**Main.xml**

<?xml version="1.0" encoding="utf-8"?>  
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <LinearLayout  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:padding="16dp">  
  
 <!-- Current Month TextView -->  
 <TextView  
 android:id="@+id/currentMonthText"  
 android:textSize="20sp"  
 android:gravity="center"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" />  
  
 <!-- Previous Button -->  
 <Button  
 android:id="@+id/prevButton"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:text="Previous" />  
  
 <!-- Next Button -->  
 <Button  
 android:id="@+id/nextButton"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:text="Next" />  
  
 <!-- Month Calendar Layout -->  
 <LinearLayout  
 android:id="@+id/month\_layout"  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content">  
 </LinearLayout>  
  
 </LinearLayout>  
</ScrollView>

**Main.java**

package com.example.practice1;  
  
import android.os.Bundle;  
import android.view.Gravity;  
import android.widget.Button;  
import android.widget.GridLayout;  
import android.widget.LinearLayout;  
import android.widget.TextView;  
import androidx.appcompat.app.AppCompatActivity;  
  
import java.util.Calendar;  
  
public class MainActivity extends AppCompatActivity {  
  
 private LinearLayout monthLayout;  
 private TextView currentMonthText;  
 private int currentMonthIndex = 0; // 0 for January, 11 for December  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 monthLayout = findViewById(R.id.month\_layout);  
 currentMonthText = findViewById(R.id.currentMonthText);  
  
 Button prevButton = findViewById(R.id.prevButton);  
 Button nextButton = findViewById(R.id.nextButton);  
  
 // Display the current month initially  
 displayMonth(currentMonthIndex);  
  
 // Button actions for changing months  
 prevButton.setOnClickListener(v -> displayMonth(--currentMonthIndex));  
 nextButton.setOnClickListener(v -> displayMonth(++currentMonthIndex));  
 }  
  
 private void displayMonth(int monthIndex) {  
 monthLayout.removeAllViews(); // Clear the previous month's calendar  
  
 String[] months = getResources().getStringArray(R.array.*months*);  
 String[] weekdays = getResources().getStringArray(R.array.*weekdays*);  
  
 // Display month name  
 currentMonthText.setText(months[monthIndex]);  
  
 Calendar calendar = Calendar.*getInstance*();  
 calendar.set(Calendar.*MONTH*, monthIndex);  
  
 // Get number of days and starting day of the week  
 int daysInMonth = calendar.getActualMaximum(Calendar.*DAY\_OF\_MONTH*);  
 int firstDayOfWeek = calendar.get(Calendar.*DAY\_OF\_WEEK*); // 1 = Sunday, 7 = Saturday  
  
 // Month title  
 TextView monthTitle = new TextView(this);  
 monthTitle.setText(months[monthIndex]);  
 monthTitle.setTextSize(20);  
 monthTitle.setGravity(Gravity.*CENTER*);  
 monthLayout.addView(monthTitle);  
  
 // Set up the GridLayout for the days of the month  
 GridLayout gridLayout = new GridLayout(this);  
 gridLayout.setRowCount(6);  
 gridLayout.setColumnCount(7);  
  
 // Weekday headers  
 for (String weekday : weekdays) {  
 TextView weekdayTextView = new TextView(this);  
 weekdayTextView.setText(weekday);  
 weekdayTextView.setGravity(Gravity.*CENTER*);  
 weekdayTextView.setTextColor(getResources().getColor(android.R.color.*darker\_gray*));  
 gridLayout.addView(weekdayTextView);  
 }  
  
 // Add empty cells for previous month's days  
 for (int i = 1; i < firstDayOfWeek; i++) {  
 gridLayout.addView(createEmptyCell());  
 }  
  
 // Add the days of the current month  
 for (int day = 1; day <= daysInMonth; day++) {  
 TextView dayTextView = new TextView(this);  
 dayTextView.setText(String.*valueOf*(day));  
 dayTextView.setGravity(Gravity.*CENTER*);  
 dayTextView.setPadding(10, 10, 10, 10);  
 dayTextView.setTextColor(firstDayOfWeek == 1 || firstDayOfWeek == 7 ?  
 getResources().getColor(android.R.color.*holo\_red\_dark*) :  
 getResources().getColor(android.R.color.*black*));  
 gridLayout.addView(dayTextView);  
  
 // Move to the next weekday  
 if (++firstDayOfWeek > 7) firstDayOfWeek = 1;  
 }  
  
 monthLayout.addView(gridLayout);  
 }  
  
 private TextView createEmptyCell() {  
 TextView emptyCell = new TextView(this);  
 emptyCell.setText("");  
 return emptyCell;  
 }  
}

strings.xml

<resources>  
 <!-- App Name -->  
 <string name="app\_name">Year Calendar</string>  
  
 <!-- Months Array -->  
 <string-array name="months">  
 <item>January</item>  
 <item>February</item>  
 <item>March</item>  
 <item>April</item>  
 <item>May</item>  
 <item>June</item>  
 <item>July</item>  
 <item>August</item>  
 <item>September</item>  
 <item>October</item>  
 <item>November</item>  
 <item>December</item>  
 </string-array>  
  
 <!-- Weekdays -->  
 <string-array name="weekdays">  
 <item>Sun</item>  
 <item>Mon</item>  
 <item>Tue</item>  
 <item>Wed</item>  
 <item>Thu</item>  
 <item>Fri</item>  
 <item>Sat</item>  
 </string-array>  
</resources>