SMD TASK March 6, 2025

Name: Muhammad Abu Huraira

Roll: 22F-3853

Section: BCS-6B

Main activity.java

package com.example.permissions;  
  
import android.os.Bundle;  
import android.Manifest;  
import android.annotation.SuppressLint;  
import android.content.pm.PackageManager;  
import android.os.Bundle;  
import android.widget.Button;  
import android.widget.Toast;  
import androidx.activity.EdgeToEdge;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
import androidx.core.content.ContextCompat;  
import androidx.core.graphics.Insets;  
import androidx.core.view.ViewCompat;  
import androidx.core.view.WindowInsetsCompat;  
  
public class MainActivity extends AppCompatActivity {  
  
 private static final int  
 *CAMERA\_PERMISSION\_REQUEST\_CODE* = 1;  
 private static final int  
 *LOCATION\_PERMISSION\_REQUEST\_CODE* = 2;  
 private static final int  
 *INTERNET\_PERMISSION\_REQUEST\_CODE* = 3;  
 private static final int  
 *STORAGE\_PERMISSION\_REQUEST\_CODE* = 4;  
 Button cameraButton, locationBtn, internetBtn,  
 storageBtn;  
 @SuppressLint("MissingInflatedId")  
 @Override  
 protected void onCreate(Bundle  
 savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 EdgeToEdge.*enable*(this);  
 setContentView(R.layout.*activity\_main*);  
// Initialize buttons  
 cameraButton = findViewById(R.id.*button*);  
 locationBtn = findViewById(R.id.*button2*);  
 internetBtn = findViewById(R.id.*button3*);  
 storageBtn = findViewById(R.id.*button4*); //  
// Add a button for storage permission  
// Add padding for system bars  
 ViewCompat.*setOnApplyWindowInsetsListener*(findViewById(R.id.*main*), (v, insets) -> {  
 Insets systemBars =  
 insets.getInsets(WindowInsetsCompat.Type.*systemBars* ());  
 v.setPadding(systemBars.left,  
 systemBars.top, systemBars.right,  
 systemBars.bottom);  
 return insets;  
 });  
// Set an OnClickListener for each button  
// to request permissions  
 cameraButton.setOnClickListener(v ->  
 requestCameraPermission());  
 locationBtn.setOnClickListener(v ->  
 requestLocationPermission());  
 internetBtn.setOnClickListener(v ->  
 requestInternetPermission());  
 storageBtn.setOnClickListener(v ->  
 requestStoragePermission());  
 }  
 // Request camera permission  
 private void requestCameraPermission() {  
 if (ContextCompat.*checkSelfPermission*(this,  
 Manifest.permission.*CAMERA*)  
 !=  
 PackageManager.*PERMISSION\_GRANTED*) {  
 ActivityCompat.*requestPermissions*(this,  
 new String[]{Manifest.permission.*CAMERA*},  
 *CAMERA\_PERMISSION\_REQUEST\_CODE*);  
 } else {  
 Toast.*makeText*(this, "Camera permission already granted", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 // Request location permission  
 private void requestLocationPermission() {  
 if (ContextCompat.*checkSelfPermission*(this,  
 Manifest.permission.*ACCESS\_FINE\_LOCATION*) !=PackageManager.*PERMISSION\_GRANTED* &&ContextCompat.*checkSelfPermission*(this,  
 Manifest.permission.*ACCESS\_COARSE\_LOCATION*) !=  
 PackageManager.*PERMISSION\_GRANTED*) {  
 ActivityCompat.*requestPermissions*(this,  
 new  
 String[]{Manifest.permission.*ACCESS\_FINE\_LOCATION*,  
 Manifest.permission.*ACCESS\_COARSE\_LOCATION*},  
 *LOCATION\_PERMISSION\_REQUEST\_CODE*);  
 } else {  
 Toast.*makeText*(this, "Location permission already granted",  
 Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 // Inform user that internet permission is  
// granted by default  
 private void requestInternetPermission() {  
 Toast.*makeText*(this, "Internet permission is granted by default", Toast.*LENGTH\_SHORT*).show();  
 }  
 // Request external storage permission  
 private void requestStoragePermission() {  
// For Android 9 (API 28) and below,  
// request both read and write external storage  
// permissions  
 if (ContextCompat.*checkSelfPermission*(this,  
 Manifest.permission.*READ\_EXTERNAL\_STORAGE*) !=  
 PackageManager.*PERMISSION\_GRANTED* ||  
 ContextCompat.*checkSelfPermission*(this,  
 Manifest.permission.*WRITE\_EXTERNAL\_STORAGE*) !=  
 PackageManager.*PERMISSION\_GRANTED*) {  
 ActivityCompat.*requestPermissions*(this,  
 new String[]{  
 Manifest.permission.*READ\_EXTERNAL\_STORAGE*,Manifest.permission.*WRITE\_EXTERNAL\_STORAGE* }, *STORAGE\_PERMISSION\_REQUEST\_CODE*);  
 Toast.*makeText*(this, "Requesting storage permissions", Toast.*LENGTH\_SHORT*).show();  
// Add a toast here  
 } else {  
 Toast.*makeText*(this, "Storage permission already granted",  
 Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 // Handle the user's response to permissions  
 @Override  
 public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {  
 super.onRequestPermissionsResult(requestCode,  
 permissions, grantResults);  
 switch (requestCode) {  
 case *CAMERA\_PERMISSION\_REQUEST\_CODE*:  
 if (grantResults.length > 0 &&  
 grantResults[0] ==  
 PackageManager.*PERMISSION\_GRANTED*) {  
 Toast.*makeText*(this, "Camera permission granted", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(this, "Camera permission denied", Toast.*LENGTH\_SHORT*).show();  
 }  
 break;  
 case *LOCATION\_PERMISSION\_REQUEST\_CODE*:  
 if (grantResults.length > 0 &&  
 grantResults[0] ==  
 PackageManager.*PERMISSION\_GRANTED*) {  
 Toast.*makeText*(this, "Location permission granted", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(this, "Location permission denied", Toast.*LENGTH\_SHORT*).show();  
 }  
 break;  
 case *STORAGE\_PERMISSION\_REQUEST\_CODE*:  
 if (grantResults.length > 0) {  
 boolean readGranted =  
 grantResults[0] ==  
 PackageManager.*PERMISSION\_GRANTED*;  
 boolean writeGranted =  
 grantResults[1] ==  
 PackageManager.*PERMISSION\_GRANTED*;  
 if (readGranted &&  
 writeGranted) {  
 Toast.*makeText*(this,  
 "Storage permission granted",  
 Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(this,  
 "Storage permission denied",  
 Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 break;  
 default:  
 super.onRequestPermissionsResult(requestCode,  
 permissions, grantResults);  
 }  
 }  
}

XML File:

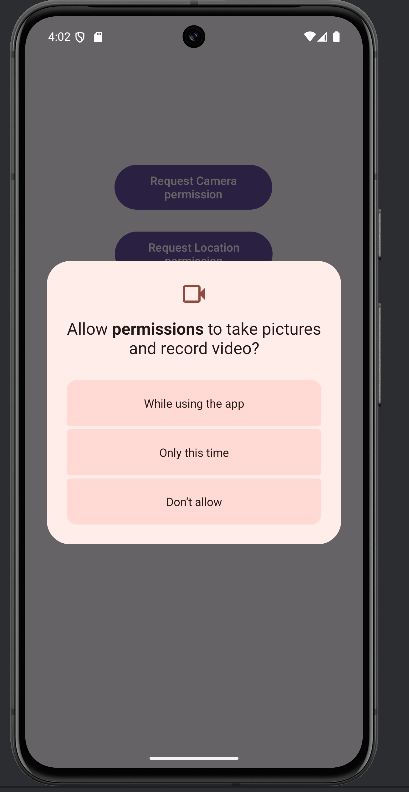
<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="192dp"  
 android:layout\_height="62dp"  
 android:layout\_marginStart="109dp"  
 android:layout\_marginTop="127dp"  
 android:layout\_marginEnd="111dp"  
 android:text="Request Camera permission"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 tools:ignore="TextSizeCheck" />  
 <Button  
 android:id="@+id/button2"  
 android:layout\_width="192dp"  
 android:layout\_height="62dp"  
 android:layout\_marginStart="109dp"  
 android:layout\_marginTop="19dp"  
 android:layout\_marginEnd="110dp"  
 android:text="Request Location permission"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/button"  
 tools:ignore="TextSizeCheck" />  
 <Button  
 android:id="@+id/button3"  
 android:layout\_width="192dp"  
 android:layout\_height="62dp"  
 android:layout\_marginStart="108dp"  
 android:layout\_marginTop="45dp"  
 android:layout\_marginEnd="111dp"  
 android:layout\_marginBottom="354dp"  
 android:text="Request Internet permission"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.516"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/button2"  
 app:layout\_constraintVertical\_bias="0.163"  
 tools:ignore="TextSizeCheck" />  
 <Button  
 android:id="@+id/button4"  
 android:layout\_width="218dp"  
 android:layout\_height="85dp"  
 android:layout\_marginStart="108dp"  
 android:layout\_marginTop="68dp"  
 android:layout\_marginEnd="111dp"  
 android:layout\_marginBottom="316dp"  
 android:text="Request External storage permission"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.516"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/button3"  
 tools:ignore="TextSizeCheck" />  
</androidx.constraintlayout.widget.ConstraintLayout>

Manifest file:

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools">  
  
 <!-- Permissions -->  
 <uses-feature  
 android:name="android.hardware.camera"  
 android:required="false" />  
  
 <uses-permission android:name="android.permission.MANAGE\_EXTERNAL\_STORAGE"  
 tools:ignore="ScopedStorage" />  
 <uses-permission android:name="android.permission.CAMERA" />  
 <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />  
 <uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" />  
 <uses-permission android:name="android.permission.INTERNET" />  
  
 <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.Permissions"  
 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>

Output:

Requesting camera permissions



A white rectangular sign with black text

AI-generated content may be incorrect.

A screenshot of a cell phone

AI-generated content may be incorrect.

A white rectangular sign with black text

AI-generated content may be incorrect.

A screenshot of a phone

AI-generated content may be incorrect.

A white rectangular object with black text

AI-generated content may be incorrect.

End of file