****

**MCA SEM-III**

**Mobile Application Development LAB MANUAL(MCA-352)**

**SESSION 2021-23**



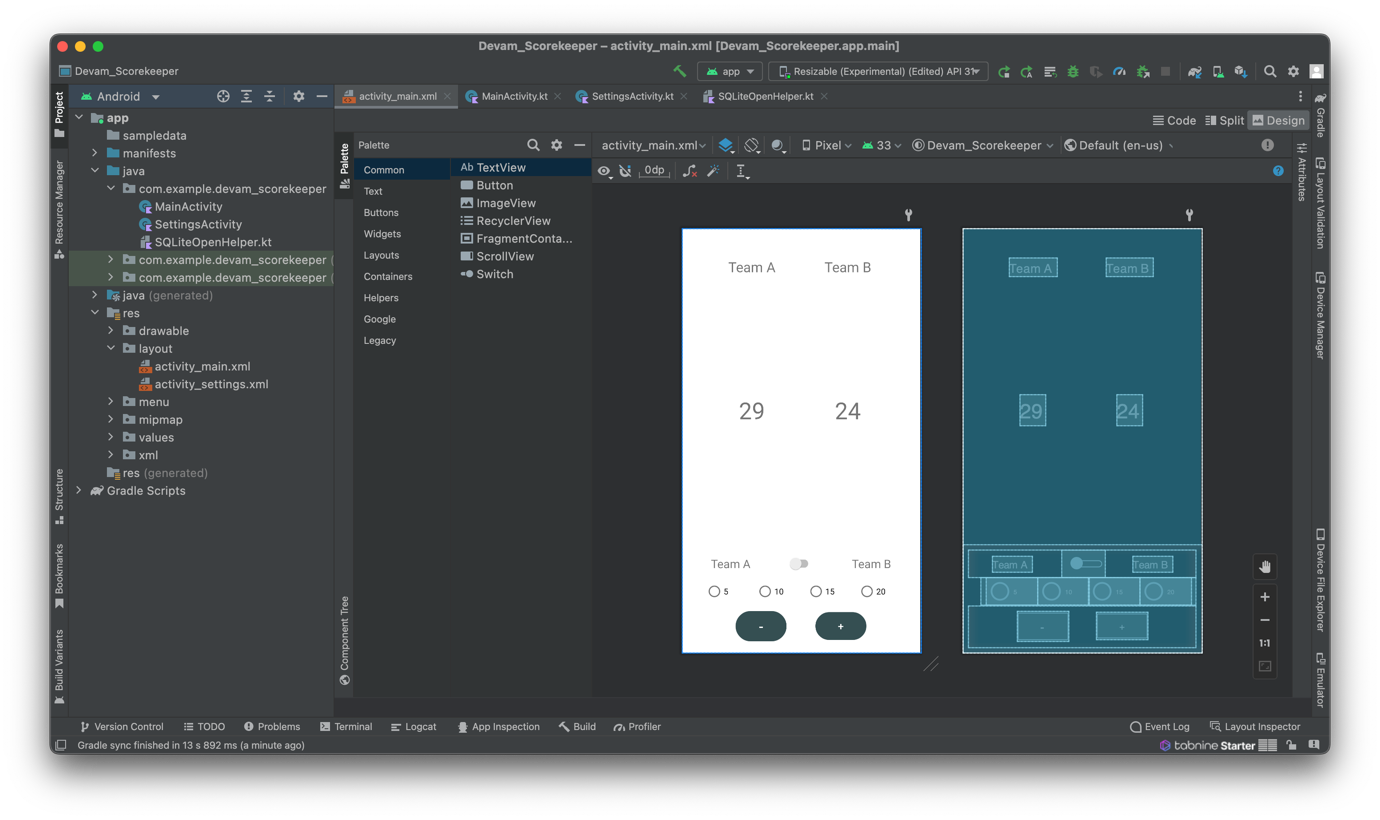
Submitted to: Submitted by

Ms.Shalini Sharma Gajendra Jangid

(Hod of MCA Department) Roll No:(21CRIXX608)

1. **Develop an application that uses GUI components, Font and Colours.**

<?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 for Team A-->  
<TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/name\_team\_a"  
 android:textSize="24sp"  
 android:layout\_marginStart="80dp"  
 android:layout\_marginTop="50dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
<!--TextView for Team B-->  
<TextView  
 android:id="@+id/textView2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/name\_team\_b"  
 android:textSize="24sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toEndOf="@+id/textView"  
 app:layout\_constraintTop\_toTopOf="@+id/textView" />  
  
<!--TextView for Score of Team A-->  
<TextView  
 android:id="@+id/textView3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/score\_team\_a"  
 android:textSize="40sp"  
 app:layout\_constraintBottom\_toTopOf="@+id/linear\_layout\_main"  
 app:layout\_constraintEnd\_toEndOf="@+id/textView"  
 app:layout\_constraintStart\_toStartOf="@+id/textView"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView" />  
  
<!--TextView for Score of Team B-->  
<TextView  
 android:id="@+id/textView4"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/score\_team\_b"  
 android:textSize="40sp"  
 app:layout\_constraintEnd\_toEndOf="@+id/textView2"  
 app:layout\_constraintStart\_toStartOf="@+id/textView2"  
 app:layout\_constraintTop\_toTopOf="@+id/textView3" />  
  
<!--Linear Layout for containing all UI Components-->  
<LinearLayout  
 android:id="@+id/linear\_layout\_main"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:orientation="vertical"  
 android:padding="10dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent">  
  
<!--Linear Layout for selecting the teams-->  
<LinearLayout  
 android:id="@+id/linear\_layout\_selector"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:orientation="horizontal">  
  
<!--TextView indicating Team A-->  
<TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="50dp"  
 android:text="@string/name\_team\_a"  
 android:textSize="20sp" />  
  
<!--Switch to choose between both teams-->  
<Switch  
 android:id="@+id/team\_switch"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:checked="false"  
 android:padding="14dp"  
 tools:ignore="UseSwitchCompatOrMaterialXml" />  
  
<!--TextView indicating Team B-->  
<TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="50dp"  
 android:text="@string/name\_team\_b"  
 android:textSize="20sp" />  
  
</LinearLayout>  
  
<!--Radio Buttons for pre-defined scores-->  
<RadioGroup  
 android:id="@+id/radio\_group"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:layout\_marginStart="20dp"  
 android:paddingHorizontal="10dp"  
 android:orientation="horizontal" >  
  
<RadioButton  
 android:id="@+id/radio\_button\_5"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/\_5"  
 android:layout\_weight="1" />  
  
<RadioButton  
 android:id="@+id/radio\_button\_10"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/\_10"  
 android:layout\_weight="1"/>  
  
<RadioButton  
 android:id="@+id/radio\_button\_15"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/\_15"  
 android:layout\_weight="1"/>  
  
<RadioButton  
 android:id="@+id/radio\_button\_20"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/\_20"  
 android:layout\_weight="1"/>  
  
</RadioGroup>  
  
<!--Linear Layout for increasing and decreasing buttons-->  
<LinearLayout  
 android:id="@+id/linear\_layout\_buttons"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:orientation="horizontal"  
 android:padding="10dp">  
  
<!-- Minus Score Button -->  
<Button  
 android:id="@+id/minus\_button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginEnd="50dp"  
 android:background="@drawable/bg\_button\_borderless"  
 android:text="@string/minus\_sign"  
 android:textSize="24sp"  
 tools:ignore="ButtonStyle" />  
  
<!-- Plus Score Button -->  
<Button  
 android:id="@+id/plus\_button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:background="@drawable/bg\_button\_borderless"  
 android:text="@string/plus\_sign"  
 android:textSize="18sp"  
 tools:ignore="ButtonStyle" />  
  
</LinearLayout>  
</LinearLayout>  
</androidx.constraintlayout.widget.ConstraintLayout>



**2. Write an android program to implement activity life cycle using toast messages with proper positioning.**

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.Gravity;

import android.widget.TextView;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

private TextView textView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

textView = findViewById(R.id.text\_view);

showToast("onCreate()");

}

@Override

protected void onStart() {

super.onStart();

showToast("onStart()");

}

@Override

protected void onResume() {

super.onResume();

showToast("onResume()");

}

@Override

protected void onPause() {

super.onPause();

showToast("onPause()");

}

@Override

protected void onStop() {

super.onStop();

showToast("onStop()");

}

@Override

protected void onDestroy() {

super.onDestroy();

showToast("onDestroy()");

}

private void showToast(String message) {

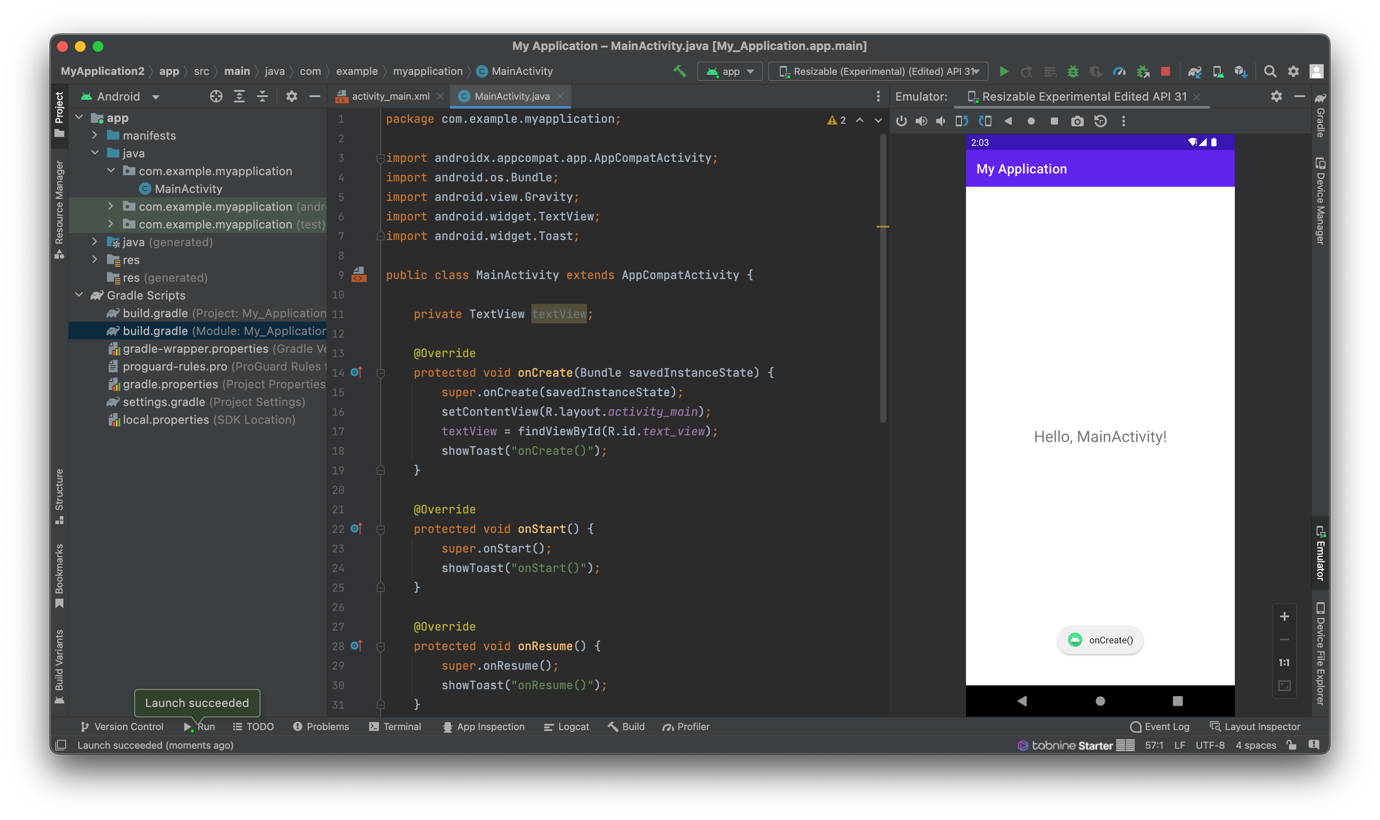
Toast toast = Toast.makeText(this, message, Toast.LENGTH\_SHORT);

toast.setGravity(Gravity.TOP, 0, 0);

toast.show();

}

}



**3.Develop an application that uses Layout Manager and event listeners.**

* In the activity\_main.xml layout file, add a RecyclerView with the android:id attribute set to "@+id/recycler\_view". Set the android:layout\_width and android:layout\_height attributes to "match\_parent".
* Create a new layout file named list\_item.xml with a TextView with an android:id attribute set to "@+id/text\_view". Set the android:layout\_width and android:layout\_height attributes to "wrap\_content".
* MainActivity.java:

import androidx.appcompat.app.AppCompatActivity;

import androidx.recyclerview.widget.LinearLayoutManager;

import androidx.recyclerview.widget.RecyclerView;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.TextView;

import android.widget.Toast;

import java.util.ArrayList;

import java.util.List;

public class MainActivity extends AppCompatActivity {

private RecyclerView recyclerView;

private List<String> items;

private Adapter adapter;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

items = new ArrayList<>();

items.add("Item 1");

items.add("Item 2");

items.add("Item 3");

items.add("Item 4");

items.add("Item 5");

recyclerView = findViewById(R.id.recycler\_view);

recyclerView.setLayoutManager(new LinearLayoutManager(this));

adapter = new Adapter();

recyclerView.setAdapter(adapter);

}

private class Adapter extends RecyclerView.Adapter<ViewHolder> implements View.OnClickListener {

@Override

public ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {

View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.list\_item, parent, false);

ViewHolder viewHolder = new ViewHolder(view);

viewHolder.itemView.setOnClickListener(this);

return viewHolder;

}

@Override

public void onBindViewHolder(ViewHolder holder, int position) {

holder.textView.setText(items.get(position));

}

@Override

public int getItemCount() {

return items.size();

}

@Override

public void onClick(View v) {

ViewHolder viewHolder = (ViewHolder) v.getTag();

int position = viewHolder.getAdapterPosition();

showToast("Item " + (position + 1) + " clicked!");

}

}

private class ViewHolder extends RecyclerView.ViewHolder {

TextView textView;

ViewHolder(View view) {

super(view);

textView = view.findViewById(R.id.text\_view);

view.setTag(this);

}

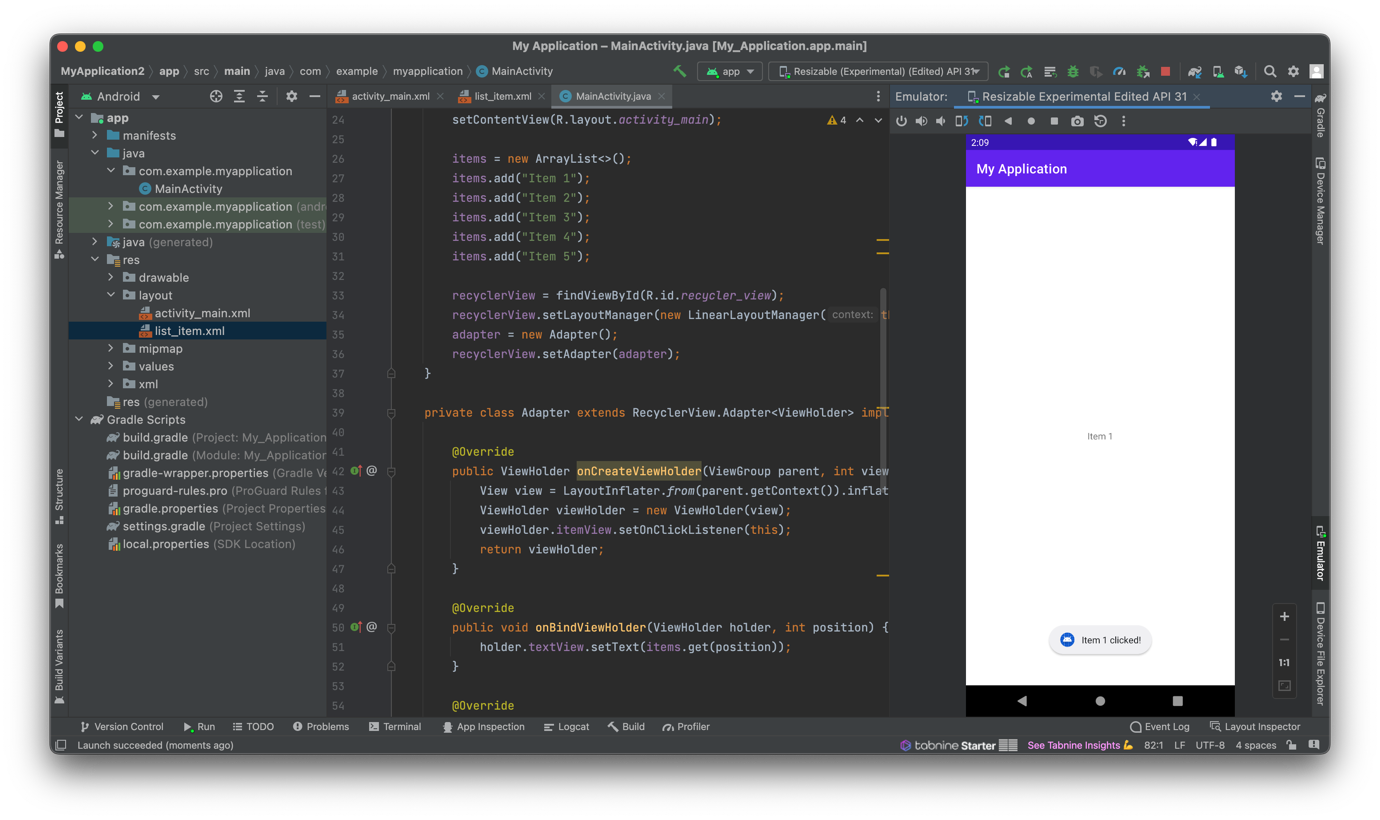
}

private void showToast(String message) {

Toast.makeText(this, message, Toast.LENGTH\_SHORT).show();

}

}



**4. Write an application that draws basic graphical primitives on the screen.**

* In the activity\_main.xml layout file, add a CustomView with the android:id attribute set to "@+id/custom\_view". Set the android:layout\_width and android:layout\_height attributes to "match\_parent".
* Create a new Java class named CustomView that extends the View class. In the onDraw() method, add code to draw a circle, a rectangle, and a line using a Paint object.
* MainActivity.java:

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

private CustomView customView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

customView = findViewById(R.id.custom\_view);

}

}

* CustomView.java:

import android.content.Context;

import android.graphics.Canvas;

import android.graphics.Color;

import android.graphics.Paint;

import android.util.AttributeSet;

import android.view.View;

public class CustomView extends View {

private Paint paint;

public CustomView(Context context) {

super(context);

init();

}

public CustomView(Context context, AttributeSet attrs) {

super(context, attrs);

init();

}

public CustomView(Context context, AttributeSet attrs, int defStyle) {

super(context, attrs, defStyle);

init();

}

private void init() {

paint = new Paint();

paint.setColor(Color.RED);

paint.setStyle(Paint.Style.FILL);

paint.setAntiAlias(true);

paint.setStrokeWidth(5);

}

@Override

protected void onDraw(Canvas canvas) {

super.onDraw(canvas);

// Draw a circle

canvas.drawCircle(200, 200, 100, paint);

// Draw a rectangle

paint.setColor(Color.BLUE);

canvas.drawRect(400, 100, 600, 300, paint);

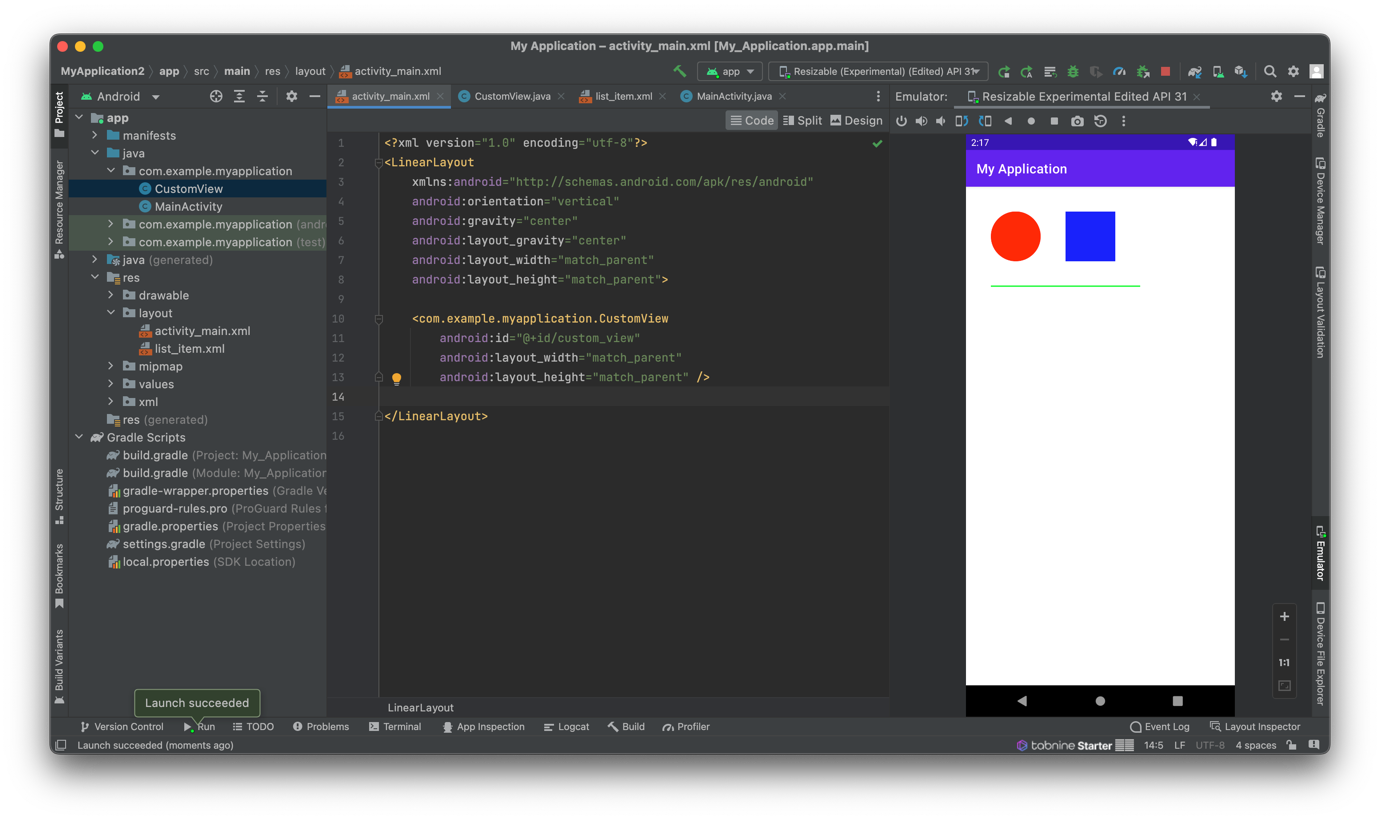
// Draw a line

paint.setColor(Color.GREEN);

canvas.drawLine(100, 400, 700, 400, paint);

}

}



**5. Write an application that basic graphical primitives and animations.**

* In the activity\_main.xml layout file, add a CustomView with the android:id attribute set to "@+id/custom\_view". Set the android:layout\_width and android:layout\_height attributes to "match\_parent".
* Create a new Java class named CustomView that extends the View class. In the onDraw() method, add code to draw a circle, a rectangle, and a line using a Paint object. Also, create a ValueAnimator object to animate the position and size of the circle.
* MainActivity.java:

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

private CustomView customView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

customView = findViewById(R.id.custom\_view);

}

}

* CustomView.java:

import android.animation.ValueAnimator;

import android.content.Context;

import android.graphics.Canvas;

import android.graphics.Color;

import android.graphics.Paint;

import android.util.AttributeSet;

import android.view.View;

import android.view.animation.LinearInterpolator;

public class CustomView extends View {

private Paint paint;

private float circleX, circleY, circleRadius;

private ValueAnimator animator;

public CustomView(Context context) {

super(context);

init();

}

public CustomView(Context context, AttributeSet attrs) {

super(context, attrs);

init();

}

public CustomView(Context context, AttributeSet attrs, int defStyle) {

super(context, attrs, defStyle);

init();

}

private void init() {

paint = new Paint();

paint.setColor(Color.RED);

paint.setStyle(Paint.Style.FILL);

paint.setAntiAlias(true);

paint.setStrokeWidth(5);

circleX = 200;

circleY = 200;

circleRadius = 100;

animator = ValueAnimator.ofFloat(0, 1);

animator.setDuration(3000);

animator.setRepeatCount(ValueAnimator.INFINITE);

animator.setRepeatMode(ValueAnimator.REVERSE);

animator.setInterpolator(new LinearInterpolator());

animator.addUpdateListener(new ValueAnimator.AnimatorUpdateListener() {

@Override

public void onAnimationUpdate(ValueAnimator animation) {

float value = (float) animation.getAnimatedValue();

circleX = getWidth() \* value;

circleY = getHeight() \* value;

circleRadius = 100 + 50 \* value;

invalidate();

}

});

animator.start();

}

@Override

protected void onDraw(Canvas canvas) {

super.onDraw(canvas);

// Draw a circle

canvas.drawCircle(circleX, circleY, circleRadius, paint);

// Draw a rectangle

paint.setColor(Color.BLUE);

canvas.drawRect(400, 100, 600, 300, paint);

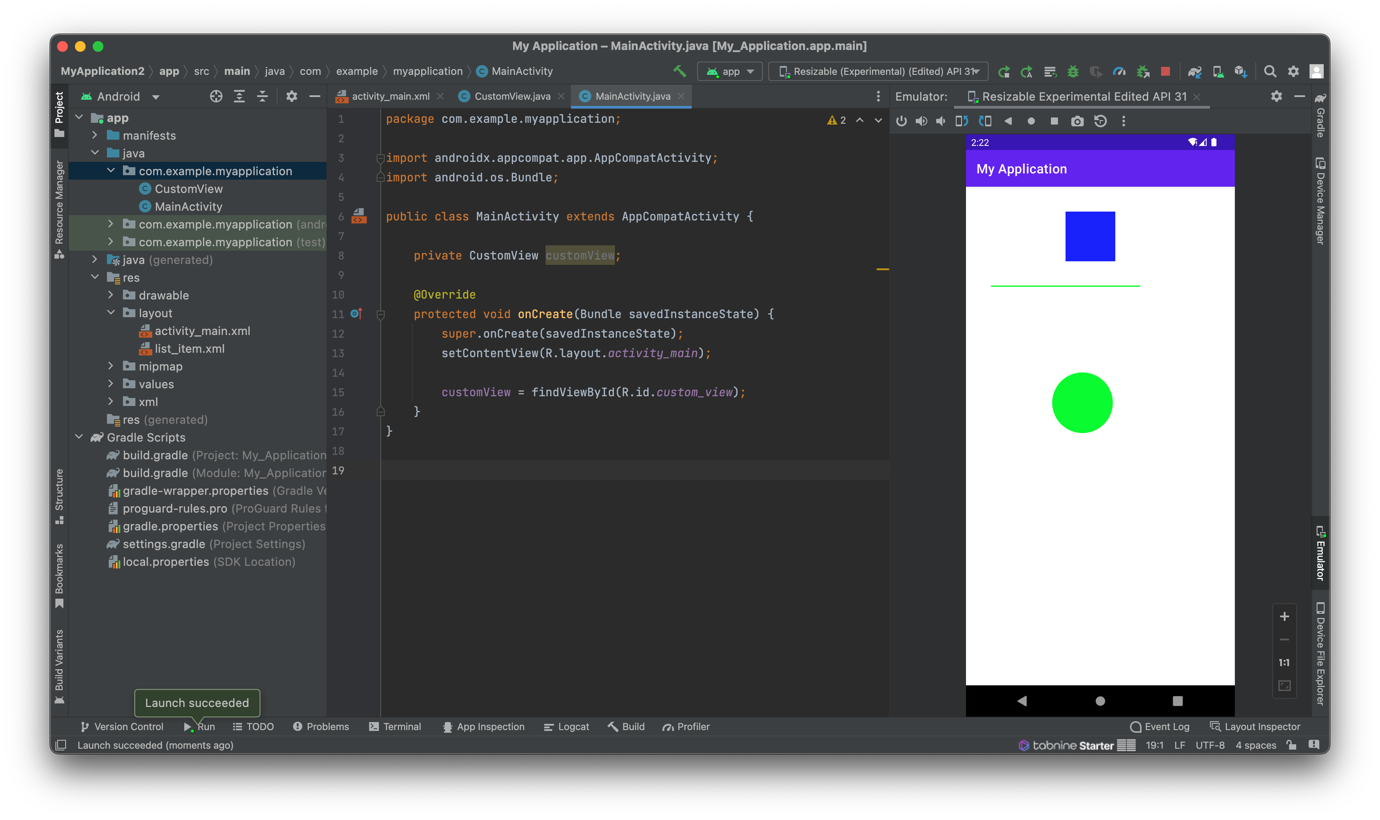
// Draw a line

paint.setColor(Color.GREEN);

canvas.drawLine(100, 400, 700, 400, paint);

}

}



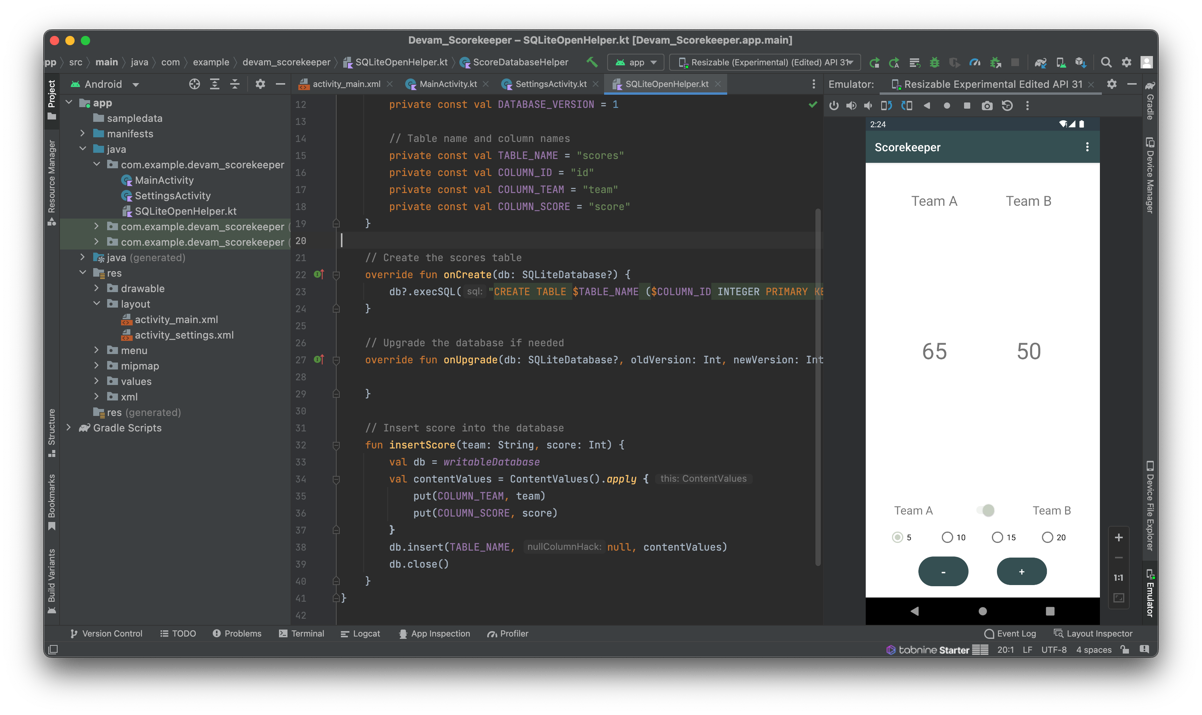
**6. Develop an application that makes use of databases.**

import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.widget.Button  
import com.example.devam\_scorekeeper.ScoreDatabaseHelper

* MainActivity.kt:

class SettingsActivity : AppCompatActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.activity\_settings)  
  
 val saveScoresButton = findViewById<Button>(R.id.saveButton)  
  
 // Saving the scores in the local database  
 saveScoresButton.setOnClickListener {  
 val scoreDatabaseHelper = ScoreDatabaseHelper(this)  
 if (selectedTeam == "A") {  
 scoreDatabaseHelper.insertScore("Team A", scoreTeamA)  
 } else {  
 scoreDatabaseHelper.insertScore("Team B", scoreTeamB)  
 }  
 }  
 }  
}

* SQLiteOpenHelper.kt:  
    
  import android.content.ContentValues  
  import android.content.Context  
  import android.database.sqlite.SQLiteDatabase  
  import android.database.sqlite.SQLiteOpenHelper  
    
  class ScoreDatabaseHelper(context: Context) : SQLiteOpenHelper(context, DATABASE\_NAME, null, DATABASE\_VERSION) {  
    
   companion object {  
   private const val DATABASE\_NAME = "score.db"  
   private const val DATABASE\_VERSION = 1  
    
   // Table name and column names  
   private const val TABLE\_NAME = "scores"  
   private const val COLUMN\_ID = "id"  
   private const val COLUMN\_TEAM = "team"  
   private const val COLUMN\_SCORE = "score"  
   }  
    
   // Create the scores table  
   override fun onCreate(db: SQLiteDatabase?) {  
   db?.execSQL("CREATE TABLE $TABLE\_NAME ($COLUMN\_ID INTEGER PRIMARY KEY AUTOINCREMENT, $COLUMN\_TEAM TEXT, $COLUMN\_SCORE INTEGER)")  
   }  
    
   // Upgrade the database if needed  
   override fun onUpgrade(db: SQLiteDatabase?, oldVersion: Int, newVersion: Int) {  
    
   }  
    
   // Insert score into the database  
   fun insertScore(team: String, score: Int) {  
   val db = writableDatabase  
   val contentValues = ContentValues().apply {  
   put(COLUMN\_TEAM, team)  
   put(COLUMN\_SCORE, score)  
   }  
   db.insert(TABLE\_NAME, null, contentValues)  
   db.close()  
   }  
  }



**7. Develop an application that makes use of Notification Manager.**

* In the activity\_main.xml layout file, add a Button with the android:id attribute set to "@+id/button". Set the android:layout\_width and android:layout\_height attributes to "wrap\_content".
* MainActivity.java:

public class MainActivity extends AppCompatActivity {

private static final int NOTIFICATION\_ID = 1;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

Button button = findViewById(R.id.button);

button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

showNotification();

}

});

}

private void showNotification() {

NotificationManager notificationManager = (NotificationManager) getSystemService(Context.NOTIFICATION\_SERVICE);

if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.O) {

NotificationChannel channel = new NotificationChannel("channel\_id", "channel\_name", NotificationManager.IMPORTANCE\_DEFAULT);

notificationManager.createNotificationChannel(channel);

}

NotificationCompat.Builder builder = new NotificationCompat.Builder(this, "channel\_id")

.setSmallIcon(R.drawable.ic\_notification)

.setContentTitle("Notification Title")

.setContentText("This is a notification message.")

.setPriority(NotificationCompat.PRIORITY\_DEFAULT);

notificationManager.notify(NOTIFICATION\_ID, builder.build());

}

}

* Ic\_notification.xml:

<vector xmlns:android="http://schemas.android.com/apk/res/android"

android:width="24dp"

android:height="24dp"

android:viewportWidth="24.0"

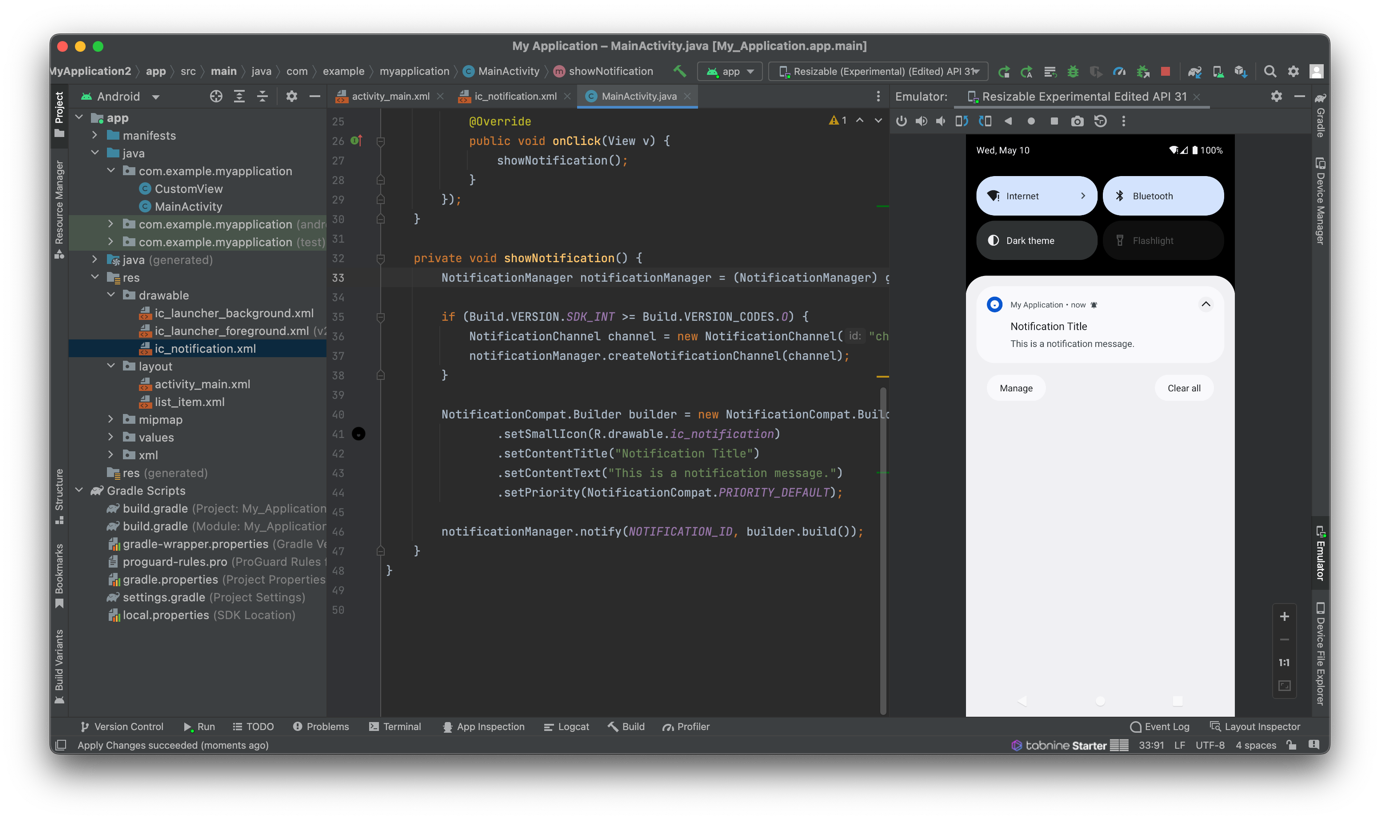
android:viewportHeight="24.0">

<path

android:fillColor="#000000"

android:pathData="M12,2C6.48,2 2,6.48 2,12s4.48,10 10,10 10,-4.48 10,-10S17.52,2 12,2zM12,16.5c-1.38,0 -2.5,-1.12 -2.5,-2.5s1.12,-2.5 2.5,-2.5 2.5,1.12 2.5,2.5 -1.12,2.5 -2.5,2.5zM12,7c-1.66,0 -3,1.34 -3,3s1.34,3 3,3 3,-1.34 3,-3 -1.34,-3 -3,-3z" />

</vector>



**8. Develop a native application that uses GPS location information.**

* MainActivity.java:

public class MainActivity extends AppCompatActivity implements LocationListener {

private LocationManager locationManager;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

locationManager = (LocationManager) getSystemService(Context.LOCATION\_SERVICE);

if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_FINE\_LOCATION) == PackageManager.PERMISSION\_GRANTED) {

locationManager.requestLocationUpdates(LocationManager.GPS\_PROVIDER, 0, 0, this);

} else {

ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.ACCESS\_FINE\_LOCATION}, 1);

}

}

@Override

public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {

if (requestCode == 1 && grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED) {

if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_FINE\_LOCATION) == PackageManager.PERMISSION\_GRANTED) {

locationManager.requestLocationUpdates(LocationManager.GPS\_PROVIDER, 0, 0, this);

}

}

}

@Override

public void onLocationChanged(Location location) {

double latitude = location.getLatitude();

double longitude = location.getLongitude();

Toast.makeText(this, "Latitude: " + latitude + ", Longitude: " + longitude, Toast.LENGTH\_SHORT).show();

}

@Override

public void onProviderDisabled(String provider) {

Toast.makeText(this, "GPS is disabled.", Toast.LENGTH\_SHORT).show();

}

@Override

public void onProviderEnabled(String provider) {

Toast.makeText(this, "GPS is enabled.", Toast.LENGTH\_SHORT).show();

}

@Override

public void onStatusChanged(String provider, int status, Bundle extras) {

}

}

