Ex. No. 9	**Discovering nearby Bluetooth devices**
Date of Exercise	**09.10.2024**

Aim

To create an app with a Button that initiates Bluetooth discovery and a ListView to display names of nearby devices as they are discovered.

Description

Discovering Nearby Bluetooth Devices refers to the process of searching for and identifying Bluetooth-enabled devices within a certain range using a Bluetooth-compatible device, such as a smartphone, tablet, or computer. This discovery process typically involves scanning for active Bluetooth signals in the vicinity and displaying a list of devices available for pairing or connection.

During this process:

- The Bluetooth device performing the scan will detect nearby devices that have their visibility settings enabled and are in range (usually between 10 to 100 meters, depending on the Bluetooth version and device power).
- Each detected device typically broadcasts its unique identifier, such as a name, type, and a media access control (MAC) address, which allows users to recognize it.
- Once discovered, users can select a device from the list to initiate a connection, often requiring authentication through a passkey or PIN for security purposes.

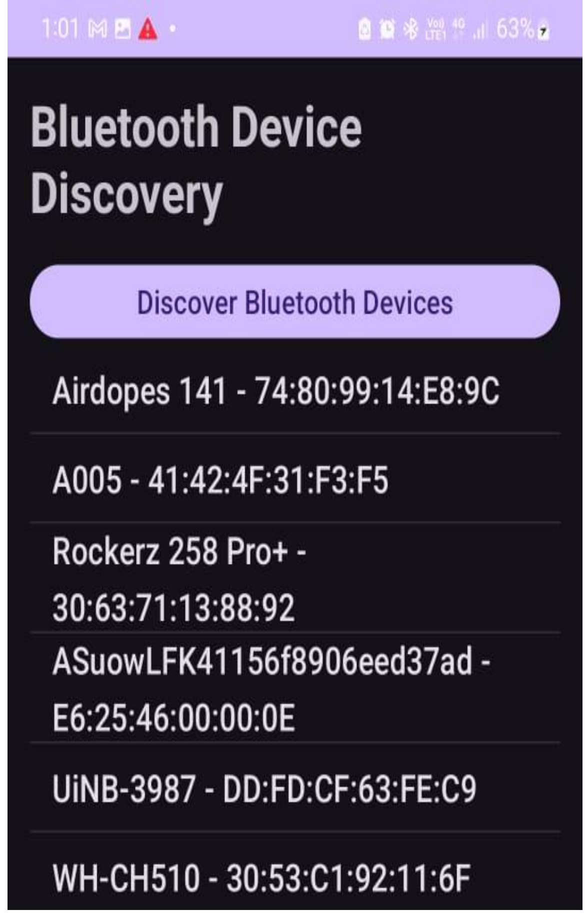
This process is essential for establishing connections between devices for data transfer, audio streaming, peripheral connections (like keyboards and headphones), and other Bluetooth services.

Program:

MainActivity.kt:

```
package com.example.ex_9

import android.Manifest
import android.bluetooth.BluetoothAdapter
import android.bluetooth.BluetoothDevice
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

class MainActivity : AppCompatActivity() {

    private lateinit var bluetoothAdapter: BluetoothAdapter
    private lateinit var btnDiscover: Button
    private lateinit var listViewDevices: ListView
    private lateinit var deviceListAdapter: ArrayAdapter<String>
    private val discoveredDevices = mutableListOf<String>()

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        btnDiscover = findViewById(R.id.btnDiscover)
        listViewDevices = findViewById(R.id.listViewDevices)

        // Initialize BluetoothAdapter
        bluetoothAdapter = BluetoothAdapter.getDefaultAdapter()
        if (bluetoothAdapter == null) {
            Toast.makeText(this, "Bluetooth not supported", Toast.LENGTH_SHORT).show()
            return
        }
        // Set up the ListView adapter
        deviceListAdapter = ArrayAdapter(this, android.R.layout.simple_list_item_1, discoveredDevices)
        listViewDevices.adapter = deviceListAdapter

        // Set button click listener to show connected devices and start discovery
        btnDiscover.setOnClickListener {
            showConnectedDevices() // Show already paired devices
            startBluetoothDiscovery() // Start discovering nearby devices
        }

        // Register for broadcasts when a device is discovered and when discovery is finished
        val filter = IntentFilter(BluetoothDevice.ACTION_FOUND)
        registerReceiver(receiver, filter)

        val filterDiscoveryFinished = IntentFilter(BluetoothAdapter.ACTION_DISCOVERY_FINISHED)
        registerReceiver(receiver, filterDiscoveryFinished)
    }

    private fun showConnectedDevices() {
        discoveredDevices.clear() // Clear the list to refresh with newly connected or discovered devices
        val pairedDevices = bluetoothAdapter.bondedDevices
        pairedDevices?.forEach { device ->
            val deviceName = device.name ?: "Unknown Device"
            val deviceAddress = device.address
            val deviceInfo = "$deviceName - $deviceAddress (Connected)"
            discoveredDevices.add(deviceInfo)
        }
        deviceListAdapter.notifyDataSetChanged()
    }

    private fun startBluetoothDiscovery() {
        // Check Bluetooth permissions for Android 12 and higher
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.S) {
            if (ContextCompat.checkSelfPermission(this, Manifest.permission.BLUETOOTH_SCAN) !=
PackageManager.PERMISSION_GRANTED ||
                ContextCompat.checkSelfPermission(this, Manifest.permission.BLUETOOTH_CONNECT) !=
PackageManager.PERMISSION_GRANTED) {
                ActivityCompat.requestPermissions(
                    this,
                    arrayOf(Manifest.permission.BLUETOOTH_SCAN,
Manifest.permission.BLUETOOTH_CONNECT),
                    1
                )
                return
            }
        } else {
            if (ContextCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
                ActivityCompat.requestPermissions(this, arrayOf(Manifest.permission.ACCESS_FINE_LOCATION),
1)
                return
            }
        }

    // BroadcastReceiver for ACTION_FOUND and ACTION_DISCOVERY_FINISHED
    private val receiver = object : BroadcastReceiver() {
        override fun onReceive(context: Context, intent: Intent) {
                }
                BluetoothAdapter.ACTION_DISCOVERY_FINISHED -> {
                    Toast.makeText(context, "Discovery finished", Toast.LENGTH_SHORT).show()
                }
            }
        }
    }

}
```

Output:



1:01 63%

Bluetooth Device Discovery

Airdopes 141 - 74:80:99:14:E8:9C

A005 - 41:42:4F:31:F3:F5

Rockerz 258 Pro+ - 30:63:71:13:88:92

ASuowLFK41156f8906eed37ad - E6:25:46:00:00:0E

UiNB-3987 - DD:FD:CF:63:FE:C9

WH-CH510 - 30:53:C1:92:11:6F

Result:

Thus the code is executed successfully and the output is displayed in the console window.