

Lab 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"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/fstTxt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="100dp"
        android:layout_marginTop="150dp"
        android:text="UserName" />

    <EditText
        android:id="@+id/txtName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="100dp"
        android:ems="10"/>

    <TextView
        android:id="@+id/secTxt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Password"
        android:layout_marginLeft="100dp" />

    <EditText
        android:id="@+id/txtCity"
        android:inputType="textPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="100dp"
        android:ems="10" />

    <Button
        android:id="@+id/btnSave"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="100dp"
        android:text="Save" />

</LinearLayout>
```

MainActivity.java

```
package com.example.practical8;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
```

```

import android.telecom.Call;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import java.io.FileOutputStream;

public class MainActivity extends AppCompatActivity {

    EditText uname, city;
    Button saveBtn;
    FileOutputStream fstream;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        uname = findViewById(R.id.txtName);
        city = findViewById(R.id.txtCity);
        saveBtn = findViewById(R.id.btnSave);

        saveBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String uName = uname.getText().toString()+"\n";
                String cityname = city.getText().toString();
                try{
                    fstream = openFileOutput("user_details", MODE_PRIVATE);
                    fstream.write(uName.getBytes());
                    fstream.write(cityname.getBytes());
                    fstream.close();
                    Toast.makeText(MainActivity.this, "User Details Saved Successfully!!", Toast.LENGTH_SHORT).show();
                    startActivity(new Intent(MainActivity.this, DetailsActivity.class));
                } catch (Exception e){
                    Toast.makeText(MainActivity.this, ""+e.getMessage(), Toast.LENGTH_SHORT).show();
                }
            }
        });
    }
}

```

activity_details.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/resultView"
        android:layout_gravity="center"
        android:layout_marginTop="170dp"
        android:textSize="20dp"/>

```

```

<Button
    android:id="@+id/btnBack"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="20dp"
    android:text="Back" />
</LinearLayout>

```

DetailsActivity.java

```

package com.example.practical8;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import java.io.FileInputStream;

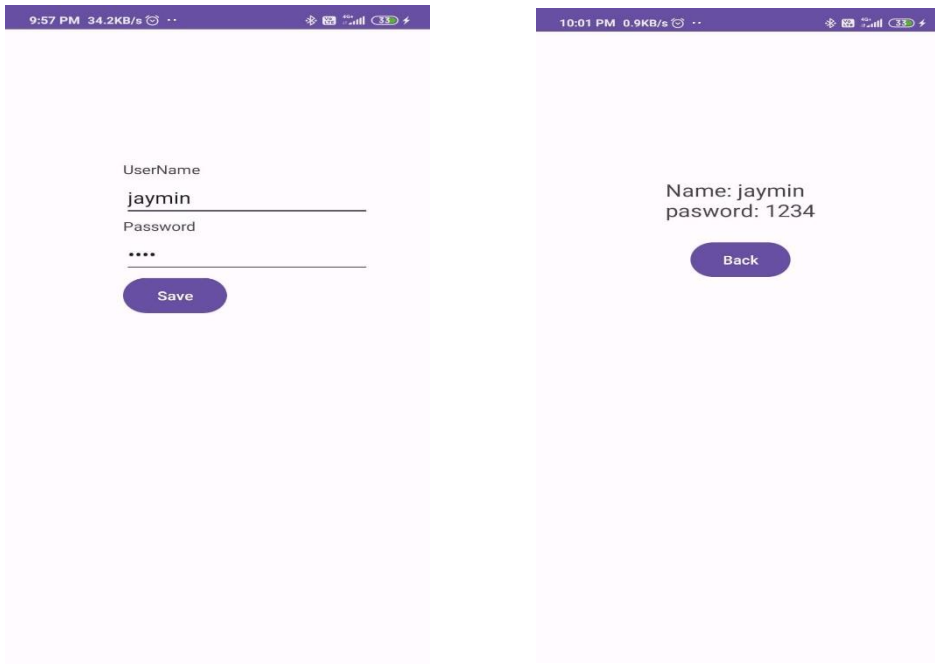
public class DetailsActivity extends AppCompatActivity {
    FileInputStream fstream;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_details);
        TextView textView = findViewById(R.id.resultView);
        Button btnBack = findViewById(R.id.btnBack);

        try{
            fstream = openFileInput("user_details");
            StringBuffer stringBuffer = new StringBuffer();
            int i;
            while((i = fstream.read()) != -1){
                stringBuffer.append((char)i);
            }
            fstream.close();
            String details[] = stringBuffer.toString().split("\n");
            textView.setText("Name: "+details[0] + "\nCity Name: "+details[1]);
        }catch (Exception e){
            Toast.makeText(this, ""+e.getMessage(), Toast.LENGTH_SHORT).show();
        }
        btnBack.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                finish();
            }
        });
    }
}

```

OutPut:



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

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

    <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.Practical8"
        tools:targetApi="31">
        <activity
            android:name=".SecondActivity"
            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=".DetailsActivity"
            android:exported="false" />
    </application>
```

```

        android:name=".MainActivity"
        android:exported="false">
    </activity>
</application>

```

```

</manifest>

```

activity_second.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">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Enter File Content"
        android:textSize="18sp"
    />
    <EditText
        android:id="@+id/edit_text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <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>

```

SecondActivity.java

```

package com.example.practical8;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.Manifest;
import android.content.pm.PackageManager;
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 java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;

public class SecondActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second);
        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) {
//            writeData();
        } else {
            requestPermissions(new String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE}, 200);
        }

        btnWrite.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (isExternalStorageAvailable() && isExternalStorageReadable()) {
                    try {
                        FileOutputStream fileOutputStream = new FileOutputStream(getStorageDir("demo.txt"), true);
                        fileOutputStream.write(editText.getText().toString().getBytes());
                        fileOutputStream.write("\n".toString().getBytes());
                        fileOutputStream.close();
                        Toast.makeText(SecondActivity.this, "File Content is Written...!!", Toast.LENGTH_SHORT).show();
                        editText.setText("");
                    } catch (Exception e) {
                        Toast.makeText(SecondActivity.this, "" + e.getMessage(), Toast.LENGTH_SHORT).show();
                    }
                } else {
                    Toast.makeText(SecondActivity.this, "External Storage is Not Available...!!",
Toast.LENGTH_SHORT).show();
                }
            }
        });

        btnRead.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (isExternalStorageAvailable() && isExternalStorageReadable()) {
                    try {
                        FileInputStream fileInputStream = new FileInputStream(getStorageDir("demo.txt"));
                        StringBuffer stringBuffer = new StringBuffer();
                        int c;
                        while ((c = fileInputStream.read()) != -1) {
                            stringBuffer.append((char) c);
                        }
                        editText.setText(stringBuffer);
                    } catch (Exception e) {

```

```

        }
    } else {
        Toast.makeText(SecondActivity.this, "External Storage is Not Available..!!",
Toast.LENGTH_SHORT).show();
    }
}
});

btnClear.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (isExternalStorageAvailable() && isExternalStorageReadable()) {
            try {
                FileOutputStream fileOutputStream = new FileOutputStream(getStorageDir("demo.txt"));
                fileOutputStream.flush();
                fileOutputStream.close();
                Toast.makeText(SecondActivity.this, "File Content is Flush... ", Toast.LENGTH_SHORT).show();
            } catch (Exception e) {
                Toast.makeText(SecondActivity.this, "" + e.getMessage(), Toast.LENGTH_SHORT).show();
            }
        } else {
            Toast.makeText(SecondActivity.this, "External Storage is Not Available..!!",
Toast.LENGTH_SHORT).show();
        }
    }
});
}

@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[]
grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if (requestCode == 200) {
        Toast.makeText(this, "Permission is Granted..!!", Toast.LENGTH_SHORT).show();
    }
}

public boolean isExternalStorageAvailable() {
    String state = Environment.getExternalStorageState();
    if (Environment.MEDIA_MOUNTED.equals(state)) {
        return true;
    }
    return false;
}

public boolean isExternalStorageReadable() {
    String state = Environment.getExternalStorageState();
    if (Environment.MEDIA_MOUNTED_READ_ONLY.equals(state)) {
        return true;
    }
    return false;
}

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;  
}  
}
```

Output:

