

Ex. No. : 8B

Date: 04/05/2025

Register No.: 221701056

Name: SNEKA SORNA.P.S

Send Email

Aim

Develop an application to send Email.

Procedure:

Step 1: Setting Up Your Environment

1. Install Android Studio: If it's not already installed, download and install Android Studio from the official Android Developer website.
2. Create a New Project: Open Android Studio, start a new project by selecting "New Project" > "Empty Activity". Name your project (e.g., "EmailSenderApp"), select Kotlin as the language, and choose an API level.

Step 2: Designing the User Interface

1. Open activity_main.xml: Navigate to app > res > layout > activity_main.xml.
2. Add UI Elements: Design your layout to include EditText fields for entering the recipient's email address, subject, and message, and a Button to send the email.

Step 3: Implementing Email Sending Functionality

1. Update MainActivity.kt: Add logic to create an email Intent when the "Send Email" button is clicked.

Step 4: Running Your Application

1. Choose a Device: Select an Android device or an emulator to run your application. Make sure the device has at least one email client installed.
2. Run the App: Click the "Run" button in Android Studio to build and run your application.
3. Test the Functionality: Fill in the recipient's email address, subject, and message, then click the "Send Email" button. An email client should open, pre-filled with the details you provided.



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.Exp8b"
        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>
    </application>

</manifest>
```



Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="24dp">

    <EditText
        android:id="@+id/editTextRecipient"
        android:hint="Recipient Email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textEmailAddress" />

    <EditText
        android:id="@+id/editTextSubject"
        android:hint="Subject"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp" />

    <EditText
        android:id="@+id/editTextMessage"
        android:hint="Message"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp"
        android:minLines="4"
        android:gravity="top"
```



```
        android:inputType="textMultiLine" />
```

```
<Button
```

```
    android:id="@+id/buttonSend"
```

```
    android:text="Send Email"
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginTop="20dp" />
```

```
</LinearLayout>
```



MainActivity.kt

```
package com.example.exp_8b

import android.content.Intent
import android.net.Uri
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

    private lateinit var editTextRecipient: EditText
    private lateinit var editTextSubject: EditText
    private lateinit var editTextMessage: EditText
    private lateinit var buttonSend: Button

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

        editTextRecipient = findViewById(R.id.editTextRecipient)
        editTextSubject = findViewById(R.id.editTextSubject)
        editTextMessage = findViewById(R.id.editTextMessage)
        buttonSend = findViewById(R.id.buttonSend)
        buttonSend.setOnClickListener {
```



```

        sendEmail()
    }
}

private fun sendEmail() {
    val recipient = editTextRecipient.text.toString()
    val subject = editTextSubject.text.toString()
    val message = editTextMessage.text.toString()

    if (recipient.isEmpty() || subject.isEmpty() || message.isEmpty()) {
        Toast.makeText(this, "Please fill all fields", Toast.LENGTH_SHORT).show()
        return
    }

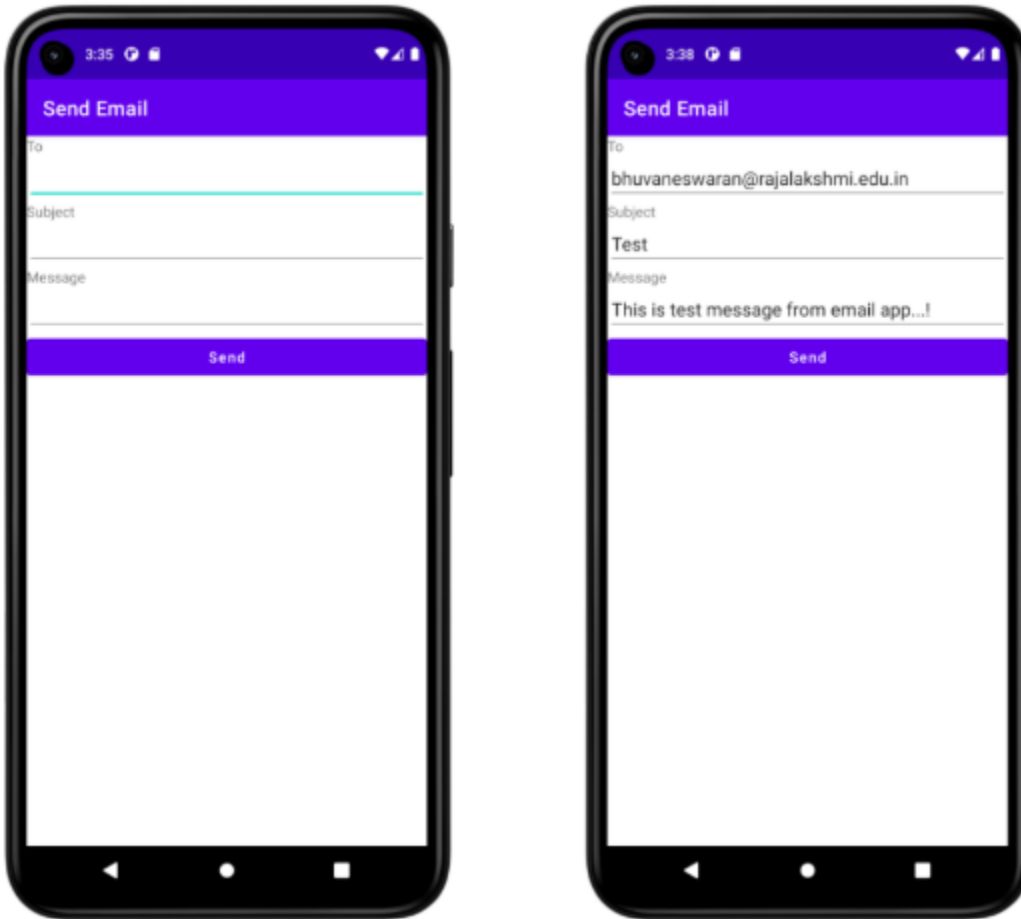
    val intent = Intent(Intent.ACTION_SENDTO).apply {
        data = Uri.parse("mailto:") // only email apps should handle this
        putExtra(Intent.EXTRA_EMAIL, arrayOf(recipient))
        putExtra(Intent.EXTRA_SUBJECT, subject)
        putExtra(Intent.EXTRA_TEXT, message)
    }

    try {
        startActivity(Intent.createChooser(intent, "Choose Email Client"))
    } catch (e: Exception) {
        Toast.makeText(this, "No email app found", Toast.LENGTH_SHORT).show()
    }
}
}

```



Output



Result:

The app sends an email by using an Intent to launch the device's default email client, allowing the user to specify the recipient, subject, and message and it is verified successfully.

