Example – Internal Storage

Creates a file in internal memory to store the details

To check the existence of the file go to Device file Explorer → data → data → package → filename

1. activity main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/relativeLayout"
    android:layout width="match parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/filenametext"
        android:layout width="wrap content"
        android:layout height="wrap content"
        android:layout alignParentLeft="true"
        android:layout_marginStart="48dp"
        android:layout marginLeft="48dp"
        android:layout marginTop="110dp"
        android:text="File Name:" />
    <TextView
        android:id="@+id/datatext"
        android:layout_width="wrap_content"
        android:layout height="wrap content"
        android:layout alignParentLeft="true"
        android:layout_alignParentRight="false"
        android:layout marginStart="48dp"
        android:layout marginLeft="48dp"
        android:layout marginTop="210dp"
        android:text="Data:" />
    <EditText
        android:id="@+id/filenameedit"
        android:layout width="wrap content"
        android:layout height="wrap content"
        android:layout alignParentRight="true"
        android:layout marginTop="100dp"
        android:layout marginEnd="50dp"
        android:layout_marginRight="50dp"
        android:ems="10"
```

```
android:inputType="textPersonName" />
               <EditText
                   android:id="@+id/dataedit"
                   android:layout width="wrap content"
                   android:layout height="wrap content"
                   android:layout alignParentRight="true"
                   android:layout marginTop="200dp"
                   android:layout marginEnd="50dp"
                   android:layout_marginRight="50dp"
                   android:ems="10"
                   android:inputType="textMultiLine" />
               <Button
                   android:id="@+id/savebutton"
                   android:layout_width="wrap_content"
                   android:layout height="wrap content"
                   android:layout marginStart="50dp"
                   android:layout marginLeft="50dp"
                   android:layout marginTop="340dp"
                   android:text="Save" />
               <Button
                   android:id="@+id/readbutton"
                   android:layout width="wrap content"
                   android:layout_height="wrap_content"
                   android:layout marginStart="250dp"
                   android:layout marginLeft="250dp"
                   android:layout marginTop="340dp"
                   android:text="Read" />
           </RelativeLayout>
           MainActivity.java
           // FileInputStream and FileOutputStream classes are used to read and write
           data into the file.
package com.example.exampleinternalstorage;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
```

2.

import android.widget.Toast;

```
import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;
public class MainActivity extends AppCompatActivity {
EditText filenameedit, dataedit;
Button savebutton, readbutton;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity main);
        filenameedit = findViewById(R.id.filenameedit);
        dataedit = findViewById(R.id.dataedit);
        savebutton = findViewById(R.id.savebutton);
        readbutton = findViewById(R.id.readbutton);
        savebutton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String filename = filenameedit.getText().toString();
                String data = dataedit.getText().toString();
                FileOutputStream fos;
                try{
                    fos = openFileOutput(filename, MODE PRIVATE);
//default mode is PRIVATE, can be APPEND etc.
                               fos.write(data.getBytes());
                               fos.close();
           Toast.makeText(getApplicationContext(),filename + "
           saved", Toast.LENGTH LONG).show();
                           } catch(FileNotFoundException e){
                              e.printStackTrace();
                           } catch (IOException e){
                              e.printStackTrace();
                       }
                   });
                   readbutton.setOnClickListener(new
           View.OnClickListener() {
                       @Override
                       public void onClick(View v) {
                           String filename =
           filenameedit.getText().toString();
                           StringBuffer stringBuffer = new
           StringBuffer();
                           try{
```

```
//Attaching BufferedReader to the FileInputStream by the help of
InputStreamReader.
//The BufferedReader class of Java is used to read the stream of characters from
the specified source (character-input stream)
                      BufferedReader inputReader = new
BufferedReader(new InputStreamReader(openFileInput(filename)));
                      String inpstring;
//Reading data line by line and storing it into the stringbuffer
           while ((inpstring=inputReader.readLine())!= null){
                                       stringBuffer.append(inpstring +
            "\n");
                                  }
                             }catch (IOException e){
                                  e.printStackTrace();
                             }
           Toast.makeText(getApplicationContext(),stringBuffer.toStr
            ing(),Toast.LENGTH_LONG).show();
                         }
                    });
                }
            }
```