <RadioButton
            android:id="@+id/radioMale"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Male" />
```

```
        <RadioButton
            android:id="@+id/radioFemale"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Female" />
    </RadioGroup>

    <Button
        android:id="@+id/btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Show" />

    <TextView
        android:id="@+id/output"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>
```

MainActivity.java:-

```
package com.example.demo2;

import android.os.Bundle;
import android.view.View;
import android.widget.*;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        RadioGroup rg = findViewById(R.id.radioGroup);
        Button btn = findViewById(R.id.btn);
        TextView out = findViewById(R.id.output);

        btn.setOnClickListener(v -> {
            int id = rg.getCheckedRadioButtonId();
            if (id != -1)
                out.setText(((RadioButton) findViewById(id)).getText());
            else
                out.setText("Nothing selected");
        });
    }
}
```

Output:-



5. Write a program to demonstrate the use of List View ?

ANS:

Activity_main.xml:-

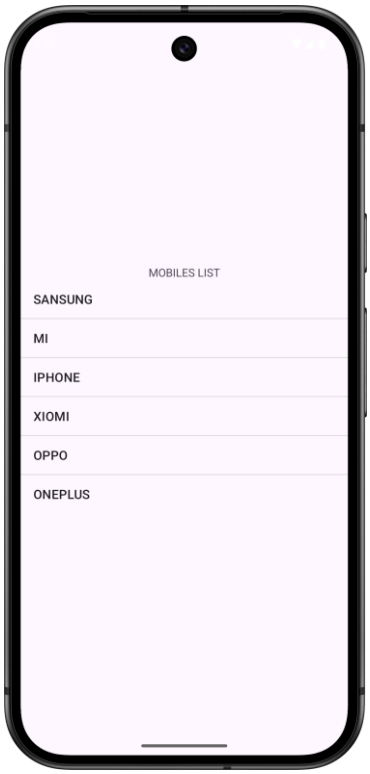
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="MOBILES LIST"/>
    <ListView
        android:id="@+id/l1"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
</LinearLayout>
```

MainActivity.java:-

```
package com.example.demo2;

import android.os.Bundle;
import android.view.View;
import android.widget.*;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    ListView l1;
    String[] mob={"SANSUNG","MI","IPHONE","XIOMI","OPPO","ONEPLUS"};
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        l1 = findViewById(R.id.l1);
        ArrayAdapter<String> ad=new ArrayAdapter<>(this,
android.R.layout.simple_list_item_1,mob);
        l1.setAdapter(ad);
    }
}
```

Output:-



6. Write a program to create a of Custom Toast ?

ANS:

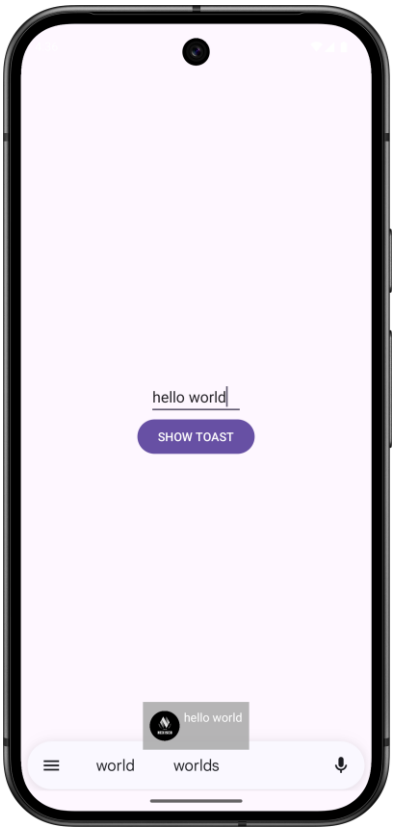
Activity_main.xml:-

```
<?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:gravity="center"
    android:orientation="vertical">

    <EditText
        android:id="@+id/e1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="ENTER TEXT"/>
    <Button
        android:id="@+id/b1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="SHOW TOAST"/>
</LinearLayout>
```

MainActivity.java:-

```
package com.example.demo2;

import android.os.Bundle;
import android.view.*;
import android.widget.*;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    EditText e1;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        e1 = findViewById(R.id.e1);
        findViewById(R.id.b1).setOnClickListener(v->{
            LayoutInflater inflater=getLayoutInflater();
            View
layout=inflater.inflate(R.layout.custom_toast,findViewById(R.id.root));
            TextView t1=layout.findViewById(R.id.t1);

            t1.setText(e1.getText().toString());
            Toast t=new Toast(getApplicationContext());
            t.setDuration(Toast.LENGTH_SHORT);
            t.setView(layout);
            t.show();
        });
    }
}
```

Output:-



7. Write a program to demonstrate the use of DatePicker

ANS:

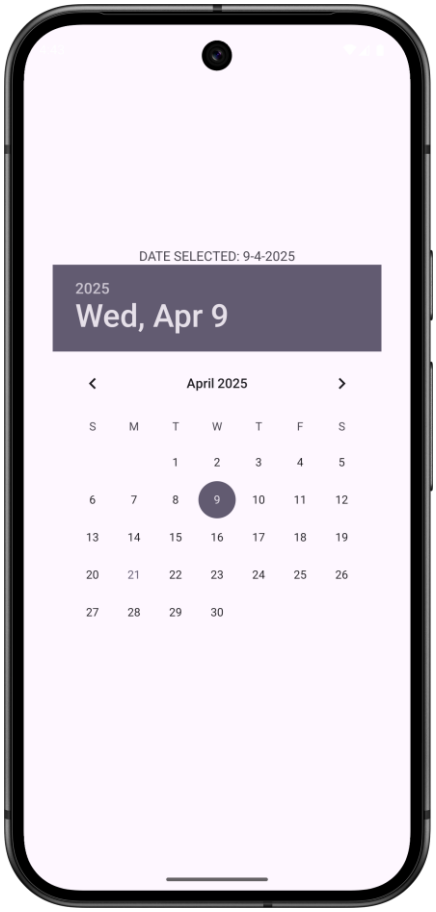
Activity_main.xml:-

```
<?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:gravity="center"
    android:orientation="vertical">

    <TextView
        android:id="@+id/t1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="DATE SELECTED: "/>
    <DatePicker
        android:id="@+id/d1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
</LinearLayout>
```

MainActivity.java:-

```java
package com.example.demo2;

import android.os.Bundle;
import android.view.*;
import android.widget.*;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
    DatePicker d1;
    TextView t1;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        d1 = findViewById(R.id.d1);
        t1 = findViewById(R.id.t1);
        d1.init(d1.getYear(), d1.getMonth(), d1.getDayOfMonth(),(view, year,
monthOfYear, dayOfMonth) -> {
            t1.setText("DATE SELECTED: " + dayOfMonth + "-" + (monthOfYear + 1) +
"-" + year);
        });
    }
}
```

Output:-



8. Android Program to enable and disable WiFi Connection

ANS:

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:gravity="center">

    <ToggleButton
        android:id="@+id/t1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textOn="WiFi ON"
        android:textOff="WiFi OFF" />
</LinearLayout>
```

MainActivity.java:-

```
package com.example.demo2;

import android.net.wifi.WifiManager;
import android.os.Bundle;
import android.widget.ToggleButton;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    WifiManager wifiManager;

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

        ToggleButton t = findViewById(R.id.t1);
        wifiManager = (WifiManager)
getApplicationContext().getSystemService(WIFI_SERVICE);

        // Set initial state
        t.setChecked(wifiManager.isWifiEnabled());

        t.setOnCheckedChangeListener((buttonView, isChecked) -> {
            wifiManager.setWifiEnabled(isChecked);
        });
    }
}
```

AndroidManifest.xml:-

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

Output:-

9. Develop a program to implement explicit intent and implicit intent

ANS:

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:gravity="center">
```

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

    <Button
        android:id="@+id/btnImplicit"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Implicit Intent"/>
</LinearLayout>
```

MainActivity.java:-

```
package com.example.demo2;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);

        findViewById(R.id.btnExplicit).setOnClickListener(v -> {
            Intent i = new Intent(this, SecondActivity.class);
            startActivity(i);
        });

        findViewById(R.id.btnImplicit).setOnClickListener(v -> {
            Intent i = new Intent(Intent.ACTION_VIEW,
Uri.parse("https://google.com"));
            startActivity(i);
        });
    }
}
```

Activity_second.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:gravity="center"
    android:orientation="vertical">

    <TextView
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="This is Second Activity"
        android:textSize="20sp"/>
</LinearLayout>
```

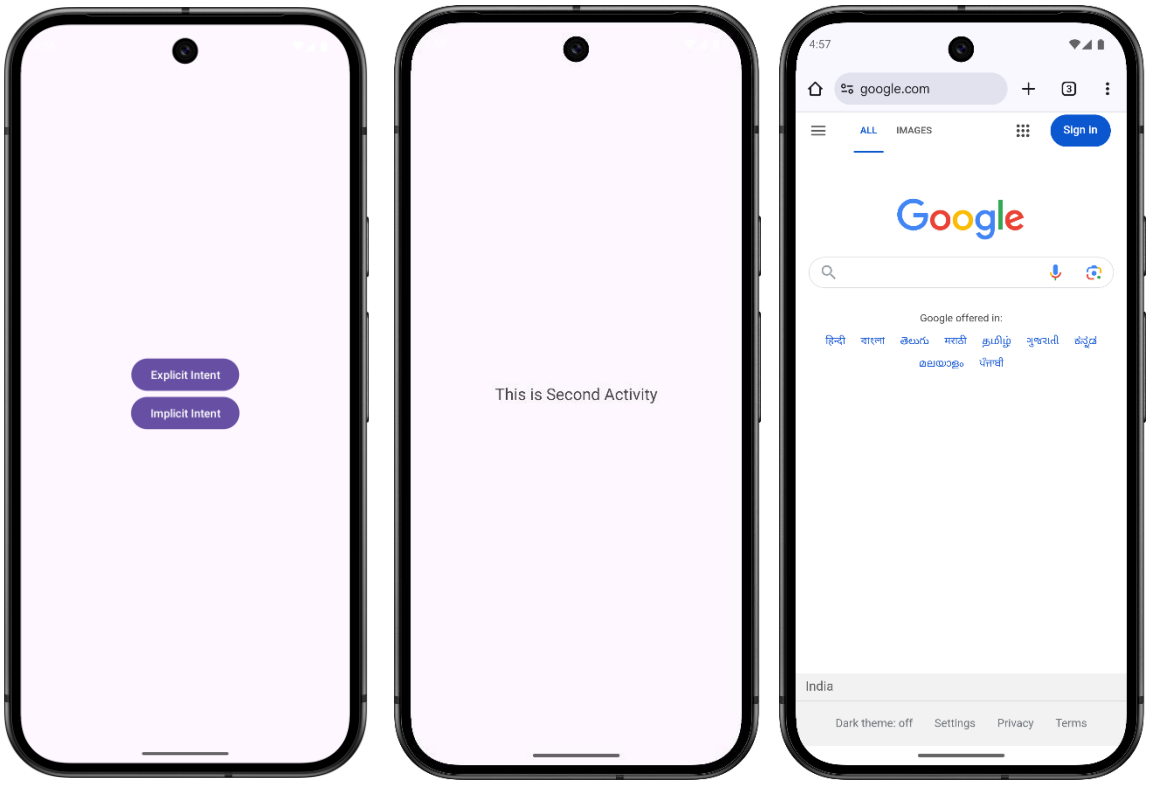
SecondActivity.java:-

```
package com.example.demo2;

import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;

public class SecondActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_second);
    }
}
```

AndroidManifest.xml:-

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" >
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Demo2"
        tools:targetApi="31" >
        <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>
        <activity android:name=".SecondActivity" />
    </application>

</manifest>
```

Output:-



10. Write a program to send SMS

ANS:

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:gravity="center">

    <EditText
        android:id="@+id/phone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Phone" />

    <EditText
        android:id="@+id/msg"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Message" />

    <Button
        android:id="@+id/send"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Send SMS" />
</LinearLayout>
```

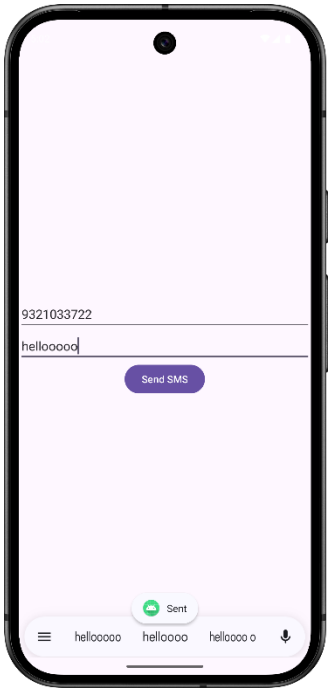
MainActivity.java:-

```
package com.example.demo2;

import android.os.Bundle;
import android.telephony.*;
import android.widget.*;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);

        EditText phone = findViewById(R.id.phone), msg = findViewById(R.id.msg);
        findViewById(R.id.send).setOnClickListener(v -> {
            try {
                SmsManager.getDefault().sendTextMessage(phone.getText().toString(),
null, msg.getText().toString(), null, null);
                Toast.makeText(this, "Sent", Toast.LENGTH_SHORT).show();
            } catch (Exception e) {
                Toast.makeText(this, "Fail", Toast.LENGTH_SHORT).show();
            }
        });

    }
}
```

ndroidManifest.xml:-

```
<uses-feature
    android:name="android.hardware.telephony"
    android:required="false" />

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

Output:-



11. Android to display all available sensors in a mobile device.

ANS:

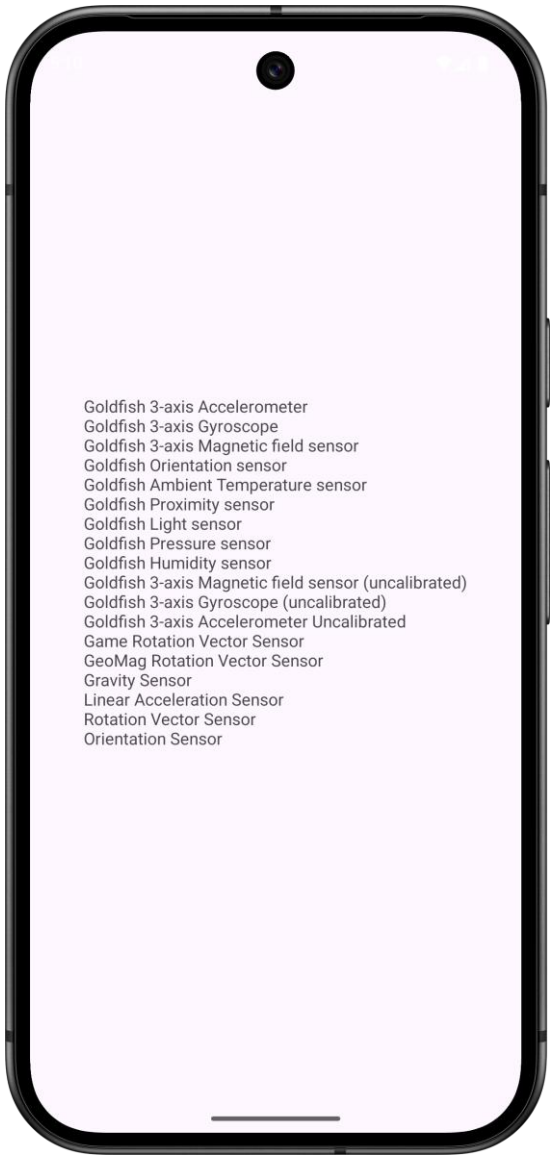
Activity_main.xml:-

```
<?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:gravity="center">

    <TextView
        android:id="@+id/t1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
</LinearLayout>
```

MainActivity.java:-

```
 package com.example.demo2;

import android.content.Context;
import android.os.Bundle;
import android.widget.*;
import android.hardware.*;
import java.util.List;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    SensorManager mgr;
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);

        TextView t1=findViewById(R.id.t1);
        mgr=(SensorManager)getSystemService(Context.SENSOR_SERVICE);
        List<Sensor> sensors=mgr.getSensorList(Sensor.TYPE_ALL);
        StringBuilder str=new StringBuilder();
        for(Sensor s:sensors){
            str.append(s.getName()+"\n");
        }
        t1.setText(str);
    }
}
```

Output:-



12. Write a Bluetooth Android program to Turn ON Bluetooth, Turn OFF Bluetooth, make device visible and display the list of all available Bluetooth devices.

ANS:

```
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical">
```

```
    <Button
        android:id="@+id/b1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Turn On" />

    <Button
        android:id="@+id/b2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Get visible" />

    <Button
        android:id="@+id/b3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="List devices" />

    <Button
        android:id="@+id/b4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="turn off" />

    <ListView
        android:id="@+id/l1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

</LinearLayout>
```

MainActivity.java :-

```
package com.example.demo1;

import android.bluetooth.*;
import android.content.Intent;
import android.os.Bundle;
import android.widget.*;
import java.util.*;

import androidx.appcompat.app.AppCompatActivity;
```

```
public class MainActivity extends AppCompatActivity {
    BluetoothAdapter b;
    ListView l1;

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

        // Get Bluetooth adapter and device list view
        b = BluetoothAdapter.getDefaultAdapter();
        l1 = findViewById(R.id.l1);

        // Set up button click listeners
        findViewById(R.id.b1).setOnClickListener(v -> {
            if (b.isEnabled()) {
                showToast("Already on");
            } else {
                startActivityForResult(new
Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE), 0);
                showToast("Turned on");
            }
        });

        findViewById(R.id.b2).setOnClickListener(v -> {
            startActivityForResult(new
Intent(BluetoothAdapter.ACTION_REQUEST_DISCOVERABLE), 0);
        });

        findViewById(R.id.b3).setOnClickListener(v -> {
            Set<BluetoothDevice> paired = b.getBondedDevices();
            ArrayList<String> dl = new ArrayList<>();

            for (BluetoothDevice d : paired) {
                dl.add(d.getName());
            }

            l1.setAdapter(new ArrayAdapter<>(this,
                    android.R.layout.simple_list_item_1, dl));
            showToast("Showing Paired Devices");
        });

        findViewById(R.id.b4).setOnClickListener(v -> {
```

```
            b.disable();
            showToast("Turned off");
        });
    }

    private void showToast(String msg) {
        Toast.makeText(this, msg, Toast.LENGTH_SHORT).show();
    }
}
```

Androidmanifest.xml –

```
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
<uses-permission android:name="android.permission.BLUETOOTH_SCAN" />
<uses-permission android:name="android.permission.BLUETOOTH_CONNECT" />
<uses-permission android:name="android.permission.ACTION_REQUEST_ENABLE"/>
```