

Experiment 13

Develop an application to enable Bluetooth and Wi-fi through the radio button.

Layout.xml

```
1. <?xml version="1.0" encoding="utf-8"?>
2. <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://
   schemas.android.com/apk/res/android"
3.     xmlns:app="http://schemas.android.com/apk/res-auto"
4.     xmlns:tools="http://schemas.android.com/tools"
5.     android:layout_width="match_parent"
6.     android:layout_height="match_parent"
7.     tools:context=".MainActivity">
8.
9.     <Button
10.         android:id="@+id/btn_bluetooth"
11.         android:layout_width="wrap_content"
12.         android:layout_height="wrap_content"
13.         android:text="Bluetooth: Off"
14.         app:layout_constraintBaseline_toBaselineOf="@+id/btn_disc
   over"
15.         app:layout_constraintEnd_toStartOf="@+id/btn_discover"
16.         app:layout_constraintHorizontal_bias="0.5"
17.         app:layout_constraintStart_toStartOf="parent" />
18.
19.     <TextView
20.         android:id="@+id/textView"
21.         android:layout_width="wrap_content"
22.         android:layout_height="wrap_content"
23.         android:text="Click On the button to activate/de-
   activate bluetooth."
24.         app:layout_constraintBottom_toBottomOf="parent"
25.         app:layout_constraintEnd_toEndOf="parent"
26.         app:layout_constraintHorizontal_bias="0.493"
27.         app:layout_constraintStart_toStartOf="parent"
28.         app:layout_constraintTop_toBottomOf="@+id/app_name"
29.         app:layout_constraintVertical_bias="0.019" />
30.
31.     <Button
32.         android:id="@+id/btn_discover"
33.         android:layout_width="wrap_content"
34.         android:layout_height="wrap_content"
35.         android:layout_marginTop="16dp"
36.         android:text="Scan"
37.         app:layout_constraintEnd_toStartOf="@+id/btn_wifi"
38.         app:layout_constraintHorizontal_bias="0.5"
39.         app:layout_constraintStart_toEndOf="@+id/btn_bluetooth"
40.         app:layout_constraintTop_toBottomOf="@+id/textView" />
41.
42.     <ListView
```

Experiment 13

Develop an application to enable Bluetooth and Wi-fi through the radio button.

```
43.         android:id="@+id/lv_deviceList"
44.         android:layout_width="0dp"
45.         android:layout_height="0dp"
46.         android:layout_marginStart="16dp"
47.         android:layout_marginTop="16dp"
48.         android:layout_marginEnd="16dp"
49.         android:layout_marginBottom="16dp"
50.         app:layout_constraintBottom_toBottomOf="parent"
51.         app:layout_constraintEnd_toEndOf="parent"
52.         app:layout_constraintStart_toStartOf="parent"
53.         app:layout_constraintTop_toBottomOf="@+id/btn_discover" /
54.     >
55.     <Button
56.         android:id="@+id/btn_wifi"
57.         android:layout_width="wrap_content"
58.         android:layout_height="wrap_content"
59.         android:text="Enable Wifi"
60.         app:layout_constraintBaseline_toBaselineOf="@+id/btn_disc
over"
61.         app:layout_constraintEnd_toEndOf="parent"
62.         app:layout_constraintHorizontal_bias="0.5"
63.         app:layout_constraintStart_toEndOf="@+id/btn_discover" />
64.
65.     <TextView
66.         android:id="@+id/app_name"
67.         android:layout_width="wrap_content"
68.         android:layout_height="wrap_content"
69.         android:layout_marginTop="32dp"
70.         android:text="Bluetooth and Wifi Enabling Service."
71.         android:textAlignment="center"
72.         android:textSize="18sp"
73.         app:layout_constraintEnd_toEndOf="parent"
74.         app:layout_constraintStart_toStartOf="parent"
75.         app:layout_constraintTop_toTopOf="parent" />
76.
77. </androidx.constraintlayout.widget.ConstraintLayout>
```

Layout.java

```
1. package com.example.wifiblueetoothserviceapp;
2.
3. import androidx.annotation.Nullable;
4. import androidx.appcompat.app.AppCompatActivity;
5.
6. import android.bluetooth.BluetoothAdapter;
```

Experiment 13

Develop an application to enable Bluetooth and Wi-fi through the radio button.

```
7. import android.bluetooth.BluetoothDevice;
8. import android.content.BroadcastReceiver;
9. import android.content.Context;
10.     import android.content.Intent;
11.     import android.content.IntentFilter;
12.     import android.net.wifi.WifiManager;
13.     import android.os.Bundle;
14.
15.     import android.provider.Settings;
16.     import android.view.View;
17.     import android.widget.AdapterView;
18.     import android.widget.Button;
19.     import android.widget.CompoundButton;
20.     import android.widget.ListView;
21.     import android.widget.Switch;
22.     import android.widget.Toast;
23.
24.     import java.util.ArrayList;
25.
26.     //! very important note: make sure the app gets location permission.
27.
28.     public class MainActivity extends AppCompatActivity {
29.
30.         private final static int BLUETOOTH_REQ_CODE = 1; //must be >0
31.
32.         private BluetoothAdapter bluetoothAdapter;
33.         Button blue_btn, scan_btn, wifiBtn;
34.
35.         //discover bluetooth
36.         ListView scanList;
37.         ArrayList<String> scannedDeviceList;
38.         ArrayAdapter<String> adapter;
39.
40.         @Override
41.         protected void onCreate(Bundle savedInstanceState) {
42.             super.onCreate(savedInstanceState);
43.             setContentView(R.layout.activity_main);
44.
45.             //bluetooth
46.             bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
47.
48.             blue_btn = findViewById(R.id.btn_bluetooth);
49.             scan_btn = findViewById(R.id.btn_discover);
50.             scanList = findViewById(R.id.lv_deviceList);
51.             scannedDeviceList = new ArrayList();
52.
53.             //wifi
54.             wifiBtn = findViewById(R.id.btn_wifi);
```

Experiment 13

Develop an application to enable Bluetooth and Wi-fi through the radio button.

```
53.
54.         //bluetooth logic-----
55.         if (bluetoothAdapter.isEnabled()) {
56.             Toast.makeText(this, "Bluetooth Already Enabled!", To
ast.LENGTH_LONG).show();
57.             blue_btn.setText("Bluetooth: On");
58.         }
59.
60.
61.         blue_btn.setOnClickListener(new View.OnClickListener() {
62.
63.             @Override
64.             public void onClick(View v) {
65.                 //already Enabbled
66.                 if (bluetoothAdapter.isEnabled()) {
67.                     Toast.makeText(getApplicationContext(), "Blue
tooth Already Enabled!", Toast.LENGTH_LONG).show();
68.                     blue_btn.setText("Bluetooth: On");
69.                 } else
70.                     checkBluetoothPermissions();
71.             }
72.         });
73.
74.
75.         adapter = new ArrayAdapter(getApplicationContext(), andro
id.R.layout.simple_list_item_1, scannedDeviceList);
76.         scanList.setAdapter(adapter);
77.
78.         scan_btn.setOnClickListener(new View.OnClickListener() {
79.
80.             @Override
81.             public void onClick(View v) {
82.
83.                 // clearing list before every request
84.                 scannedDeviceList.clear();
85.
86.                 // If older discovery is in progress, cancels it
before starting the new one.
87.                 if (bluetoothAdapter.isDiscovering()) {
88.                     bluetoothAdapter.cancelDiscovery();
89.                 }
90.
91.                 checkBluetoothPermissions();
92.                 bluetoothAdapter.startDiscovery();
93.                 Toast.makeText(MainActivity.this, "Scanning devic
es!", Toast.LENGTH_LONG).show();
94.
```

Experiment 13

Develop an application to enable Bluetooth and Wi-fi through the radio button.

```
95.         }
96.     });
97.
98.     //wifi Intent
99.     wifiBtn.setOnClickListener(new View.OnClickListener() {
100.         @Override
101.         public void onClick(View v) {
102.             Intent wifi_intent = new Intent(Settings.Panel.AC
103. TION_WIFI);
104.             startActivity(wifi_intent);
105.         }
106.     });
107. }
108.
109. // Create a BroadcastReceiver for ACTION_FOUND.
110. private final BroadcastReceiver receiver = new BroadcastRecei
111. ver() {
112.     public void onReceive(Context context, Intent intent) {
113.         String action = intent.getAction();
114.         if (BluetoothDevice.ACTION_FOUND.equals(action)) {
115.             // Discovery has found a device. Get the Bluetooth
116. hDevice
117.             // object and its info from the Intent.
118.             BluetoothDevice device = intent.getParcelableExtr
119. a(BluetoothDevice.EXTRA_DEVICE);
120.             scannedDeviceList.add("Device: " + device.getName
121. () + "\nMAC: " + device.getAddress());
122.             adapter.notifyDataSetChanged(); //update list whe
123. n new device is dicovered
124.         }
125.     }
126. };
127.
128. //bluetooth -----
129.
130. void checkBluetoothPermissions() {
131.     if (bluetoothAdapter == null) {
132.         // Device doesn't support Bluetooth
133.         Toast.makeText(this, "Your device doesn't supports bl
134. uetooth service!", Toast.LENGTH_LONG).show();
135.         blue_btn.setText("Bluetooth: Not Support");
136.     }
137.     if (!bluetoothAdapter.isEnabled()) {
138.         Intent enableBtIntent = new Intent(BluetoothAdapter.A
139. CTION_REQUEST_ENABLE);
140.         startActivityForResult(enableBtIntent, BLUETOOTH_REQ_
141. CODE);
142.         blue_btn.setText("Bluetooth: On");
143.     }
144. }
```

Experiment 13

Develop an application to enable Bluetooth and Wi-fi through the radio button.

```
135.         }
136.     }
137.
138.     @Override
139.     protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
140.         super.onActivityResult(requestCode, resultCode, data);
141.
142.         if (requestCode == BLUETOOTH_REQ_CODE) {
143.             if (resultCode == RESULT_OK) {
144.                 Toast.makeText(MainActivity.this, "Bluetooth Permission Granted!", Toast.LENGTH_LONG).show();
145.             } else if (resultCode == RESULT_CANCELED) {
146.                 Toast.makeText(MainActivity.this, "Bluetooth Permission Denied/Failed!", Toast.LENGTH_LONG).show();
147.             }
148.         }
149.     }
150.
151.
152.     @Override
153.     protected void onDestroy() {
154.         super.onDestroy();
155.         // Don't forget to unregister the ACTION_FOUND receiver.
156.         unregisterReceiver(receiver);
157.     }
158.
159.     @Override
160.     protected void onStart() {
161.         super.onStart();
162.
163.         //Bluetooth
164.         //Intent filter for discovering devices
165.         // Register for broadcasts when a device is discovered.
166.         IntentFilter filter = new IntentFilter(BluetoothDevice.ACTION_FOUND);
167.         registerReceiver(receiver, filter);
168.     }
169.
170. }
```

Experiment 13

Develop an application to enable Bluetooth and Wi-fi through the radio button.

Snapshots:

