****

**İsim Soyisim: Salih Karakaya**

**Öğrenci No: 191816775**

1. **Ödeve ait isterler ve bu isterlerin açıklanması**

Bu ödevde, Android Studio kullanarak basit bir hesap makinesi uygulaması yapmamız gerekmektedir. Uygulama, kullanıcıdan iki sayı alçak ve dört işlem yapmamı sağlayacak butonlar içerecek. İşlemin sonucunu ekranda gösterecek bir TextView de olacak.

**İsterler;**

**Kullanıcı Arayüzü (UI)**

* İki adet EditText bileşeni: Kullanıcının işlem yapmak istediği iki tam sayıyı girmesi için kullanılacaktır.
* Dört adet buton: Kullanıcının yapmak istediği işlemi seçmesini sağlar (Toplama, Çıkarma, Çarpma, Bölme).
* Bir adet TextView: İşlem sonucunun ekranda görüntülenmesi için kullanılacaktır.

**Giriş Kontrolleri**

* Kullanıcı iki sayıyı girmeden işlem yapmaya çalışırsa, ekrana bir hata mesajı gösterilmelidir.
* Bölme işlemi sırasında sıfıra bölme hatası önlenmeli ve kullanıcıya uygun bir hata mesajı verilmelidir.

**İşlem Mantığı**

* Dört işlem butonu tıklandığında, ilgili işlem harici bir fonksiyon (metot) tarafından gerçekleştirilmeli ve sonuç TextView bileşeni içerisinde gösterilmelidir.
* Matematiksel işlemler switch-case veya if-else yapıları kullanılarak işlenmelidir.

**Kod Yapısı ve Optimizasyon**

* Kod modüler ve okunabilir bir yapıda olmalıdır.
* İşlemler hesapla() adlı bir metot içinde yapılmalıdır. Böylece kod tekrarı önlenerek temiz bir yapı oluşturulmalıdır.
* Gerekli yorum satırları eklenerek kod açıklanmalıdır.

1. **Var ise XML kodlarının metin olarak yapıştırılması ve açıklanması**
2. <?xml version="1.0" encoding="utf-8"?>  
   <androidx.constraintlayout.widget.ConstraintLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:tools="http://schemas.android.com/tools"  
    android:layout\_width="match\_parent"  
    android:layout\_height="match\_parent"  
    xmlns:app="http://schemas.android.com/apk/res-auto"  
    android:orientation="vertical"  
    android:gravity="center"  
    tools:context=".MainActivity">  
    <!-- Acıklama Satırı Denemesi -->  
     
     
     
    <TextView  
    android:id="@+id/sonucTxt"  
    android:layout\_width="wrap\_content"  
    android:layout\_height="wrap\_content"  
    app:layout\_constraintTop\_toTopOf="parent"  
    app:layout\_constraintBottom\_toBottomOf="parent"  
    app:layout\_constraintStart\_toStartOf="parent"  
    app:layout\_constraintEnd\_toEndOf="parent"  
    android:textSize="25sp"  
    android:layout\_marginTop="20dp"  
    android:textAlignment="center"  
    android:textStyle="bold|italic"  
    app:layout\_constraintVertical\_bias="0.2"  
    />  
     
    <Button  
    android:id="@+id/toplaBtn"  
    android:layout\_width="0dp"  
    android:layout\_height="wrap\_content"  
    app:layout\_constraintStart\_toStartOf="@id/number2"  
    app:layout\_constraintTop\_toTopOf="parent"  
    app:layout\_constraintEnd\_toEndOf="@id/number2"  
    app:layout\_constraintBottom\_toBottomOf="parent"  
    app:layout\_constraintVertical\_bias="0.6"  
    app:layout\_constraintHorizontal\_bias="0"  
    app:layout\_constraintWidth\_percent="0.3"  
    android:text="+"  
    android:textStyle="bold"  
    android:textSize="25sp"  
    />  
    <Button  
    android:id="@+id/cikarBtn"  
    android:layout\_width="0dp"  
    android:layout\_height="wrap\_content"  
    app:layout\_constraintStart\_toStartOf="@id/number2"  
    app:layout\_constraintTop\_toTopOf="parent"  
    app:layout\_constraintEnd\_toEndOf="@id/number2"  
    app:layout\_constraintBottom\_toBottomOf="parent"  
    app:layout\_constraintVertical\_bias="0.6"  
    app:layout\_constraintHorizontal\_bias="1"  
    app:layout\_constraintWidth\_percent="0.3"  
    android:text="-"  
    android:textStyle="bold"  
    android:textSize="25sp"  
    />  
    <Button  
    android:id="@+id/carpBtn"  
    android:layout\_width="0dp"  
    android:layout\_height="wrap\_content"  
    app:layout\_constraintStart\_toStartOf="@id/number2"  
    app:layout\_constraintTop\_toTopOf="@id/number2"  
    app:layout\_constraintEnd\_toEndOf="@id/number2"  
    app:layout\_constraintBottom\_toBottomOf="parent"  
    app:layout\_constraintVertical\_bias="0.6"  
    app:layout\_constraintHorizontal\_bias="0"  
    app:layout\_constraintWidth\_percent="0.3"  
    android:text="\*"  
    android:textStyle="bold"  
    android:textSize="25sp"  
    />  
    <Button  
    android:id="@+id/bolBtn"  
    android:layout\_width="0dp"  
    android:layout\_height="wrap\_content"  
    app:layout\_constraintStart\_toStartOf="@id/number2"  
    app:layout\_constraintTop\_toTopOf="@id/number2"  
    app:layout\_constraintEnd\_toEndOf="@id/number2"  
    app:layout\_constraintBottom\_toBottomOf="parent"  
    app:layout\_constraintVertical\_bias="0.6"  
    app:layout\_constraintHorizontal\_bias="1"  
    app:layout\_constraintWidth\_percent="0.3"  
    android:text="/"  
    android:textStyle="bold"  
    android:textSize="25sp"  
    />  
    <EditText  
    android:id="@+id/number1"  
    android:layout\_width="0dp"  
    android:layout\_height="wrap\_content"  
    app:layout\_constraintStart\_toStartOf="parent"  
    app:layout\_constraintTop\_toTopOf="parent"  
    app:layout\_constraintEnd\_toEndOf="parent"  
    app:layout\_constraintBottom\_toBottomOf="parent"  
    app:layout\_constraintVertical\_bias="0.30"  
    app:layout\_constraintWidth\_percent="0.7"  
    android:hint="Ilk Sayı"  
    android:inputType="number"/>  
    <EditText  
    android:id="@+id/number2"  
    android:layout\_width="0dp"  
    android:layout\_height="wrap\_content"  
    app:layout\_constraintStart\_toStartOf="@+id/number1"  
    app:layout\_constraintTop\_toBottomOf="@+id/number1"  
    app:layout\_constraintEnd\_toEndOf="@id/number1"  
    app:layout\_constraintBottom\_toBottomOf="parent"  
    app:layout\_constraintVertical\_bias="0.05"  
    app:layout\_constraintWidth\_percent="0.7"  
    android:hint="Ikinci Sayı"  
    android:inputType="number"/>  
     
    <!--<ImageView  
    android:id="@+id/gaziImg"  
    android:layout\_width="0dp"  
    android:layout\_height="0dp"  
    app:layout\_constraintStart\_toStartOf="parent"  
    app:layout\_constraintEnd\_toEndOf="parent"  
    app:layout\_constraintTop\_toTopOf="parent"  
    app:layout\_constraintBottom\_toBottomOf="parent"  
    android:src="@drawable/gazi\_logo"  
    app:layout\_constraintHeight\_percent="0.25"  
    app:layout\_constraintWidth\_percent="0.45"  
    app:layout\_constraintVertical\_bias="0.25"  
    />-->  
     
   </androidx.constraintlayout.widget.ConstraintLayout>

Bu XML kodu, Android Studio’da **ConstraintLayout** kullanılarak oluşturulmuş bir basit hesap makinesi arayüzünü tanımlar.

**Genel Yapı ve Ana Layout**

XML dosyası, ConstraintLayout kullanılarak oluşturulmuştur. Bu yapı, bileşenlerin birbirine olan konumunu belirleyerek esnek bir tasarım sağlar.

* match\_parent: Tüm ekranı kaplamasını sağlar.
* android:gravity="center": İçeriğin ekranda ortalanmasını sağlar.

**Sonuç Gösterme Alanı (TextView)**

Hesaplanan sonucu ekranda göstermek için kullanılır.

* android:textSize="25sp" → Yazı boyutunu 25sp olarak ayarlar.
* android:textStyle="bold|italic" → Yazıyı kalın ve italik yapar.
* app:layout\_constraintVertical\_bias="0.2" → Ekranın üst kısmına hizalanmasını sağlar.

**Sayı Giriş Alanları (EditText)**

Kullanıcının birinci sayıyı girmesi için kullanılır.

* android:hint="Ilk Sayı" → Kullanıcıya girilecek değerin ipucu olarak gösterilmesini sağlar.
* android:inputType="number" → Kullanıcının yalnızca sayısal veri girmesini sağlar.
* app:layout\_constraintWidth\_percent="0.7" → Ekranın %70'ini kaplamasını sağlar.

**İşlem Butonları (Button)**

Her buton, belirli bir işlemi gerçekleştirmek için tasarlanmıştır. Kullanıcının toplama işlemini seçmesini sağlar.

* android:text="+" → Butonun üzerindeki yazıyı belirler.
* app:layout\_constraintWidth\_percent="0.3" → Butonun ekranın %30’unu kaplamasını sağlar.
* app:layout\_constraintVertical\_bias="0.6" → Butonun ekranda hizalanmasını sağlar.

Diğer butonlar (Çıkarma (-), Çarpma (\*), Bölme (/)) benzer özelliklere sahiptir.

1. **XML kodları sonucunda ortaya çıkan ekran tasarımının ekran görüntüsü**

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

1. **Var ise Java kodlarının (yorum satırına ek olarak) metin halinde yapıştırılması ve bloklar halinde açıklanması**

**package com.example.mobilprogramlamahafta1;  
  
import android.os.Bundle;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
 // onCreate methodu ve xml dosyamı tanıtma  
 TextView sonuctxt;  
 EditText number1, number2;  
 Button toplabtn, cikarbtn, carpbtn, bolbtn;**

* MainActivity, AppCompatActivity sınıfından türetilmiştir. Bu, uygulamanın bir aktivite olarak çalışmasını sağlar.
* TextView sonuctxt → İşlem sonucunu göstermek için kullanılan metin bileşeni.
* EditText number1, number2 → Kullanıcının girdiği iki tam sayı için giriş kutuları.
* Button toplabtn, cikarbtn, carpbtn, bolbtn → Kullanıcının seçtiği işlemi gerçekleştiren düğmeler.

**@Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);**

* onCreate() → Uygulama başlatıldığında çalışan metottur.
* setContentView(R.layout.activity\_main) → XML dosyasında oluşturulan arayüzü (activity\_main.xml) uygulamaya bağlar.

**// findViewById() metoduyla, XML’de tanımlanan bileşenler Java koduyla ilişkilendirilmiştir  
 sonuctxt = findViewById(R.id.*sonucTxt*);  
 number1 = findViewById(R.id.*number1*);  
 number2 = findViewById(R.id.*number2*);  
 toplabtn = findViewById(R.id.*toplaBtn*);  
 cikarbtn = findViewById(R.id.*cikarBtn*);  
 carpbtn = findViewById(R.id.*carpBtn*);  
 bolbtn = findViewById(R.id.*bolBtn*);  
  
 // setOnClickListener() → Butona tıklandığında hesapla() metodunu çağırır.  
 toplabtn.setOnClickListener(view -> hesapla("+"));  
 cikarbtn.setOnClickListener(view -> hesapla("-"));  
 carpbtn.setOnClickListener(view -> hesapla("\*"));  
 bolbtn.setOnClickListener(view -> hesapla("/"));  
 }  
 void hesapla(String islem){  
 String s1 = number1.getText().toString();  
 String s2 = number2.getText().toString();**

* number1.getText().toString() → Kullanıcının birinci sayı girişini alır.
* number2.getText().toString() → Kullanıcının ikinci sayı girişini alır.

**if (s1.isEmpty() || s2.isEmpty()) {  
 sonuctxt.setText("Lutfen Iki Sayi Giriniz");  
 return;  
 }  
 //**  **Kullanıcının girdiği sayılar int tipine çevrilir.  
 int num1 = Integer.*parseInt*(s1);  
 int num2 = Integer.*parseInt*(s2);  
 int sonuc = 0;  
  
 switch (islem) {  
 case "+":  
 sonuc = num1 + num2;  
 break;  
 case "-":  
 sonuc = num1 - num2;  
 break;  
 case "\*":  
 sonuc = num1 \* num2;  
 break;  
 case "/":  
 if (num2 == 0) {  
 sonuctxt.setText("Hata: Sifira Bölme!");  
 return;  
 }  
 sonuc = num1 / num2;  
 break;  
 }  
 // İşlem tamamlandıktan sonra, sonuç TextView içinde gösterilir  
 sonuctxt.setText("Sonuç: " + sonuc);  
 }  
}**

1. **Kullanılan ek teknolojiler ve yapıların belirtilmesi ve açıklanması**

Bu ödev kapsamında Android Studio kullanılarak basit bir hesap makinesi uygulaması geliştirilmiştir. Uygulamanın tasarım ve işlevselliğini oluşturmak için aşağıdaki teknolojiler ve yapılar kullanılmıştır:

**1. Android Studio ve Java Programlama Dili**

* Android Studio, Android uygulamaları geliştirmek için kullanılan resmi geliştirme ortamıdır (IDE). XML ile arayüz tasarımı, Java ile de uygulama mantığı kodlanmıştır.
* Java, Android uygulamalarında yaygın olarak kullanılan nesne yönelimli bir programlama dili olup, bu projede de uygulamanın iş mantığı Java ile geliştirilmiştir.

**2. ConstraintLayout ile Esnek Arayüz Tasarımı**

* Uygulama arayüzü, bileşenlerin birbirine göre hizalanmasını sağlayan ConstraintLayout kullanılarak tasarlanmıştır.
* Avantajları:  
   Ekran boyutlarına duyarlı, responsive tasarım sunar.  
   Daha düzenli ve optimize edilmiş bir arayüz sağlar.

**3. UI Bileşenleri (TextView, EditText, Button)**

* TextView: İşlem sonucunun kullanıcıya gösterilmesini sağlar.
* EditText: Kullanıcının sayıları girmesine olanak tanır.
* Button: Dört işlem butonları (Toplama, Çıkarma, Çarpma, Bölme) sayesinde kullanıcı işlemi seçer.

Bu bileşenler, findViewById() metodu ile Java koduna bağlanmıştır.

**4. Olay Yönetimi (Event Handling)**

* Kullanıcı butonlara tıkladığında OnClickListener kullanılarak ilgili işlem gerçekleştirilir.
* Lambda ifadeleri (view -> hesapla("+") kullanılarak kodun daha okunaklı olması sağlanmıştır.

**5. Hata Kontrolleri**

* Kullanıcının boş giriş yapması veya sıfıra bölme hatası yapması durumları kontrol edilmiştir.
* Boş giriş kontrolü: Kullanıcı her iki sayıyı da girmeden işlem yapmaya çalışırsa, TextView üzerinde uyarı mesajı gösterilir.
* Sıfıra bölme kontrolü: Eğer ikinci sayı 0 ise, hata mesajı ekranda gösterilir ve işlem sonlandırılır.

**6. Fonksiyonel Kod Tasarımı (Modüler Yapı)**

* hesapla() metodu sayesinde, kod tekranı önlenerek her işlem için tek bir metot üzerinden hesaplama yapılmıştır.
* switch-case yapısı ile hangi işlemin yapılacağı kontrol edilmiştir.
* Kodun modüler olması, ileride geliştirme yapılmasını kolaylaştıracaktır.

**Sonuç**

Bu projede, Android Studio, Java, ConstraintLayout, temel UI bileşenleri ve hata yönetimi gibi teknolojiler kullanılmıştır. Uygulamanın daha temiz, hatasız ve kullanıcı dostu olması için fonksiyonel bir yaklaşım benimsenmiş, işlemler tek bir metot içinde yönetilerek kod tekrarından kaçınılmıştır.