Sanat kitabım uygulaması

main activity xml kodları

<?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">  
  
 <androidx.recyclerview.widget.RecyclerView  
 android:id="@+id/recyclerView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_marginStart="1dp"  
 android:layout\_marginTop="1dp"  
 android:layout\_marginEnd="1dp"  
 android:layout\_marginBottom="1dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

Art activity xml kodları :

<?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=".Artactivity">  
  
 <ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="selectImage"  
 app:srcCompat="@drawable/seciliresim"  
 tools:layout\_editor\_absoluteX="55dp"  
 tools:layout\_editor\_absoluteY="76dp" />  
  
 <EditText  
 android:id="@+id/artnameText"  
 android:layout\_width="233dp"  
 android:layout\_height="58dp"  
 android:ems="10"  
 android:hint="Enter Art Name "  
 android:inputType="textPersonName"  
 tools:layout\_editor\_absoluteX="85dp"  
 tools:layout\_editor\_absoluteY="315dp" />  
  
 <EditText  
 android:id="@+id/artistNameText"  
 android:layout\_width="233dp"  
 android:layout\_height="58dp"  
 android:ems="10"  
 android:hint="Enter Artist Name "  
 android:inputType="textPersonName"  
 tools:layout\_editor\_absoluteX="85dp"  
 tools:layout\_editor\_absoluteY="380dp" />  
  
 <EditText  
 android:id="@+id/yearText"  
 android:layout\_width="233dp"  
 android:layout\_height="58dp"  
 android:ems="10"  
 android:hint="Enter year"  
 android:inputType="textPersonName"  
 tools:layout\_editor\_absoluteX="85dp"  
 tools:layout\_editor\_absoluteY="450dp" />  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:onClick="save"  
 android:text="Save"  
 tools:layout\_editor\_absoluteX="150dp"  
 tools:layout\_editor\_absoluteY="518dp" />  
</androidx.constraintlayout.widget.ConstraintLayout>

Art activity kodları ve onclickler

package com.selimcinar.sanatkitabmuygulamasjava;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import android.view.View;  
  
public class Artactivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_artactivity*);  
 }  
  
 public void save(View view){  
  
 }  
 public void selectImage(View view){  
  
 }  
}

Menu ekleme işlemleri

Menulerin kendine ait klasörü var .

Res altında new directory dosya adı menu ile menu eklenir.





İşlemleri ile menü eklenir eklenen menu artbook\_menu xml kodları :

<?xml version="1.0" encoding="utf-8"?>  
<menu xmlns:android="http://schemas.android.com/apk/res/android">  
<item android:id="@+id/add\_art" android:title="Add art"> //title dediği menu adı başlığı>  
</menu>

Main activity kodları :

package com.selimcinar.sanatkitabmuygulamasjava;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.Menu;  
import android.view.MenuInflater;  
import android.view.MenuItem;  
import android.view.View;  
  
import com.selimcinar.sanatkitabmuygulamasjava.databinding.ActivityMainBinding;  
  
public class MainActivity extends AppCompatActivity {  
  
 //View binding ekleme  
 private ActivityMainBinding binding;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 binding = ActivityMainBinding.*inflate*(getLayoutInflater());  
 View view = binding.getRoot();  
 setContentView(view);  
  
  
 }  
  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 //Seçenekli menu oluştuğunda ne olsun  
 MenuInflater menuInflater = getMenuInflater();  
 menuInflater.inflate(R.menu.*artbook\_menu*,menu);  
  
 return super.onCreateOptionsMenu(menu);  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(@NonNull MenuItem item) {  
 //Seçili menü elemanına tıklanınca ne olacak  
 if (item.getItemId() == R.id.*add\_art*){  
 Intent intent = new Intent(this,Artactivity.class);  
 startActivity(intent);  
 }  
 return super.onOptionsItemSelected(item);  
 }  
}

View Binding eklendi

package com.selimcinar.sanatkitabmuygulamasjava;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.SimpleAdapter;  
  
import com.selimcinar.sanatkitabmuygulamasjava.databinding.ActivityArtactivityBinding;  
  
public class Artactivity extends AppCompatActivity {  
 private ActivityArtactivityBinding binding;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_artactivity*);  
 binding = ActivityArtactivityBinding.*inflate*(getLayoutInflater());  
 View view = binding.getRoot();  
 setContentView(view);  
  
 }  
  
 public void save(View view){  
  
 }  
 public void selectImage(View view){  
  
 }  
}

Permission İzinler

Manifest xml dosya ayarlarınını yapıldığı yer

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools" >  
<!--İZİN KULLANIR DOSYA OKUMA İZNİ -->  
<uses-permission android:name="android.permission.READ\_EXTERNAL\_STORAGE"></uses-permission>  
<!--33 ve üstünde api seviyesi -->  
<uses-permission android:name="android.permission.READ\_MEDIA\_IMAGES"></uses-permission>

<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:supportsRtl="true"  
 android:theme="@style/Theme.SanatKitabımUygulamasıJava"  
 tools:targetApi="31" >  
 <activity  
 android:name=".Artactivity"  
 android:exported="false" />  
 <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>

Permission izin alma main activity kodları :

package com.selimcinar.sanatkitabmuygulamasjava;  
  
import androidx.activity.result.ActivityResult;  
import androidx.activity.result.ActivityResultCallback;  
import androidx.activity.result.ActivityResultLauncher;  
import androidx.activity.result.contract.ActivityResultContracts;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
import androidx.core.content.ContextCompat;  
  
import android.Manifest;  
import android.content.Intent;  
import android.content.pm.PackageManager;  
import android.graphics.Bitmap;  
import android.graphics.ImageDecoder;  
import android.net.Uri;  
import android.os.Build;  
import android.os.Bundle;  
import android.provider.MediaStore;  
import android.view.View;  
import android.widget.SimpleAdapter;  
import android.widget.Toast;  
  
import com.google.android.material.snackbar.Snackbar;  
import com.selimcinar.sanatkitabmuygulamasjava.databinding.ActivityArtactivityBinding;  
  
public class Artactivity extends AppCompatActivity {  
 //Global kapsamlar  
 Bitmap selectedImage ;  
 ActivityResultLauncher <Intent> activityResultLauncher; // Galeriye gitmek için galeriye gidince ne olsun  
 ActivityResultLauncher <String> permissionLauncher; // İzin almak için izin alınca ne olsun  
 private ActivityArtactivityBinding binding;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_artactivity);  
 binding = ActivityArtactivityBinding.inflate(getLayoutInflater());  
 View view = binding.getRoot();  
 setContentView(view);  
 registerLauncher();  
 }  
  
 public void save(View view){  
  
 }  
 public void selectImage(View view){  
 if (Build.VERSION.SDK\_INT>= Build.VERSION\_CODES.TIRAMISU){  
 // Android 33+ -> READ\_MEDİA\_IMAGES  
 //Api seviyesine bakmadan izin sorgula(activitesi this,Manifest içinde izinlerde Dosya okuma iznini) != eşitdeğil ise Paketyöneticisinde.izin verilmişse  
 //Granted izin verilmiş Denied izin verilmemiş  
 if (ContextCompat.checkSelfPermission(this, Manifest.permission.READ\_MEDIA\_IMAGES)!= PackageManager.PERMISSION\_GRANTED){  
 //ActivityCompate kullanıcı izin vermedi ona mantığı göstereyim mi izin vermesi için  
 if (ActivityCompat.shouldShowRequestPermissionRationale(this, Manifest.permission.READ\_MEDIA\_IMAGES)) {  
 //Mantığı göstermek için alert Diaologa benzeyern snack bar yapısı  
 //Snackabar.yap(görünüm,İçindeki yazı ,Kullanıcı bir buttona tıklayana kadar bekle),buttonaksiyonu("button adı",tıklanınca ne olacağına dair listener){  
 // Tıklanınca şu olsun {  
 // }}  
 Snackbar.make(view,"Permisson needed for gallery",Snackbar.LENGTH\_INDEFINITE).setAction("Give Permission", new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 //Request permission izin sorgulama  
 permissionLauncher.launch(Manifest.permission.READ\_MEDIA\_IMAGES);  
 }  
 }).show(); //snackbarı göster.  
 }  
 else {  
 //Request permission izin sorgulama  
 permissionLauncher.launch(Manifest.permission.READ\_MEDIA\_IMAGES);  
 }  
 }  
 else {  
 //gallery'e git  
 /\*Intent intentToGallery objesi = new ıntent(Intente tıklanınca galeriye git ve resmi al resmiUrı urlden konumundan al )\*/  
 Intent intentToGallery = new Intent(Intent.ACTION\_PICK, MediaStore.Images.Media.EXTERNAL\_CONTENT\_URI);  
 activityResultLauncher.launch(intentToGallery);  
 }  
 }  
 else {  
 // android 33 ve altı --> READ\_EXTERNAL\_STORAGE  
 //Api seviyesine bakmadan izin sorgula(activitesi this,Manifest içinde izinlerde Dosya okuma iznini) != eşitdeğil ise Paketyöneticisinde.izin verilmişse  
 //Granted izin verilmiş Denied izin verilmemiş  
 if (ContextCompat.checkSelfPermission(this, Manifest.permission.READ\_EXTERNAL\_STORAGE)!= PackageManager.PERMISSION\_GRANTED){  
 //ActivityCompate kullanıcı izin vermedi ona mantığı göstereyim mi izin vermesi için  
 if (ActivityCompat.shouldShowRequestPermissionRationale(this, Manifest.permission.READ\_EXTERNAL\_STORAGE)) {  
 //Mantığı göstermek için alert Diaologa benzeyern snack bar yapısı  
 //Snackabar.yap(görünüm,İçindeki yazı ,Kullanıcı bir buttona tıklayana kadar bekle),buttonaksiyonu("button adı",tıklanınca ne olacağına dair listener){  
 // Tıklanınca şu olsun {  
 // }}  
 Snackbar.make(view,"Permisson needed for gallery",Snackbar.LENGTH\_INDEFINITE).setAction("Give Permission", new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 //Request permission izin sorgulama  
 permissionLauncher.launch(Manifest.permission.READ\_EXTERNAL\_STORAGE);  
 }  
 }).show(); //snackbarı göster.  
 }  
 else {  
 //Request permission izin sorgulama  
 permissionLauncher.launch(Manifest.permission.READ\_EXTERNAL\_STORAGE);  
 }  
 }  
 else {  
 //gallery'e git  
 /\*Intent intentToGallery objesi = new ıntent(Intente tıklanınca galeriye git ve resmi al resmiUrı urlden konumundan al )\*/  
 Intent intentToGallery = new Intent(Intent.ACTION\_PICK, MediaStore.Images.Media.EXTERNAL\_CONTENT\_URI);  
 activityResultLauncher.launch(intentToGallery);  
 }  
 }  
  
 }  
  
 private void registerLauncher(){  
 activityResultLauncher = registerForActivityResult(new ActivityResultContracts.StartActivityForResult(), new ActivityResultCallback<ActivityResult>() {  
 @Override  
 public void onActivityResult(ActivityResult result) {  
 //Aktiviye gidince cevap verme  
 if (result.getResultCode()==RESULT\_OK) {  
 Intent intentFromResult = result.getData();  
 if (intentFromResult != null){  
 Uri imageData = intentFromResult.getData();  
 //binding.imageView.setImageURI(imageData);  
  
 try {  
 if(Build.VERSION.SDK\_INT >= 28) {  
 //ImageDecoder görsel dönüştürücü kodlayıcı apı 28 ve üstünde gçerli  
 ImageDecoder.Source source = ImageDecoder.createSource(Artactivity.this.getContentResolver(),imageData);  
 selectedImage = ImageDecoder.decodeBitmap(source);  
 binding.imageView.setImageBitmap(selectedImage);  
 }  
 else {  
 //Resimi dönüştürme işlemi apı 28 ve altında  
 selectedImage = MediaStore.Images.Media.getBitmap(Artactivity.this.getContentResolver(),imageData);  
 binding.imageView.setImageBitmap(selectedImage);  
 }  
  
 }  
 catch (Exception e){  
 e.printStackTrace(); // hataları loglara yaz  
 }  
 }  
 }  
 }  
 });  
  
 //İzin almak için yapılan bazı işlemler.  
 permissionLauncher=registerForActivityResult(new ActivityResultContracts.RequestPermission(), new ActivityResultCallback<Boolean>() {  
 @Override  
 public void onActivityResult(Boolean result) {  
 // result doğruysa izin verildi yada verilmedi  
 if (result){  
 //permission granted izin verildi  
 Intent intentToGallery=new Intent(Intent.ACTION\_PICK,MediaStore.Images.Media.EXTERNAL\_CONTENT\_URI);  
 activityResultLauncher.launch(intentToGallery);  
 }  
 else {  
 //permission denied.  
 Toast.makeText(Artactivity.this, "Permission needed!", Toast.LENGTH\_SHORT).show();  
 }  
  
 }  
 });  
 }  
}

Küçük görsel oluşturmak işlemi

public Bitmap makeSmallerImage(Bitmap image, int maximumSize){  
 //Küçük görsel oluşturmak  
  
 int width = image.getWidth();  
 int height = image.getHeight();  
  
 //Resimlerin oranı ayarlama işlemi bitmap ratio  
 float bitmapRatio = (float) width / (float) height;  
  
 //Oran birden büyükse işlemler  
 if (bitmapRatio > 1) {  
 //landscape image yatay görsel  
 width = maximumSize;  
 height = (int) (width / bitmapRatio);  
 }  
 else {  
 // portrait image dikey resim  
 height = maximumSize;  
 width = (int) (height \* bitmapRatio);  
 }  
  
  
  
 return image.*createScaledBitmap*(image,width,height,true);  
  
}

// Sqlite veritabanına kaydetme işlemi

public void save(View view){  
 // Dışardan girilen değerler değişkenlerde tutuldu.  
 String name = binding.artnameText.getText().toString();  
 String artistName = binding.artistNameText.getText().toString();  
 String year = binding.yearText.getText().toString();  
 Bitmap smallImage= makeSmallerImage(selectedImage,300);  
  
 //Image resimleri veriye 1,0 ' a byta döndürme  
 ByteArrayOutputStream outputStream = new ByteArrayOutputStream();  
 smallImage.compress(Bitmap.CompressFormat.*PNG*,50,outputStream);  
 byte [] byteArray = outputStream.toByteArray();  
  
 try {  
 //Veritabanına SqlLite'a kaydetme işlemleri  
  
 //Veritabanı oluştur  
 database = this.openOrCreateDatabase("Arts",*MODE\_PRIVATE*,null);  
 //Veritabanına tablo oluştur  
 database.execSQL("CREATE TABLE IF NOT EXISTS arts(id INTEGER PRIMARY KEY,artname VARCHAR,paintername VARCHAR,year VARCHAR,image BLOB)");  
 //Tabloya eklemeler yap , ? işaretleri yerine sqlitestatement 'deki bağlanan yerler örnek 1 name gelecek.  
 String sqlString = "INSERT INTO arts (artname,paintername,year,image) VALUES (?,?,?,?)";  
 //SQLiteStatement Bağlma binding işlemlerini kolaylaştırır.  
 SQLiteStatement sqLiteStatement = database.compileStatement(sqlString);  
 sqLiteStatement.bindString(1,name);  
 sqLiteStatement.bindString(2,artistName);  
 sqLiteStatement.bindString(3,year);  
 sqLiteStatement.bindBlob(4,byteArray);  
 //İşlemleri execute et çalıştır.  
 sqLiteStatement.execute();  
 }  
 catch (Exception e){  
 e.printStackTrace();  
 }  
  
 Intent intent = new Intent(Artactivity.this, MainActivity.class);  
 //addFlags ile öncesini sil yerine getir gibi işlemler yapılır.  
 //Flag\_Actıvıty\_Clear\_Top tüm aktiviteleri kapat yeni açacağımı aç  
 intent.addFlags(Intent.*FLAG\_ACTIVITY\_CLEAR\_TOP*);  
 startActivity(intent);  
}

Verileri çekme işlemi

Art Classı

package com.selimcinar.sanatkitabmuygulamasjava;  
  
public class Art {  
 String name ;  
 int id ;  
  
  
 //Özelliklere ait clasa constructor ekle  
 public Art (String name,int id){  
 this.name = name;  
 this.id = id;  
 }  
}

Main activity içerisinde

public class MainActivity extends AppCompatActivity {  
  
 //View binding ekleme  
 private ActivityMainBinding binding;  
 //Art clasından bir Array list oluştur.  
 ArrayList <Art> artArrayList;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 binding = ActivityMainBinding.*inflate*(getLayoutInflater());  
 View view = binding.getRoot();  
 setContentView(view);  
  
 getData();  
 }  
 public void getData(){  
 //Verileri çekme işlemi  
 try {  
 //Databaseyi aç  
 SQLiteDatabase sqLiteDatabase = this.openOrCreateDatabase("Arts",*MODE\_PRIVATE*,null);  
 //Veritabanından sorgu başlat  
 Cursor cursor = sqLiteDatabase.rawQuery("Select \* from arts",null);  
 //Verileri veritabanından al  
 int nameIx = cursor.getColumnIndex("artname");  
 int idIx = cursor.getColumnIndex("id");  
  
 //Veriler üzerinde dolaşımyap  
 while (cursor.moveToNext()){  
 String name = cursor.getString(nameIx);  
 int id = cursor.getInt(idIx);  
 //Art clasından obje oluştur ve constructora verileri ekle  
 Art art = new Art(name,id);  
 //artArrayListe işlemleri ekle  
 artArrayList.add(art);  
 }  
 cursor.close();  
 }  
 catch (Exception e){  
 e.printStackTrace();  
 }  
 }