Sabarivasan V
205001085
CSE - B

Ex 7: Android Application to write/read a file to/from the SD Card

Aim:

Develop an Android Application to write/read some contents to/from the SD Card.

1. In a TextView write the contents of the file.
2. Use another TextView to read the file name from the user.
3. On clicking 'Write' Button,

   Create a file mentioned in 2nd TextView.

   Write the contents (using 1st TextView) in the file.

   Store the file in the SD card.

4. On clicking 'Read' Button,

   Move to a new activity.
   Read the file name(TextView)
   Read the contents of the file from SD card and display in a new TextView.

Layouts Used: Main Activity and Read Intents. Edit and Text Views.

Code:

MainActivity.java:

```
package com.example.ex7;



import android.Manifest;
import android.content.Intent;

import android.content.pm.PackageManager;
```

```
import android.os.Bundle;
import android.os.Environment;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;


import androidx.annotation.NonNull;

import androidx.appcompat.app.AppCompatActivity;

import androidx.core.app.ActivityCompat;

import androidx.core.content.ContextCompat;


import java.io.File;

import java.io.FileOutputStream;

import java.io.IOException;


public class MainActivity extends AppCompatActivity {


   private static final int REQUEST_WRITE_EXTERNAL_STORAGE = 1;
    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);


        // Request the WRITE_EXTERNAL_STORAGE permission if not granted
```

```
            if (ContextCompat.checkSelfPermission(this,
Manifest.permission.WRITE_EXTERNAL_STORAGE) !=

                    PackageManager.PERMISSION_GRANTED) {

                ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE},
REQUEST_WRITE_EXTERNAL_STORAGE);

            } else {

                // Permission already granted, perform file operations

                Button button = findViewById(R.id.write);

                button.setOnClickListener(new View.OnClickListener() {

                    @Override

                    public void onClick(View v) {

                        EditText et1 = findViewById(R.id.et1);

                        String file = et1.getText().toString();


                        EditText et2 = findViewById(R.id.et2);

                        String content = et2.getText().toString();

                        createAndWriteFileToSDCard(file,content);

                    }
                });

            }


            Button read = findViewById(R.id.read1);

            read.setOnClickListener(new View.OnClickListener() {

                @Override

                public void onClick(View v) {

                    Intent intent = new Intent(MainActivity.this,Read.class);
```

```
                startActivity(intent);

            }

        });

    }


    // Handle permission request results

    @Override

    public void onRequestPermissionsResult(int requestCode,
@NonNull String[] permissions, @NonNull int[] grantResults) {

        super.onRequestPermissionsResult(requestCode, permissions,
grantResults);

        if (requestCode == REQUEST_WRITE_EXTERNAL_STORAGE) {

            if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {

                Toast.makeText(this, "Permission granted. Can write to SD card.",
Toast.LENGTH_SHORT).show();

                Button button = findViewById(R.id.write);
                button.setOnClickListener(new View.OnClickListener() {

                    @Override

                    public void onClick(View v) {

                        EditText et1 = findViewById(R.id.et1);

                        String file = et1.getText().toString();



                        EditText et2 = findViewById(R.id.et2);

                        String content = et2.getText().toString();

                        createAndWriteFileToSDCard(file,content);

                    }
                });
```

```
                } else {
                    Toast.makeText(this, "Permission denied. Cannot write to SD card.", Toast.LENGTH_SHORT).show();
                }
            }
        }

        private void createAndWriteFileToSDCard(String fileName,String fileContent) {
            // Check if external storage is available
            if (isExternalStorageWritable()) {
                File sdCard = Environment.getExternalStorageDirectory();
                File directory = new File(sdCard.getAbsolutePath() + "/ex7"); // Change to your desired directory
                directory.mkdirs();

                File file = new File(directory, fileName+".txt"); // Change the file name as needed

                try {
                    FileOutputStream fos = new FileOutputStream(file);
                    fos.write(fileContent.getBytes());
                    fos.close();
                    Toast.makeText(this, "File created and written to SD card", Toast.LENGTH_SHORT).show();
                } catch (IOException e) {
                    Log.e("FileWriteError", "Error writing to file on SD card: " + e.getMessage());
```

```
            }
        } else {
            Toast.makeText(this, "SD card is not available for writing.",
Toast.LENGTH_SHORT).show();
        }
    }

    private boolean isExternalStorageWritable() {
        String state = Environment.getExternalStorageState();
        return Environment.MEDIA_MOUNTED.equals(state);
    }
}
```

Read.java:

```
package com.example.ex7;

import android.Manifest;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.os.Environment;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
```

```
import androidx.annotation.NonNull;

import androidx.appcompat.app.AppCompatActivity;

import androidx.core.app.ActivityCompat; import

androidx.core.content.ContextCompat;


import java.io.BufferedReader;

import java.io.File;
import java.io.FileReader;

import java.io.IOException;


public class Read extends AppCompatActivity {


    private static final int REQUEST_READ_EXTERNAL_STORAGE = 2;


    private TextView fileContentsTextView;



    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.read);



         fileContentsTextView = findViewById(R.id.content);
          // Request the READ_EXTERNAL_STORAGE permission if not granted

         if (ContextCompat.checkSelfPermission(this,
Manifest.permission.READ_EXTERNAL_STORAGE)
```

```
                != PackageManager.PERMISSION_GRANTED) {

            ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.READ_EXTERNAL_STORAGE},
REQUEST_READ_EXTERNAL_STORAGE);

        } else {

            // Permission already granted, perform file reading
            Button readButton = findViewById(R.id.read2);

            readButton.setOnClickListener(new View.OnClickListener() {

                @Override

                public void onClick(View v) {

                    EditText fileNameEditText = findViewById(R.id.name);

                    String fileName = fileNameEditText.getText().toString();

                    readFileFromSDCard(fileName);

                }

            });

        }

    }


    // Handle permission request results

    @Override

    public void onRequestPermissionsResult(int requestCode,
@NonNull String[] permissions, @NonNull int[] grantResults) {

        super.onRequestPermissionsResult(requestCode, permissions,
grantResults);

        if (requestCode == REQUEST_READ_EXTERNAL_STORAGE) {

            if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
```

```
                    Toast.makeText(this, "Permission granted. Can read from SD card.",
Toast.LENGTH_SHORT).show();

                    Button readButton = findViewById(R.id.read2);

                    readButton.setOnClickListener(new View.OnClickListener() {
                        @Override

                        public void onClick(View v) {

                            EditText fileNameEditText = findViewById(R.id.name);

                            String fileName = fileNameEditText.getText().toString();

                            readFileFromSDCard(fileName);

                        }

                    });

                } else {

                    Toast.makeText(this, "Permission denied. Cannot read from SD card.",
Toast.LENGTH_SHORT).show();

                }

            }

        }


        private void readFileFromSDCard(String fileName) {

            if (isExternalStorageReadable()) {

                File sdCard = Environment.getExternalStorageDirectory();

                File directory = new File(sdCard.getAbsolutePath() + "/ex7"); // Change to
your directory
                File file = new File(directory, fileName + ".txt");



                if (file.exists()) {

                    try {
```

```
                    BufferedReader br = new BufferedReader(new FileReader(file));
                    StringBuilder text = new StringBuilder();

                    String line;


                    while ((line = br.readLine()) != null) {

                        text.append(line);

                        text.append('\n');

                    }


                    br.close();

                    fileContentsTextView.setText(text.toString());

                } catch (IOException e) {

                    Log.e("FileReadError", "Error reading file on SD card: " +
e.getMessage());

                    fileContentsTextView.setText("Error reading file.");

                }

            } else {

                fileContentsTextView.setText("File not found.");

            }

        } else {

            fileContentsTextView.setText("SD card is not available for reading.");
        }

    }


    private boolean isExternalStorageReadable() {
        String state = Environment.getExternalStorageState();
```

```
            return Environment.MEDIA_MOUNTED.equals(state)
|| Environment.MEDIA_MOUNTED_READ_ONLY.equals(state);

        }

    }
```

Activity_main.xml:

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

<androidx.constraintlayout.widget.ConstraintLayout
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:layout_width="match_parent"

    android:layout_height="match_parent"

    tools:context=".MainActivity">


    <TextView

        android:id="@+id/tv2"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:text="Enter text"

        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintHorizontal_bias="0.498"

        app:layout_constraintStart_toStartOf="parent"
```

```
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.387" />

<EditText
    android:id="@+id/et1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="184dp"
    android:ems="10"
    android:inputType="textPersonName"
    android:text=""
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<EditText
    android:id="@+id/et2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    android:text=""
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
```

```
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<TextView
    android:id="@+id/tv1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="128dp"
    android:text="Enter file name"
    android:textSize="24sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<Button
    android:id="@+id/write"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Write"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.65" />
```

```
        <Button
            android:id="@+id/read1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="156dp"
            android:text="Read"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.498"
            app:layout_constraintStart_toStartOf="parent" />


    </androidx.constraintlayout.widget.ConstraintLayout>
```

Read.xml:

```
    <?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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:layout_width="match_parent"
        android:layout_height="match_parent">


        <TextView
            android:id="@+id/textView3"
            android:layout_width="207dp"
```

```
    android:layout_height="47dp"

    android:text="File Content:"

    android:textSize="24sp"

    app:layout_constraintBottom_toBottomOf="parent"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toTopOf="parent"

    app:layout_constraintVertical_bias="0.602" />


<TextView

    android:id="@+id/textView"

    android:layout_width="125dp"

    android:layout_height="50dp"

    android:text="File Name"
    android:textSize="24sp"

    app:layout_constraintBottom_toBottomOf="parent"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintHorizontal_bias="0.461"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toTopOf="parent"

    app:layout_constraintVertical_bias="0.19" />


<EditText

    android:id="@+id/name"

    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    android:text=""
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.496"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.29" />
<EditText
    android:id="@+id/content"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    android:text=""
    android:textSize="24sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.496"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.724" />
```

```
        <Button
            android:id="@+id/read2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Read"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.498"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.407" />

    </androidx.constraintlayout.widget.ConstraintLayout>
```

AndroidManifest.xml:

```
    <?xml version="1.0" encoding="utf-8"?>
    <manifest xmlns:android="http://schemas.android.com/apk/res/android"
        package="com.example.ex7">
<application
android:allowBackup="true"
android:icon="@mipmap/ic_launcher"
android:label="@string/app_name"
android:roundIcon="@mipmap/ic_launcher_round"
android:supportsRtl="true"
```

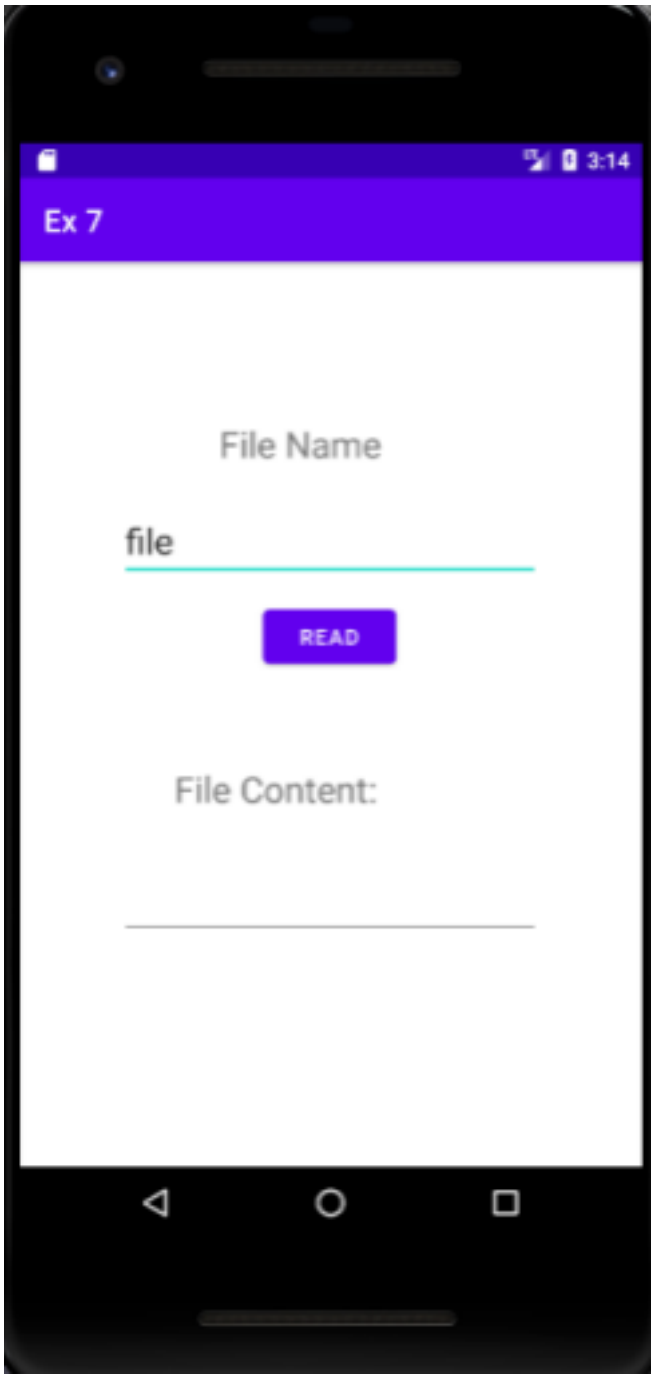
```
android:theme="@style/Theme.Ex7">

<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>

<activity android:name=".Read"></activity>

</application>

<uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

<uses-permission
android:name="android.permission.READ_EXTERNAL_STORAGE" />

<uses-permission
android:name="android.permission.MANAGE_EXTERNAL_STORAGE" /> </manifest>
```

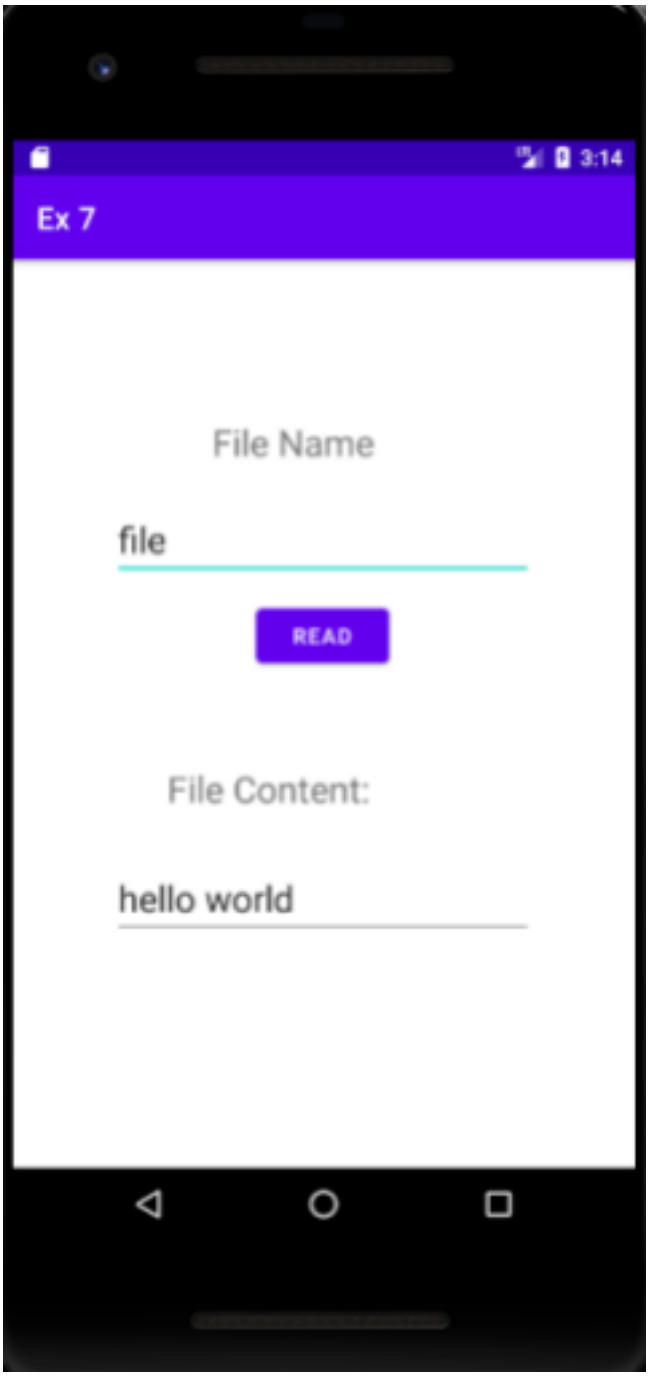
Output:

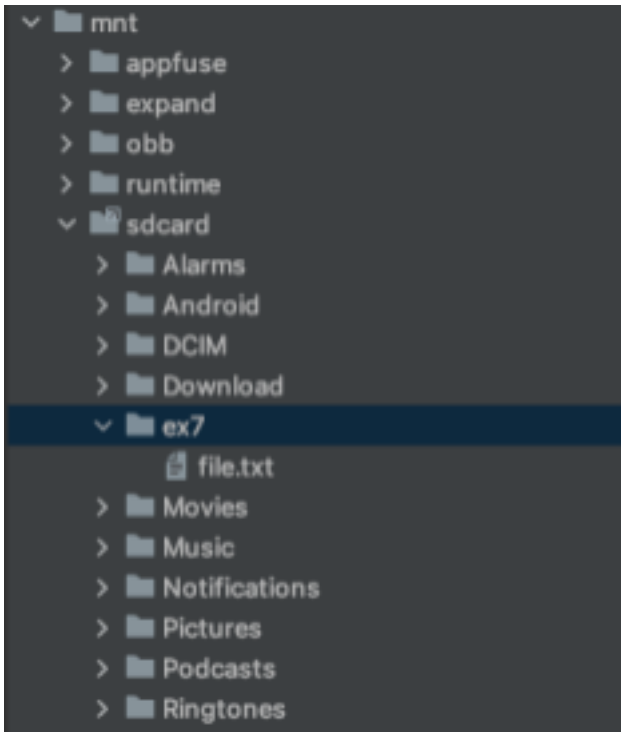

3:13
Ex 7
Enter file name
Enter text
WRITE
READ


3:13
Ex 7
Enter file name
file
Enter text
hello world
WRITE
READ


3:13
Ex 7
File Name
READ
File Content:


3:14
Ex 7
File Name
file
READ
File Content:


3:14
Ex 7
File Name
file
READ
File Content:
hello world



Best Practices:

- Used appropriate ids for buttons, views and intents
- Aligned views.

Learning Outcomes:

- Learnt to create a file
- Learnt to write to a file and store the file in SD card
- Learnt to read from a file that is in SD card