**PROGRAM 8: Build an application that reads text and image data from one activity and displays the output on a different activity page with the help of SharedPreference.**

**NAME: LENA GEO**

**REGISTER NUMBER: 2241139**

**MainActivity.kt**

package com.example.p8  
  
import android.app.Activity  
import android.content.Context  
import android.content.Intent  
import android.content.SharedPreferences  
import android.graphics.Bitmap  
import android.net.Uri  
import android.os.Bundle  
import android.provider.MediaStore  
import android.util.Base64  
import android.view.View  
import android.view.ViewGroup  
import android.widget.BaseAdapter  
import android.widget.Button  
import android.widget.EditText  
import android.widget.GridView  
import android.widget.ImageView  
import android.widget.Toast  
import androidx.appcompat.app.AppCompatActivity  
import java.io.ByteArrayOutputStream  
import kotlin.collections.ArrayList  
  
class MainActivity : AppCompatActivity() {  
  
 lateinit var sharedPreferences: SharedPreferences  
 lateinit var editor: SharedPreferences.Editor  
 lateinit var gridView: GridView  
 private val PICK\_IMAGE\_REQUEST = 1  
 private val imageUriList = ArrayList<Uri>()  
 private val encodedImagesList = ArrayList<String>()  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
 // Initialize SharedPreferences  
 sharedPreferences = getSharedPreferences("MyPrefs", Context.*MODE\_PRIVATE*)  
 editor = sharedPreferences.edit()  
  
 val editText: EditText = findViewById(R.id.*editText*)  
 gridView = findViewById(R.id.*gridView*)  
 val buttonSelectImages: Button = findViewById(R.id.*buttonSelectImages*)  
 val buttonSubmit: Button = findViewById(R.id.*buttonSubmit*)  
  
 // Handle Select Images button click  
 buttonSelectImages.setOnClickListener {  
 openGallery()  
 }  
  
 // Handle Submit button click  
 buttonSubmit.setOnClickListener {  
 val textInput = editText.*text*.toString()  
  
 if (imageUriList.*isNotEmpty*()) {  
 // Encode selected images to Base64 strings  
 for (uri in imageUriList) {  
 val bitmap = MediaStore.Images.Media.getBitmap(this.*contentResolver*, uri)  
 val encodedImage = encodeImageToBase64(bitmap)  
 encodedImagesList.add(encodedImage)  
 }  
  
 // Store text and image list in SharedPreferences  
 editor.putString("text", textInput)  
 editor.putStringSet("images", encodedImagesList.*toSet*())  
 editor.apply()  
  
 // Start DisplayActivity  
 val intent = Intent(this, DisplayActivity::class.*java*)  
 startActivity(intent)  
 } else {  
 Toast.makeText(this, "Please select at least one image", Toast.*LENGTH\_SHORT*).show()  
 }  
 }  
 }  
  
 // Open gallery to select multiple images  
 private fun openGallery() {  
 val intent = Intent(Intent.*ACTION\_PICK*, MediaStore.Images.Media.*EXTERNAL\_CONTENT\_URI*)  
 intent.putExtra(Intent.*EXTRA\_ALLOW\_MULTIPLE*, true)  
 startActivityForResult(Intent.createChooser(intent, "Select Pictures"), PICK\_IMAGE\_REQUEST)  
 }  
  
 // Handle the result of the image selection  
 override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {  
 super.onActivityResult(requestCode, resultCode, data)  
 if (requestCode == PICK\_IMAGE\_REQUEST && resultCode == Activity.*RESULT\_OK*) {  
 if (data != null) {  
 if (data.*clipData* != null) {  
 // Handle multiple images  
 val clipData = data.*clipData* for (i in 0 *until* clipData!!.*itemCount*) {  
 val imageUri = clipData.getItemAt(i).*uri* imageUriList.add(imageUri)  
 }  
 } else if (data.*data* != null) {  
 // Handle single image  
 val imageUri = data.*data* if (imageUri != null) {  
 imageUriList.add(imageUri)  
 }  
 }  
 // Display selected images in GridView  
 displaySelectedImages()  
 }  
 }  
 }  
  
 // Display the selected images in GridView  
 private fun displaySelectedImages() {  
 gridView.*adapter* = ImageAdapter(this, imageUriList)  
 }  
  
 // Helper function to encode bitmap to Base64 string  
 private fun encodeImageToBase64(bitmap: Bitmap): String {  
 val byteArrayOutputStream = ByteArrayOutputStream()  
 bitmap.compress(Bitmap.CompressFormat.*PNG*, 100, byteArrayOutputStream)  
 val byteArray = byteArrayOutputStream.toByteArray()  
 return Base64.encodeToString(byteArray, Base64.*DEFAULT*)  
 }  
  
 // ImageAdapter for displaying Uri images  
 inner class ImageAdapter(private val context: Context, private val imageList: ArrayList<Uri>) : BaseAdapter() {  
  
 override fun getCount(): Int {  
 return imageList.size  
 }  
  
 override fun getItem(position: Int): Any {  
 return imageList[position]  
 }  
  
 override fun getItemId(position: Int): Long {  
 return position.toLong()  
 }  
  
 override fun getView(position: Int, convertView: View?, parent: ViewGroup?): View {  
 val imageView: ImageView  
 if (convertView == null) {  
 imageView = ImageView(context)  
 imageView.*layoutParams* = ViewGroup.LayoutParams(300, 300)  
 imageView.*scaleType* = ImageView.ScaleType.*CENTER\_CROP* } else {  
 imageView = convertView as ImageView  
 }  
  
 imageView.setImageURI(imageList[position])  
 return imageView  
 }  
 }  
}

**DisplayActivity.kt**

package com.example.p8  
  
import android.content.Context  
import android.graphics.Bitmap  
import android.graphics.BitmapFactory  
import android.os.Bundle  
import android.util.Base64  
import android.view.View  
import android.view.ViewGroup  
import android.widget.BaseAdapter  
import android.widget.GridView  
import android.widget.ImageView  
import android.widget.TextView  
import androidx.appcompat.app.AppCompatActivity  
  
class DisplayActivity : AppCompatActivity() {  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_display*)  
  
 val sharedPreferences = getSharedPreferences("MyPrefs", Context.*MODE\_PRIVATE*)  
  
 val textView: TextView = findViewById(R.id.*textView*)  
 val gridView: GridView = findViewById(R.id.gridViewDisplay)  
  
 // Retrieve text and image list from SharedPreferences  
 val text = sharedPreferences.getString("text", "No text available")  
 val imageSet = sharedPreferences.getStringSet("images", *emptySet*())  
  
 // Decode each Base64 image and store in an array  
 val decodedImageList = ArrayList<Bitmap>()  
 imageSet?.*forEach* **{** encodedImage **->** val decodedString = Base64.decode(encodedImage, Base64.*DEFAULT*)  
 val decodedBitmap = BitmapFactory.decodeByteArray(decodedString, 0, decodedString.size)  
 decodedImageList.add(decodedBitmap)  
 **}** // Set text and populate GridView with images  
 textView.*text* = text  
 gridView.*adapter* = BitmapAdapter(this, decodedImageList)  
 }  
  
 // BitmapAdapter for displaying Bitmap images  
 inner class BitmapAdapter(private val context: Context, private val bitmaps: ArrayList<Bitmap>) : BaseAdapter() {  
  
 override fun getCount(): Int {  
 return bitmaps.size  
 }  
  
 override fun getItem(position: Int): Any {  
 return bitmaps[position]  
 }  
  
 override fun getItemId(position: Int): Long {  
 return position.toLong()  
 }  
  
 override fun getView(position: Int, convertView: View?, parent: ViewGroup?): View {  
 val imageView: ImageView  
 if (convertView == null) {  
 imageView = ImageView(context)  
 imageView.*layoutParams* = ViewGroup.LayoutParams(300, 300)  
 imageView.*scaleType* = ImageView.ScaleType.*CENTER\_CROP* } else {  
 imageView = convertView as ImageView  
 }  
  
 imageView.setImageBitmap(bitmaps[position])  
 return imageView  
 }  
 }  
}

**activity\_main.xml**

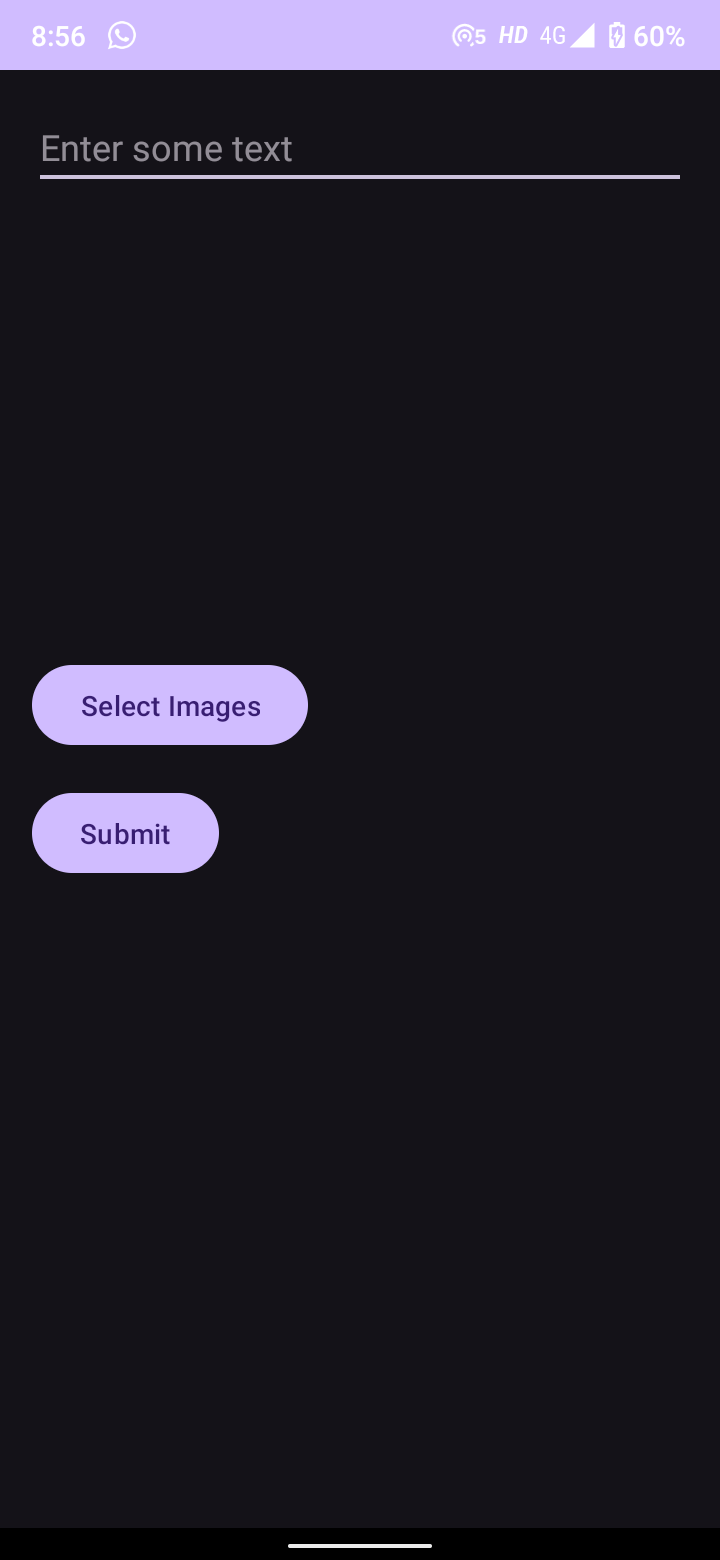
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:padding="16dp">  
  
 <EditText  
 android:id="@+id/editText"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Enter some text" />  
  
 <GridView  
 android:id="@+id/gridView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="200dp"  
 android:layout\_marginTop="16dp"  
 android:numColumns="3"  
 android:verticalSpacing="10dp"  
 android:horizontalSpacing="10dp" />  
  
 <Button  
 android:id="@+id/buttonSelectImages"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Select Images"  
 android:layout\_marginTop="16dp" />  
  
 <Button  
 android:id="@+id/buttonSubmit"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Submit"  
 android:layout\_marginTop="16dp" />  
</LinearLayout>

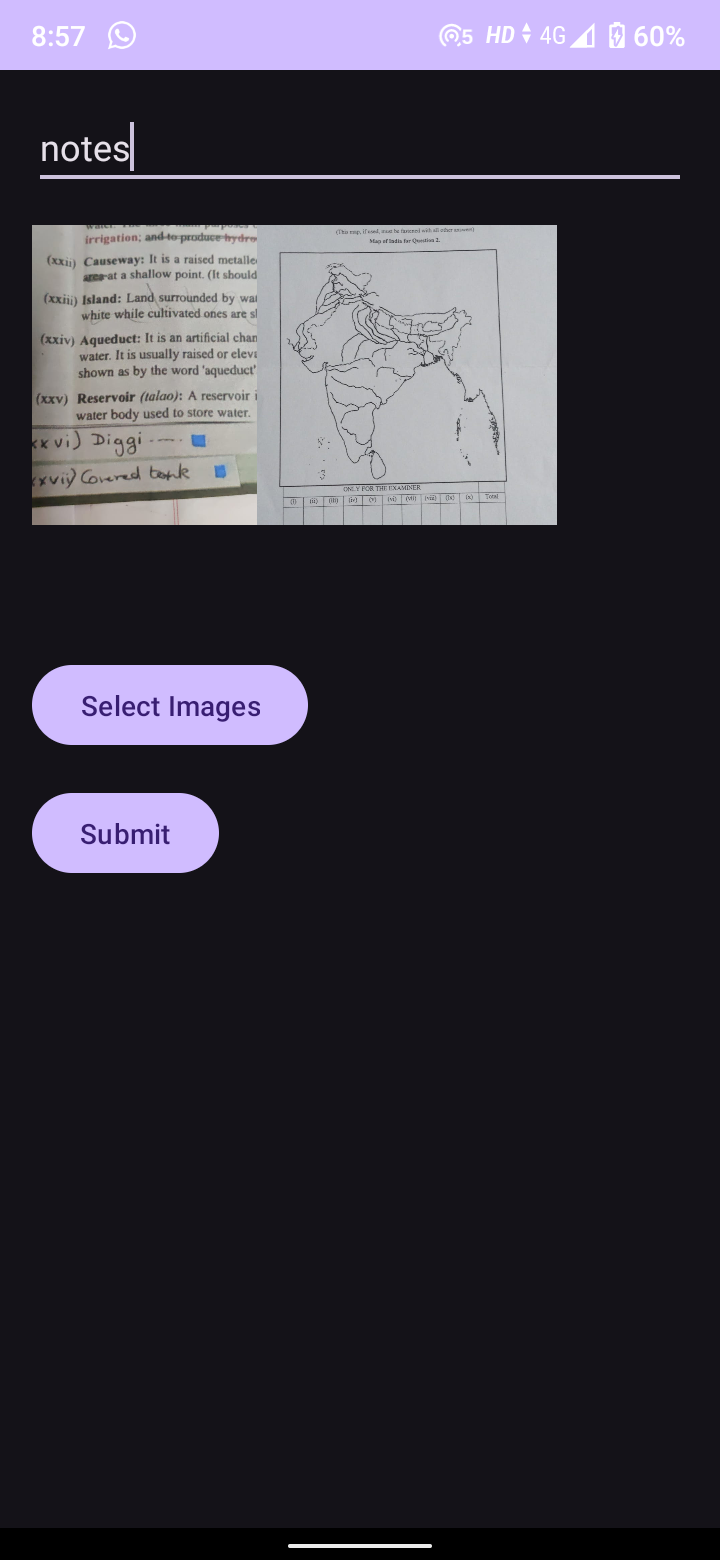
**activity\_display.xml**

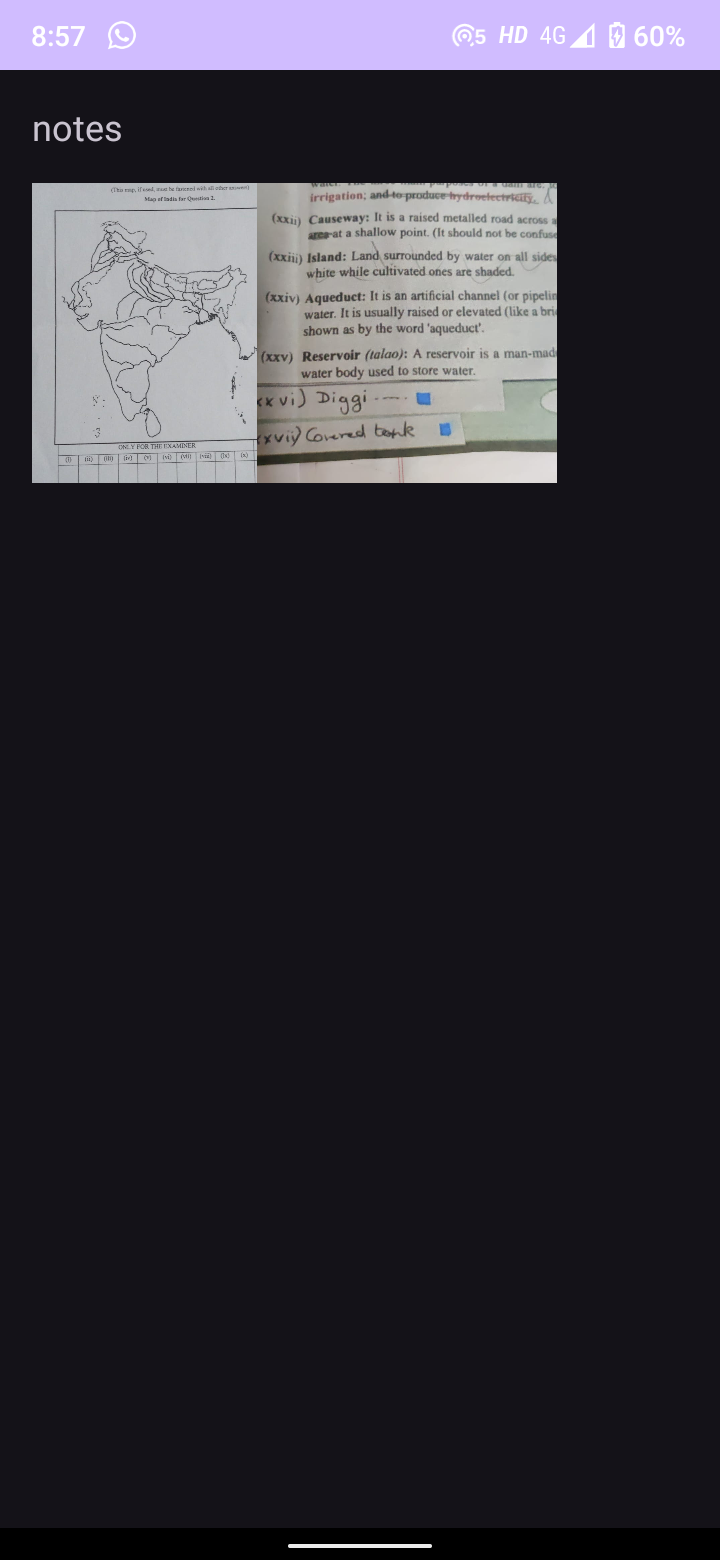
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:padding="16dp">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Text will be displayed here"  
 android:textSize="18sp"  
 android:layout\_marginBottom="16dp" />  
  
 <GridView  
 android:id="@+id/gridViewDisplay"  
 android:layout\_width="match\_parent"  
 android:layout\_height="200dp"  
 android:numColumns="3"  
 android:verticalSpacing="10dp"  
 android:horizontalSpacing="10dp" />  
</LinearLayout>

**OUTPUT**

**Complexity: Selecting and uploading of multiple images**

****

****

****