Ex. No. : 06 **Date: 17.02.2025**

Register No.: 221701060 **Name: Tamilarasi R**

SD Card

Aim

Implement an application to write the name and CGPA to SD card in text file format.

Procedure:

Step 1 : File -> NewProject

Provide the application name and Click “Next”

Step 2 : Select the target android devices

Select the minimum SDK to run the application. Click “Next”.

Step 3 : Choose the activity for the application (By default choose “Blank Activity).

Click “Next”.

Step 4 : Enter activity name and click &quot;Finish&quot;.

Step 5 : Edit the program.

Step 6 : Run the application, 2-ways to run the application.

1. Running through emulator

2. Running through mobile device

AndroidManifest.xml

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.ex6">

    <!-- Permissions for accessing external storage -->
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"
/>
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />


    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="SD Card File Writer"
        android:theme="@style/Theme.Ex6">
        <!-- Main Activity -->
        <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"?>
<RelativeLayout
xmlns:android="http://sche
mas.android.com/apk/res/a
ndroid"

android:layout_width="mat
ch_parent"
```

```
android:layout_height="ma
tch_parent">

   <!-- EditText for entering
Name -->
   <EditText

android:id="@+id/etName"

android:layout_width="mat
ch_parent"

android:layout_height="wr
ap_content"
      android:hint="Enter
Name"

android:layout_marginTop
="50dp"

android:layout_alignParent
Top="true"

android:padding="16dp"/>

   <!-- EditText for entering
Marks -->
   <EditText

android:id="@+id/etMarks"

android:layout_width="mat
ch_parent"
```

```
android:layout_height="wr
ap_content"
        android:hint="Enter
Marks"

android:inputType="numbe
r"

android:layout_below="@id
/etName"

android:layout_marginTop
="20dp"

android:padding="16dp"/>

    <!-- Button to trigger the
save action -->
    <Button

android:id="@+id/btnSaveD
ata"

android:layout_width="wra
p_content"

android:layout_height="wr
ap_content"
        android:text="Save
Data"

android:layout_below="@id
```

```
/etMarks"

android:layout_centerHoriz
ontal="true"

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

MainActivity.kt

```
 package com.example.ex6

import android.Manifest
import
android.content.pm.Packag
eManager
import android.os.Build
import android.os.Bundle
import
android.widget.EditText
import
android.widget.Toast
import
androidx.appcompat.app.A
ppCompatActivity
import
androidx.core.app.Activity
Compat
import
androidx.core.content.Cont
extCompat
import java.io.File
```

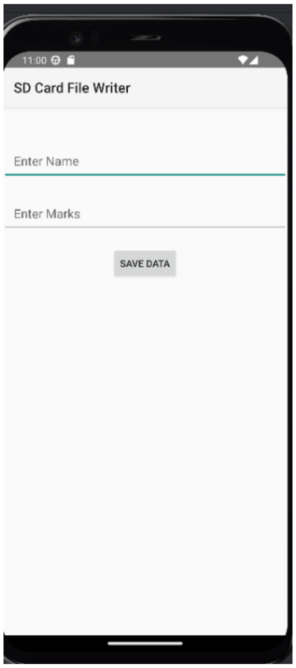
```
import
java.io.FileOutputStream
import java.io.IOException

class MainActivity :
AppCompatActivity() {

    private val
REQUEST_CODE_STORA
GE_PERMISSION = 1
    private lateinit var
etName: EditText
    private lateinit var
etMarks: EditText

    override fun
onCreate(savedInstanceSta
te: Bundle?) {

super.onCreate(savedInsta
nceState)

setContentView(R.layout.a
ctivity_main)

        // Initialize the
EditText fields
        etName =
findViewById(R.id.etName)
        etMarks =
findViewById(R.id.etMarks
)
```

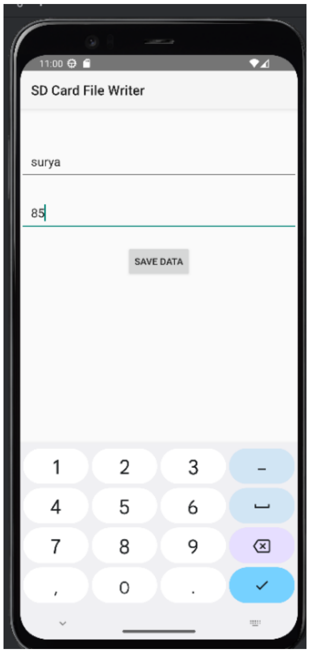
```
    // Check for
permissions before
allowing the user to save
data
    if
(Build.VERSION.SDK_IN
T >=
Build.VERSION_CODES.
M) {
      if
(ContextCompat.checkSelf
Permission(
          this,
Manifest.permission.WRIT
E_EXTERNAL_STORAGE
        ) !=
PackageManager.PERMIS
SION_GRANTED
      ) {

ActivityCompat.requestPer
missions(
          this,

arrayOf(Manifest.permissi
on.WRITE_EXTERNAL_S
TORAGE),

REQUEST_CODE_STORA
GE_PERMISSION
        )
      } else {
        // Permission is
already granted, proceed to
save data

setupSaveButton()
            }
        } else {
            // If on older
versions of Android,
permission is automatically
granted
            setupSaveButton()
        }
    }

    // Handle permission
result
    override fun
onRequestPermissionsResu
lt(
        requestCode: Int,
        permissions:
Array<String>,
        grantResults:
IntArray
    ) {

super.onRequestPermissio
nsResult(requestCode,
permissions, grantResults)
        if (requestCode ==
REQUEST_CODE_STORA
GE_PERMISSION) {
            if
```

```
(grantResults.isNotEmpty()
&& grantResults[0] ==
PackageManager.PERMIS
SION_GRANTED) {
                // Permission
granted, set up the save
button

setupSaveButton()
            } else {

Toast.makeText(this,
"Permission Denied",
Toast.LENGTH_SHORT).s
how()
            }
        }
    }

    // Setup button click to
save data to SD card
    private fun
setupSaveButton() {
        val btnSaveData =
findViewById<android.wid
get.Button>(R.id.btnSaveD
ata)

btnSaveData.setOnClickLi
stener {
            val name =
etName.text.toString()
            val marks =
etMarks.text.toString()

            if
(name.isNotEmpty() &&
marks.isNotEmpty()) {
                // Convert marks
to an integer
                val marksInt =
marks.toInt()

                // Write the data
to the file

writeToFile(name,
marksInt)
            } else {

Toast.makeText(this,
"Please enter both name
and marks",
Toast.LENGTH_SHORT).s
how()
            }
        }
    }

    // Function to write the
name and marks to a text
file
    private fun
writeToFile(name: String,
marks: Int) {
        try {
            // Get the file path
            val file =
File(getExternalFilesDir(n
ull), "student_marks.txt")

            // Open file output
stream in append mode
            val fos =
FileOutputStream(file,
true)

            // Prepare content to
be written to the file
            val content =
"Name: $name, Marks:
$marks\n"

fos.write(content.toByteArr
ay())
            fos.close()


Toast.makeText(this,
"Data saved to SD card",
Toast.LENGTH_SHORT).s
how()
        } catch (e:
IOException) {
            e.printStackTrace()

Toast.makeText(this,
"Failed to write to file",
Toast.LENGTH_SHORT).s
```

```
how()
        }
    }
}
```

Output :





surya

85

Result:
The Application was developed using Kotlin in Android Studio.