**MainActivity.java**

package com.example.hayatsinyalim;  
import android.Manifest;  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.content.SharedPreferences;  
import android.content.pm.PackageManager;  
import android.net.Uri;  
import android.os.Bundle;  
import android.provider.Settings;  
import android.view.View;  
import android.widget.Button;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
import androidx.core.app.NotificationManagerCompat;  
import androidx.core.content.ContextCompat;  
  
import com.google.firebase.auth.FirebaseAuth;  
import com.google.firebase.auth.FirebaseUser;  
import com.google.firebase.database.DataSnapshot;  
import com.google.firebase.database.DatabaseError;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
import com.google.firebase.database.ValueEventListener;  
public class MainActivity extends AppCompatActivity {  
 Button btnGiris, btnKayıt;  
 String adSoyad;  
 String Yetki;  
 String Durum;  
 private static final int *PERMISSION\_REQUEST\_CODE* = 100;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 btnGiris = findViewById(R.id.*loginButton*);  
 btnKayıt = findViewById(R.id.*registerButton*);  
 checkLocationPermission();  
 SharedPreferences preferences = getSharedPreferences("ilkGirisDurumu", *MODE\_PRIVATE*);  
 boolean hoşgeldinGösterildiMi = preferences.getBoolean("soru", false);  
 if (!hoşgeldinGösterildiMi) {  
 new AlertDialog.Builder(this)  
 .setTitle("Seslere Erişim İzni")  
 .setMessage("Uygulamanın seslere erişim izni gerekiyor.\nİzin vermek ister misiniz?\nYönlendirildiğiniz sayfasa 'Seslere İzin Ver' Seçeneğini seçmeniz gerekir.")  
 .setPositiveButton("Evet", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
 Intent intent = new Intent(Settings.*ACTION\_APP\_NOTIFICATION\_SETTINGS*);  
 intent.putExtra(Settings.*EXTRA\_APP\_PACKAGE*, getPackageName());  
 startActivity(intent);  
 SharedPreferences.Editor editor = preferences.edit();  
 editor.putBoolean("soru", true);  
 editor.apply();  
 }  
 })  
 .setNegativeButton("Hayır", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which) {  
 finish();  
 }  
 })  
 .show();  
 }  
 FirebaseAuth mAuth = FirebaseAuth.*getInstance*();  
 mAuth.addAuthStateListener(new FirebaseAuth.AuthStateListener() {  
 @Override  
 public void onAuthStateChanged(@NonNull FirebaseAuth firebaseAuth) {  
 FirebaseUser currentUser = firebaseAuth.getCurrentUser();  
 if (currentUser != null) {  
 FirebaseDatabase database = FirebaseDatabase.*getInstance*();  
 DatabaseReference ref = database.getReference("users").child(mAuth.getUid());  
 ref.addListenerForSingleValueEvent(new ValueEventListener() {  
 @Override  
 public void onDataChange(DataSnapshot dataSnapshot) {  
 if (dataSnapshot.exists()) {  
 adSoyad = dataSnapshot.child("AdSoyad").getValue(String.class);  
 Yetki = dataSnapshot.child("Yetki").getValue(String.class);  
 } else  
 {  
 System.*out*.println("Hatakjşfghslokdjfhgslkşdjfhgsdfgsdfgsdfg");  
 }  
 }  
 @Override  
 public void onCancelled(DatabaseError databaseError) {  
 }  
 });  
 System.*out*.println("Oturum Var");  
 DatabaseReference databaseReference = FirebaseDatabase.*getInstance*().getReference().child("AcilDurumDurumu").child("Durum");  
 ValueEventListener valueEventListener = new ValueEventListener() {  
 @Override  
 public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  
 String durumGelen = dataSnapshot.getValue(String.class);  
 if (durumGelen != null) {  
 System.*out*.println(durumGelen);  
 if(durumGelen.equals("AcilDurum"))  
 {  
 DatabaseReference databaseReference = FirebaseDatabase.*getInstance*().getReference().child("users").child(mAuth.getUid()).child("Gördümü");  
 databaseReference.addListenerForSingleValueEvent(new ValueEventListener() {  
 @Override  
 public void onDataChange(@NonNull DataSnapshot dataSnapshot) {  
 if (dataSnapshot.exists()) {  
 String gorulduMu = dataSnapshot.getValue(String.class);  
 if (gorulduMu.equals("Görmedi"))  
 {  
 Intent intent = new Intent(MainActivity.this, GeriBildirim.class);  
 startActivity(intent);  
 overridePendingTransition(0, 0);  
 finish();  
 }  
 else if (gorulduMu.equals("Gördü"))  
 {  
 Intent intent = new Intent(MainActivity.this, YetkiKontrol.class);  
 startActivity(intent);  
 overridePendingTransition(0, 0);  
 finish();  
 }  
 else  
 {  
 Intent intent = new Intent(MainActivity.this, YetkiKontrol.class);  
 startActivity(intent);  
 overridePendingTransition(0, 0);  
 finish();  
 }  
 }   
 else  
 {  
 }  
 }  
 @Override  
 public void onCancelled(@NonNull DatabaseError databaseError) {  
 System.*out*.println("Veri okuma işlemi iptal edildi: " + databaseError.getMessage());  
 }  
 });  
 }  
 else if (durumGelen.equals("Normal"))  
 {  
 Intent intent = new Intent(MainActivity.this, YetkiKontrol.class);  
 startActivity(intent);  
 overridePendingTransition(0, 0);  
 finish();  
 }  
 else{System.*out*.println("Yok Ya bu");}  
 }  
 }  
 @Override  
 public void onCancelled(@NonNull DatabaseError databaseError) {  
 }  
 };  
 databaseReference.addListenerForSingleValueEvent(valueEventListener);  
 }  
 else  
 {  
 System.*out*.println("Oturum Yok");  
 }  
 }  
 });  
 btnGiris.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(MainActivity.this, GirisYap.class);  
 startActivity(intent);  
  
 }  
 });  
 btnKayıt.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(MainActivity.this, KayitOl.class);  
 startActivity(intent);  
 }  
 });  
 }  
  
 private boolean checkLocationPermission() {  
 if (ContextCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED*) {  
 ActivityCompat.*requestPermissions*(this, new String[]{Manifest.permission.*ACCESS\_FINE\_LOCATION*}, *PERMISSION\_REQUEST\_CODE*);  
 return false;  
 }  
 return true;  
 }  
 public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {  
 super.onRequestPermissionsResult(requestCode, permissions, grantResults);  
 if (requestCode == *PERMISSION\_REQUEST\_CODE*) {  
 if (grantResults.length > 0 && grantResults[0] == PackageManager.*PERMISSION\_GRANTED*) {  
 } else {  
 showPermissionRationale();  
 }  
 }  
 }  
 private void showPermissionRationale() {  
 AlertDialog.Builder builder = new AlertDialog.Builder(this);  
 builder.setTitle("Konum İzni");  
 builder.setMessage("Uygulama konum bilgisi almak için izin istiyor. Devam etmek istiyor musunuz? Daha fazla bilgi için [buraya tıklayın].");  
 builder.setPositiveButton("Evet", (dialog, which) -> {  
 ActivityCompat.*requestPermissions*(this,  
 new String[]{Manifest.permission.*ACCESS\_FINE\_LOCATION*},  
 *PERMISSION\_REQUEST\_CODE*);  
 });  
 builder.setNegativeButton("Hayır", (dialog, which) -> {  
 showPermissionRationale();  
 });  
 AlertDialog alertDialog = builder.create();  
 alertDialog.setOnShowListener(dialog -> {  
 alertDialog.getButton(DialogInterface.*BUTTON\_NEUTRAL*).setOnClickListener(view -> {  
 // "Daha fazla bilgi" linkine tıklandığında yapılacak işlem  
 // Örneğin: Kullanıcıyı bir web sayfasına yönlendirme  
 Uri uri = Uri.*parse*("https://hayatsinyalim.tomtal.com.tr/"); // Yönlendirilecek link  
 Intent intent = new Intent(Intent.*ACTION\_VIEW*, uri);  
 startActivity(intent);  
 });  
 });  
 alertDialog.show();  
 }  
}

**Açıklama ;**

Bu kod parçacığı, **MainActivity** adlı bir sınıf içeriyor ve uygulamanın giriş ekranını oluşturuyor. Burada **FirebaseAuth** kullanarak kullanıcı oturum durumunu kontrol ediyor, ayrıca **FirebaseDatabase** ile veritabanından belirli verileri çekiyor.

Bu aktivite ayrıca kullanıcının ilk kez giriş yaptığında bir açılış ekranı gösteriyor. Eğer izin verilmezse veya reddedilirse, kullanıcıyı izin için ikna etmeye çalışıyor. Aynı zamanda, kullanıcı izin verirse konum izni istiyor.

Giriş butonuna basıldığında **GirisYap** aktivitesine geçiş yapılmasını, kayıt butonuna basıldığında ise **KayitOl** aktivitesine geçiş yapılmasını sağlıyor.

Ayrıca, kullanıcı oturum durumunu kontrol ediyor ve duruma göre farklı aktivitelere yönlendiriyor (**GeriBildirim** veya **YetkiKontrol** gibi). Bu durum, **AcilDurumDurumu** adlı bir Firebase veritabanı düğümünden alınıyor.

Bir izin reddedildiğinde veya **onRequestPermissionsResult** çağrıldığında kullanıcıya bir açıklama ekranı gösteriliyor ve kullanıcı bu durumda tekrar izin istenebilir.