**MainActivty.java**

package com.example.internalstorage; import android.content.Context;

import android.os.Bundle;

import android.os.Environment; import android.view.View;

import android.widget.\*;

import androidx.appcompat.app.AppCompatActivity; import java.io.\*;

public class MainActivity extends AppCompatActivity { EditText dataInput, fileNameInput;

RadioButton internalStorageRadio, externalStorageRadio;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState);

getSupportActionBar().hide();

setContentView(R.layout.*activity\_main*); dataInput = findViewById(R.id.*dataInput*);

fileNameInput = findViewById(R.id.*fileNameInput*); internalStorageRadio = findViewById(R.id.*internalStorageRadio*); externalStorageRadio = findViewById(R.id.*externalStorageRadio*);

}

public void saveData(View view) {

String fileName = fileNameInput.getText().toString(); String data = dataInput.getText().toString();

if (fileName.isEmpty() || data.isEmpty()) { showToast("Enter filename and data"); return;

}

try {

if (internalStorageRadio.isChecked()) {

FileOutputStream fos = openFileOutput(fileName, Context.*MODE\_PRIVATE*); fos.write(data.getBytes());

fos.close();

showToast("Saved to Internal Storage");

} else if (externalStorageRadio.isChecked()) {

File file = new File(getExternalFilesDir(null), fileName); FileOutputStream fos = new FileOutputStream(file); fos.write(data.getBytes());

fos.close();

showToast("Saved to External Storage");

} else {

showToast("Select storage type");

}

} catch (Exception e) {

showToast("Error: " + e.getMessage());

}

}

public void loadData(View view) {

String fileName = fileNameInput.getText().toString(); if (fileName.isEmpty()) {

showToast("Enter filename"); return;

}

try {

FileInputStream fis;

if (internalStorageRadio.isChecked()) { fis = openFileInput(fileName);

} else if (externalStorageRadio.isChecked()) {

File file = new File(getExternalFilesDir(null), fileName); fis = new FileInputStream(file);

} else {

showToast("Select storage type"); return;

}

BufferedReader reader = new BufferedReader(new InputStreamReader(fis)); StringBuilder sb = new StringBuilder();

String line;

while ((line = reader.readLine()) != null) { sb.append(line).append("\n");

}

reader.close(); fis.close();

dataInput.setText(sb.toString().trim()); showToast("Data Loaded");

} catch (Exception e) {

showToast("Error: " + e.getMessage());

}

}

private void showToast(String message) {

Toast.*makeText*(this, message, Toast.*LENGTH\_SHORT*).show();

}

}

**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:padding="16dp" android:orientation="vertical">

<EditText

android:id="@+id/fileNameInput" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:hint="Enter Filename" />

<EditText

android:id="@+id/dataInput" android:layout\_width="match\_parent" android:layout\_height="150dp" android:hint="Enter Data" android:gravity="top" android:inputType="textMultiLine" />

<RadioGroup

android:id="@+id/storageOption" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:orientation="horizontal">

<RadioButton

android:id="@+id/internalStorageRadio" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Internal" />

<RadioButton

android:id="@+id/externalStorageRadio" android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" android:text="External" />

</RadioGroup>

<Button

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:text="Save"

android:onClick="saveData" />

<Button

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:text="Load"

android:onClick="loadData" />

</LinearLayout>

**AndroidManifest**

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="<http://schemas.android.com/apk/res/android>" xmlns:tools="[http://schemas.android.com/tools"](http://schemas.android.com/tools)>

**<uses-permission android:name="android.permission.READ\_EXTERNAL\_STORAGE" />**

**<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" android:maxSdkVersion="28"/>**

<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:supportsRtl="true"

android:theme="@style/Theme.InternalStorage" 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>

**For internal storage: Device File Explorer-**

**>data/data/com.example.yourproject\_name/files/nameOfTxt.txt**

**For external storage: Device File Explorer->storage/emulated/0/Android/data/ com.example.yourproject\_name/files/nameOfTxt.txt**