Practical-8

Implementation of Storage in Android

1. Create an android application to save data in a text file (internal storage). Then load file from memory and show in the view. **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/textView2"
        android:layout_width="337dp"
        android:layout_height="28dp" android:text="
        File Content " android:textAlignment="center"
        android:textColor="#000"
        android:textSize="24sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.52" />

    <Button android:id="@+id/write_button"
        android:layout_width="wrap_content"
        android:layout_height="48dp"
        android:layout_marginStart="160dp"
        android:layout_marginEnd="159dp"
        android:layout_marginBottom="16dp"
        android:text="Write"
        app:layout_constraintBottom_toTopOf
        ="@+id/read_button"
        app:layout_constraintEnd_toEndOf="p
        arent"
        app:layout_constraintHorizontal_bia
        s="0.0"
        app:layout_constraintStart_toStartO
```

Practical-8

Implementation of Storage in Android

```
    f="parent"
    app:layout_constraintTop_toTopOf="p
    arent"
    app:layout_constraintVertical_bias=
    "0.904" />

<Button android:id="@+id/read_button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="160dp"
    android:layout_marginEnd="158dp"
    android:layout_marginBottom="48dp"
    android:text="Read"
    app:layout_constraintBottom_toTopOf="@+id/textView2"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent" />

<EditText android:id="@+id/userInput"
    android:layout_width="319dp"
    android:layout_height="50dp"
    android:layout_marginStart="46dp"
    android:layout_marginTop="91dp"
    android:layout_marginEnd="46dp"
    android:ems="10"
    android:inputType="textPersonName"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<TextView android:id="@+id/content"
    android:layout_width="332dp"
    android:layout_height="306dp"
    android:layout_marginStart="33dp"
    android:layout_marginTop="21dp"
    android:layout_marginEnd="33dp"
    android:layout_marginBottom="6dp"
    android:text=""
    android:textAlignment="center"
    android:textColor="#000"
```

Practical-8

Implementation of Storage in Android

```
        app:layout_constraintBottom_toBot
        tomOf="parent"
        app:layout_constraintEnd_toEndOf=
        "parent"
        app:layout_constraintHorizontal_b
        ias="0.461"
        app:layout_constraintStart_toStar
        tOf="parent"
        app:layout_constraintTop_toBottom
        Of="@+id/textView2"
        app:layout_constraintVertical_bia
        s="0.0" />

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

Main_Activity.java

```
package com.example.l81; import
android.content.Context; import
android.os.Bundle; import android.view.View;
import android.widget.Button; import
android.widget.EditText; import
android.widget.TextView; import
android.widget.Toast; import
androidx.appcompat.app.AppCompatActivity;
import java.io.FileInputStream; import
java.io.FileOutputStream; import
java.io.IOException; import com.example.l81.R;


public class MainActivity extends AppCompatActivity implements
View.OnClickListener {

   // declare the variables
   Button read, write;
```

Practical-8

Implementation of Storage in Android

```
EditText userInput;
TextView fileContent;

private String filename = "demoFile.txt";

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    read = findViewById(R.id.read_button);
    write = findViewById(R.id.write_button);
    userInput = findViewById(R.id.userInput);
    fileContent = findViewById(R.id.content);

    read.setOnClickListener(this);
    write.setOnClickListener(this);
}

public void printMessage(String m) {
    Toast.makeText(this, m, Toast.LENGTH_LONG).show();
}

@Override
public void onClick(View view) {
    Button b = (Button) view;

    String b_text = b.getText().toString();

    switch (b_text.toLowerCase()) {
        case "write": {
        writeData(); break; } case
        "read": { readData();
        break;
        }
    }
}

private void writeData() {
```

Practical-8

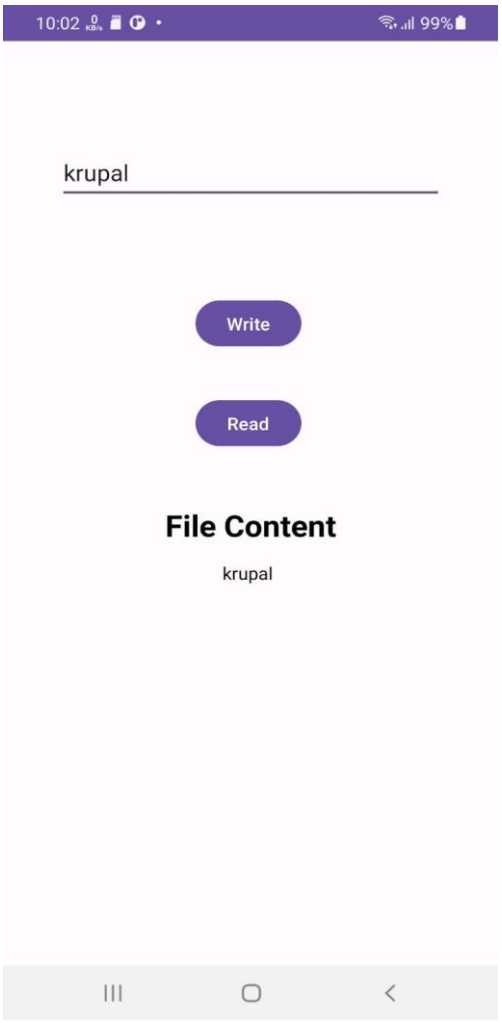
Implementation of Storage in Android

```
        try {
            FileOutputStream fos = openFileOutput(filename,
Context.MODE_PRIVATE);
            String data =
            userInput.getText().toString();
            fos.write(data.getBytes()); fos.flush();
            fos.close();
        } catch (IOException e) {
            e.printStackTrace(); } userInput.setText("");
        printMessage("writing to file " + filename + "completed...");
    }


    private void readData() {

        try {
            FileInputStream fin = openFileInput(filename);
            int a;

            StringBuilder temp = new StringBuilder();

            while ((a = fin.read()) != -1) {
                temp.append((char) a);
            }

            // setting text from the file.
            fileContent.setText(temp.toString());
            fin.close();

        } catch (IOException e) {
            e.printStackTrace();
        }

        printMessage("reading to file " + filename + " completed..");
    }

}
```

Practical-8

Implementation of Storage in Android

Output:-



Practical-8

Implementation of Storage in Android

2. Create an android application for storing and retrieving data file from external memory.

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent" android:orientation="vertical"
    android:padding="10dp">
    <EditText
        android:id="@+id/edit_text"
        android:layout_width="match_pare
        nt"
        android:layout_height="wrap_cont
        ent"/>
    <Button android:id="@+id/btnWrite"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginVertical="10dp"
        android:text="Write Data"/>
    <Button android:id="@+id/btnRead"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginVertical="10dp"
        android:text="Read Data"/>
    <Button android:id="@+id/btnClear"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginVertical="10dp"
        android:text="Clear Data"/>
</LinearLayout>
```

Practical-8

Implementation of Storage in Android

Main_Activity.java

```
package com.example.182; import
android.content.pm.PackageManager; import
android.Manifest; import android.os.Bundle; import
android.os.Environment; import android.view.View;
import android.widget.Button; import
android.widget.EditText; import
android.widget.Toast; import
androidx.appcompat.app.AppCompatActivity; import
java.io.File; import java.io.FileInputStream; import
java.io.FileNotFoundException; import
java.io.FileOutputStream; import
java.io.IOException; public class MainActivity
extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        final EditText editText = findViewById(R.id.edit_text);
        Button btnWrite = findViewById(R.id.btnWrite);
        Button btnRead = findViewById(R.id.btnRead);
        Button btnClear = findViewById(R.id.btnClear);
        if(checkSelfPermission(Manifest.permission.WRITE_EXTERNAL_STORAGE)=
                = PackageManager.PERMISSION_GRANTED){
        }else
        { requestPermissions(new String[]
{Manifest.permission.WRITE_EXTERNAL_STORAGE},200 );
        } if (isExternalStorageAvailable() || isExternalStorageReadable())
        { btnWrite.setOnClickListener(new View.OnClickListener() {
                @Override public void
                onClick(View v) { try {
                        FileOutputStream fileOutputStream = new
```

Practical-8

Implementation of Storage in Android

```
FileOutputStream(getStorageDir("demo.txt"), true);

fileOutputStream.write(editText.getText().toString().getBytes());
                        fileOutputStream.write("\n".toString().getBytes());
                        fileOutputStream.close();
                        Toast.makeText(getApplicationContext(), "Data
Written in the SDCARD!",
                                Toast.LENGTH_SHORT).show();
                        editText.setText("");
                    } catch (FileNotFoundException e) {
                        e.printStackTrace();
                    } catch (IOException e) {
                        e.printStackTrace();
                    }
                } }); btnRead.setOnClickListener(new
            View.OnClickListener() {
                @Override public void
                onClick(View v) { try {

FileInputStream fileInputStream = new
FileInputStream(getStorageDir("demo.txt"));
                        StringBuffer str = new StringBuffer();
                        int c;
                        while ((c = fileInputStream.read()) != -1) {
                            str.append((char) c);
                        }
                        editText.setText(str);
                    } catch (FileNotFoundException e) {

                        e.printStackTrace();
                    } catch (IOException e) {
                        e.printStackTrace();
                    }
                } });
```

Lab 8

Implementation of Storage in Android

```
            btnClear.setOnClickListener(new View.OnClickListener()
            {
                @Override public void
                onClick(View v) { try {
                        FileOutputStream fileOutputStream = new
FileOutputStream(getStorageDir("demo.txt"));
                        fileOutputStream.flush();
                        fileOutputStream.close();
                    } catch (FileNotFoundException e) {
                        e.printStackTrace();
                    } catch (IOException e) {
                        e.printStackTrace();
                    }
                    Toast.makeText(getApplicationContext(), "File Clear!",
Toast.LENGTH_SHORT).show();
                }
            });

        } else {
            Toast.makeText(getApplicationContext(), "SDCARD is not
available!", Toast.LENGTH_SHORT).show();
        }
    }
    // check if external storage is available
    public boolean isExternalStorageAvailable() {
        String state = Environment.getExternalStorageState();
        if (Environment.MEDIA_MOUNTED.equals(state)) {
            return true;
        } return
        false;
    }

    //checks if external storage is available for read
    public boolean isExternalStorageReadable() {
        String state = Environment.getExternalStorageState(); if
        (Environment.MEDIA_MOUNTED_READ_ONLY.equals(state)) {
        return true;
```

Lab 8

Implementation of Storage in Android

```
        } return
        false;
    }

    //get file path
    public String getStorageDir(String fileName) {
        File file = new File(Environment.getExternalStorageDirectory(),
"Demo"); if
        (!file.mkdirs()) {
        file.mkdirs();
        }
        String filePath = file.getAbsolutePath() + File.separator
+ fileName; return filePath;
    } @Override public void onRequestPermissionsResult(int
    requestCode, String[]
permissions, int[] grantResults) {
        super.onRequestPermissionsResult(requestCode, permissions,
grantResults);
        if(requestCode==200)
        {
            //writedata();
        }
    }
}
```

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">

    <uses-permission
        android:name="android.permission.WRITE_EXTERNAL_STORAGE"
        android:maxSdkVersion="30" />
```

Lab 8

Implementation of Storage in Android

```
    <uses-permission
        android:name="android.permission.READ_EXTERNAL_STORAGE"
        android:maxSdkVersion="30" />



    <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.L82" 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>
```

Lab 8

Implementation of Storage in Android

Output: