

Android

İçerikler

- [Faydalı Eğitim Setleri](#)
- [Telefonu WiFi Üzerinden PC'ye Bağlama](#)
- [Animasyonlar](#)
 - [Aşağıdan Gelme Animasyonu](#)
 - [Soluk Belirme](#)
- [Kaydırmalı Uygulama \(ViewPager ve PageAdapter\)](#)
- [SharedPreferences ile Veri Saklama](#)
 - [Veri Oluşturma ve Alma](#)
 - [Veri Güncelleme](#)
 - [Veri Silme](#)
- [SQLite ile Database Oluşturma](#)
 - [SQLite Giriş Temelleri](#)
 - [SQLite ile Basit DB Oluşturma](#)
 - [SQLite DB Oluşturma Kodları](#)
 - [SQLite DB İşlemleri Değiştirme](#)
 - [SQLite DB Okuma](#)
- [Karma Notlar](#)
 - [Component İsimleri](#)
 - [SeekBar](#)
 - [ProgressBar](#)
 - [Navigation Bar](#)
 - [Navigation View](#)
 - [AlertDialog](#)
 - [Gecikmeli Activity Başlatma](#)
 - [Bütün Eski Activity'leri Sonlandırıp Yeni Activity Açma](#)
 - [Projeyi Oluşturduktan Sonra Package İsmi Değiştirme](#)
- [Online Android Emulator](#)
- [Google Play Store](#)
- [Harici Bağlantılar](#)
 - [Kotlin için Harici Bağlantılar](#)
- [Karma Notlar](#)

Faydalı Eğitim Setleri

- [Udemy \(Ücretli\)](#)
- [YouTube \(İngilizce\), \(Ek video\)](#)
- [Android \(İngilizce\)](#)
- [Turkcell](#)
- [Yazılımcılar Dünyası](#)
- [Youtube](#)
- [Ümit KÖSE](#)

- [AbhiAndroid Android App Development eBook](#)

Telefonu WiFi Üzerinden PC'ye Bağlama

► ADB ile Telefonu PC'ye Bağlama

```
# C:\Users\[user]\AppData\Local\Android\sdk\platform-tools
# Telefonu USB ile bağlayın
adb usb # USB moduna alır
adb devices # Cihazları listeler
adb tcpip <port> # Port açar
adb connect <IP>:<port> # IP'ye verilen açılan port ile bağlanma
adb list devices # Bağlanıldığını kontrol etme

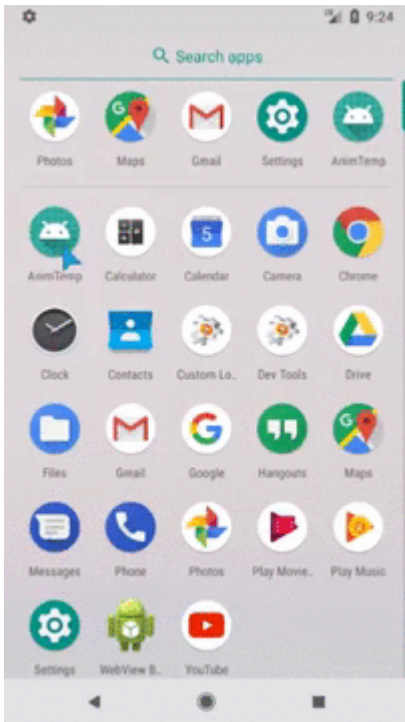
# ADB daemon işlemleri https://stackoverflow.com/a/52458945
which adb # Adb konumunu görme
locate adb
```

IP değerini öğrenmek için [Ayarları - WiFi - Gelişmiş](#) kısmına bakabilirsiniz (ya da `adb shell netcfg`).

- Açıklamalara [buradan](#) erişebilirsin.
- Telefonu PC üzerinden yönetmek için [buraya](#) bakabilirsin

Animasyonlar

Aşağıdan Gelme Animasyonu



Proje dizinine `anim` adlı klasör oluşturup içinde bir `resource file`'a bu kodları yazıyoruz.

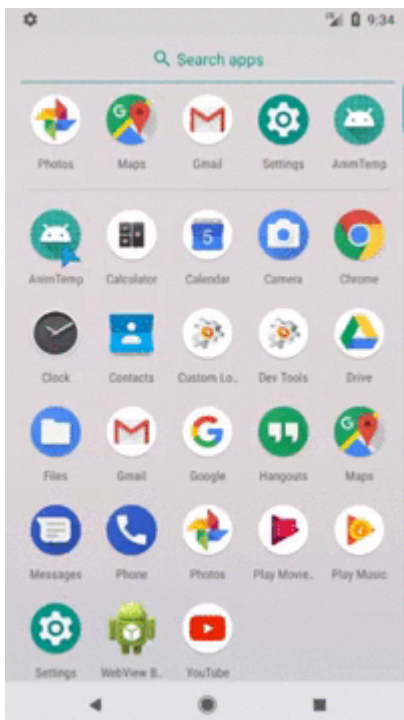
```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
  <translate
    android:duration="800"
    android:fromXDelta="0%p"
    android:fromYDelta="100%p"/>
</set>
```

Main'e alttakiler eklenmeli:

```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        btn_start.animation = AnimationUtils.loadAnimation(this,
R.anim.frombutton)
    }
    ...
}
```

- `btn_start` = Get Started adlı view'ın ID'si
- `frombutton` = üstteki kodların yazıldığı dosyanın adı

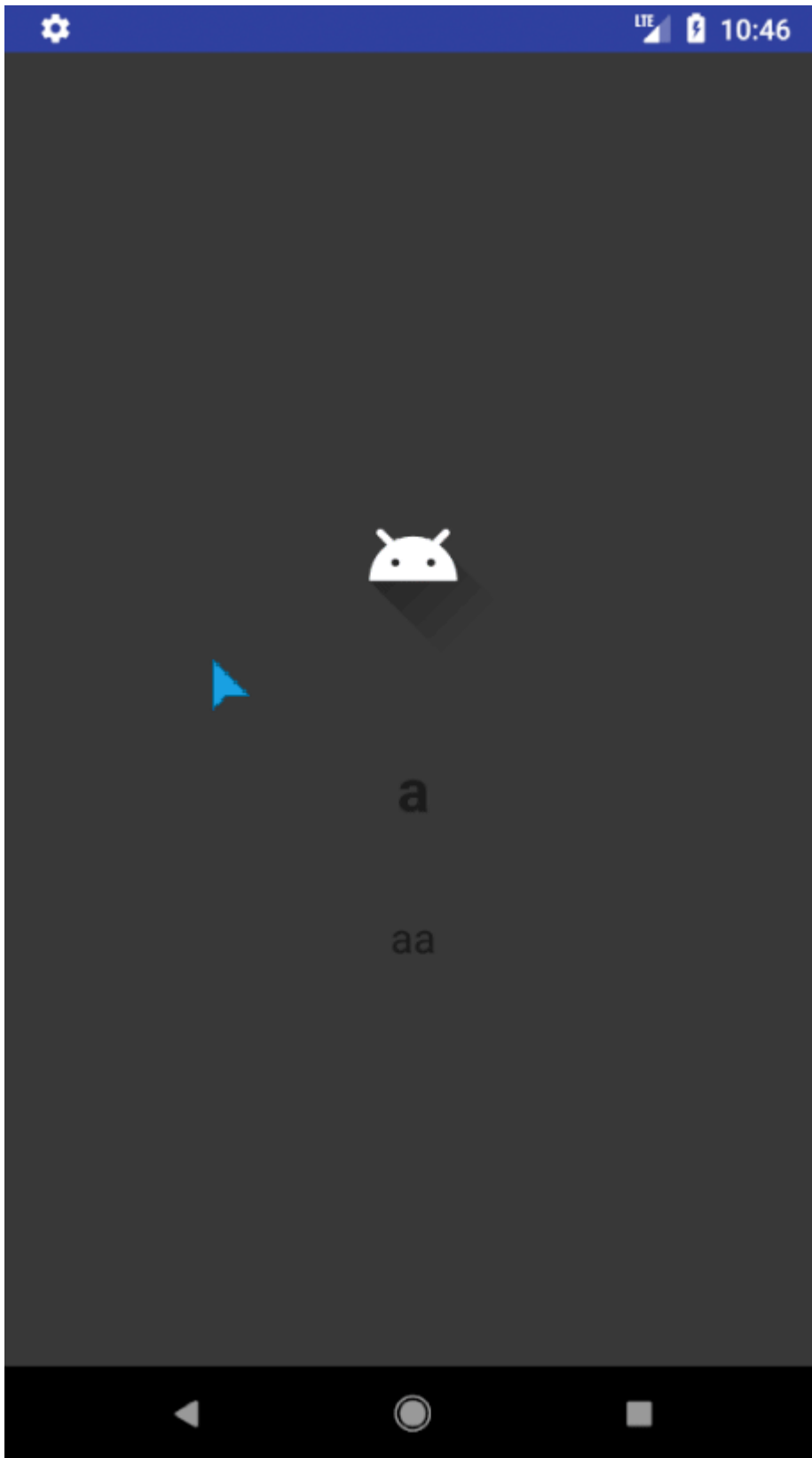
Soluk Belirme



```
<?xml version="1.0" encoding="utf-8"?>
<alpha xmlns:android="http://schemas.android.com/apk/res/android"
  android:duration="2000"
```

```
android:fromAlpha="0.0"  
android:toAlpha="1.0" />
```

Kaydırmalı Uygulama (ViewPager ve PageAdapter)



Manifest kodları:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.yemreak.iuapp">
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MainActivity"
            android:theme="@style/LogInTheme">
            <intent-filter>

                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Main kodları:

```
package com.yemreak.iuapp

import android.support.v7.app.AppCompatActivity
import android.os.Bundle
import android.support.v4.view.ViewPager
import kotlinx.android.synthetic.main.activity_main.*

class MainActivity : AppCompatActivity() {

    private lateinit var slideAdapter: SlideAdapter

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        slideAdapter = SlideAdapter(applicationContext)
        vp_main.adapter = slideAdapter
    }
    ...
}
```

SlideAdapter dosyası kodları:

```
package com.yemreak.iuapp

import android.content.Context
import android.graphics.Color
import android.support.v4.view.PagerAdapter
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.ImageView
import android.widget.LinearLayout
import android.widget.TextView

class SlideAdapter(context : Context) : PagerAdapter() {

    lateinit var inflater: LayoutInflater

    private val context = context
    private val arr_images = arrayListOf(
        R.drawable.ic_launcher_foreground,
        R.drawable.ic_launcher_foreground,
        R.drawable.water,
        R.drawable.text
    )
    private val arr_titles = arrayListOf("a", "b", "c", "d")
    private val arr_descriptions = arrayListOf("aa", "bb", "cc", "dd")
    private val arr_bgColors = arrayListOf(
        Color.rgb(55, 55, 55),
        Color.rgb(239, 85, 85),
        Color.rgb(110, 49, 89),
        Color.rgb(1, 118, 212)
    )

    override fun isViewFromObject(view: View, `object` : Any): Boolean {
        return (view == `object` as LinearLayout)
    }

    override fun getCount(): Int {
        return arr_images.size
    }

    override fun instantiateItem(container: ViewGroup, position: Int): Any {
        inflater =
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE) as LayoutInflater

        val view = inflater.inflate(R.layout.slide, container, false)
        val layout = view.findViewById<LinearLayout>(R.id.ll_slide1)
        val image = view.findViewById<ImageView>(R.id.iv_slide)
        val title = view.findViewById<TextView>(R.id.tv_slide_tittle)
        val description = view.findViewById<TextView>(R.id.tv_slide_description)

        layout.setBackgroundColor(arr_bgColors[position])
    }
}
```

```
        image.setImageResource(arr_images[position])
        title.text = arr_titles[position]
        description.text = arr_descriptions[position]
        container.addView(view)
        return view
    }
    override fun destroyItem(container: ViewGroup, position: Int, `object` : Any)
    {
        container.removeView(`object` as LinearLayout)
    }
}
```

Styles.xml kodları:

```
<resources>
    <!-- Base application theme. -->    <style name="AppTheme"
parent="Theme.AppCompat.Light.DarkActionBar">    <!-- Customize your theme
here. -->        <item name="colorPrimary">@color/colorPrimary</item>        <item
name="colorPrimaryDark">@color/colorPrimaryDark</item>        <item
name="colorAccent">@color/colorAccent</item>    </style>
    <style name="LogInTheme" parent="Theme.AppCompat.Light.NoActionBar">
<item name="colorPrimary">@color/colorPrimary</item>        <item
name="colorPrimaryDark">@color/colorPrimaryDark</item>        <item
name="colorAccent">@color/colorAccent</item>    </style>
</resources>
```

String.xml kodları:

```
<resources>    <string name="app_name">IUApp</string>    <string
name="title_login">Bu uygulama da nedir?</string></resources>
```

activity_main kodları:

```
<?xml version="1.0" encoding="utf-8"?><LinearLayout
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"
android:orientation="vertical"    tools:context=".MainActivity">
    <android.support.v4.view.ViewPager    android:id="@+id/vp_main"
android:layout_width="match_parent"    android:layout_height="match_parent" />

</LinearLayout>
```

slide.xml kodları:

```
<?xml version="1.0" encoding="utf-8"?><LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"    android:id="@+id/ll_slide1"
android:layout_width="match_parent"    android:layout_height="match_parent"
android:orientation="vertical"    android:gravity="center">

    <ImageView        android:id="@+id/iv_slide"
android:layout_width="wrap_content"        android:layout_height="wrap_content"
app:srcCompat="@mipmap/ic_launcher" />
    <TextView        android:id="@+id/tv_slide_tittle"
android:layout_width="wrap_content"        android:layout_height="wrap_content"
android:text="@string/title_login"        android:textSize="30sp"
android:textStyle="bold"        android:textAlignment="center"
android:layout_marginTop="40dp"/>
    <TextView        android:id="@+id/tv_slide_description"
android:layout_width="wrap_content"        android:layout_height="wrap_content"
android:text="@string/title_login"        android:textSize="21sp"
android:textAlignment="center"        android:layout_marginTop="40dp"/>
</LinearLayout>
```

SharedPreferences ile Veri Saklama

Veri Oluşturma ve Alma

- `val veri= this.getSharedPreferences(this.packageName, android.content.Context.MODE_PRIVATE)` // Veri kaydını değişkene atama
 - `this.packageName` : paket ismi (com.... en üst satırdaki)
 - `MODE_PRIVATE` : sadece benim uygulamamdan erişilebilirlik
- `var age1 = 30`
- `veri.edit().putInt("userAge", age1).apply()` // Veriyi kaydetme
 - `userAge` : anahtar
 - `age1` : değer / değişken
- `val age2= veri.getInt("userAge", 0)` // Kayıtlı veriyi alma
 - `userAge` : anahtar (put'takini almak için aynı olmalı)
 - `0` : Eğer anahtar yoksa, varsayılan değer ataması
- `println("stored age : $storedAge")` // veriyi gösterme

Veri Güncelleme

```
age = 31
veri.edit().putInt("userAge", age).apply() // Daha önceden olan bir anahtarın
üstüne kaydedilirse güncelleme olur.
```

Veri Silme

- `veri.edit().remove("userAge").apply()` // Veri silindi
 - `userAge` : silinecek anahtar

- `val age3 = veri.getInt("userAge", 0) // Veri olmadığı için age3 = 0 olacak.`
 - `userAge` : anahtar
 - `0` : varsayılan değer

SQLite ile Database Oluşturma

SQLite Giriş Temelleri

İlk olarak try - catch yapısı kurulur ve olası sorunda programın kapanması engellenir.

```
try {  
    ...  
}  
catch (e : Exception){  
    e.printStackTrace()  
}
```

Bütün kodları `...` olan yere yazacağız. Artık başlayabiliriz.

SQLite ile Basit DB Oluşturma

```
database = openOrCreateDatabase("Datas", Context.MODE_PRIVATE, null)
```

- "Datas" : Oluşturmak istediğimiz database'in adı ("Veriler", "Hey", "hop" vb.)
 - Yazım kuralları gereği database adı büyük harfle başlamalı
- `Context.MODE_PRIVATE` : Database'i private (özel) sadece bizim erişebileceğimiz halde kurmak.
 - (`Context.MODE` yazıp ALT+ SPACE yaparsanız detaylar çıkacaktır karşınıza)
- `null` : `CursorFactory`

SQLite DB Oluşturma Kodları

```
try {  
    val database = openOrCreateDatabase("Datas", Context.MODE_PRIVATE, null)  
  
    database.execSQL("CREATE TABLE IF NOT EXISTS datas (name VARCHAR, age INT(2))")  
    // INT(2) ile 2 rakam olacağını belli ediyoruz  
} catch (e : Exception){  
    e.printStackTrace()  
}
```

- `CREATE TABLE IF NOT EXISTS` table oluşturma
- `datas` table ismi
- `VARCHAR` char
- `INT` Int

SQLite DB İşlemleri Değiştirme

Temel yapısı `database.execSQL("...")` şeklindedir.

```
database.execSQL("INSERT INTO datas (name, age) VALUES ('Yunus' , 21)") // Veri Ekleme
database.execSQL("INSEER INTO datas (name, age) VALUES ('Emre', 15)") // Veri Ekleme
database.execSQL("UPDATE datas SET age = 21 WHERE name = 'Yunus") // Veri güncelleme
database.execSQL("DELETE FROM datas WHERE age = 15") // Veri silme
database.execSQL("SELECT FROM datas WHERE name = 'Yunus") // Yunus isimli olan dataları alır
database.execSQL("SELECT FROM datas WHERE name LIKE '%s'") // sonunda 's' harfi olanları alır
database.execSQL("SELECT FROM datas WHERE name LIKE 'y%") // başında 'y' harfi olanları alır
database.execSQL("SELECT FROM datas WHERE name LIKE '%u%") // içinde 'u' harfi olanları alır
```

- `INSERT INTO` Veri ekleme için SQL kodu
- `UPDATE` Veri güncelleme
- `SELECT` Veri seçme
- `datas` table ismi
- `name` değişken ismi
- `age` değişken ismi
- `VALUES` değerleri atamak için SQL kodu
- `'Yunus'` VARCHAR (string) tipindeki veri
- `21` INT(2) (Int) tipindeki veri

SQLite DB Okuma

```
if (database != null) {
    val cursor = database!!.rawQuery("SELECT * FROM datas", null)

    val nameIndex = cursor.getColumnIndex("name")
    val ageIndex = cursor.getColumnIndex("age")

    cursor.moveToFirst()

    while (cursor != null){
        println("İsim : ${cursor.getString(nameIndex)}" )
        println("Yaş : ${cursor.getString(ageIndex)}")

        cursor.moveToNext()
    }
}
```

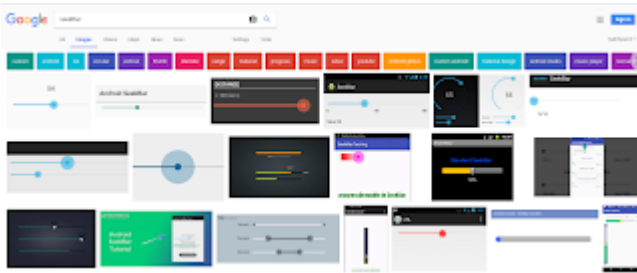
- `rawQuery(...)` SQL kodu ile veri alma
- `SELECT * FROM` Bütün verileri almak için SQL kodu

- `nameIndex` name sütündeki verilerin indexi
- `ageIndex` age sütündeki verilerin indexi
- `Cursor.moveToFirst()` Cursoru ilk elemana atıyoruz
- `Cursor.getString()` İstenen indexteki string olarak döndürür.
- `Cursor.moveToNext()` cursoru bir sütun aşağı indirme

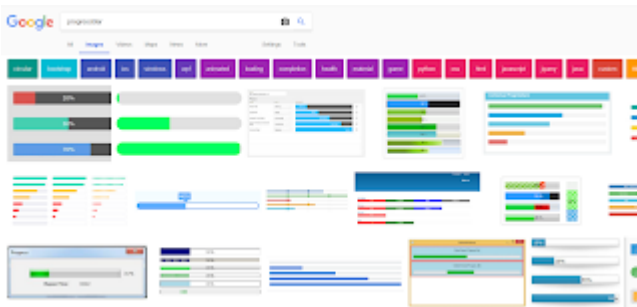
Karma Notlar

Component İsimleri

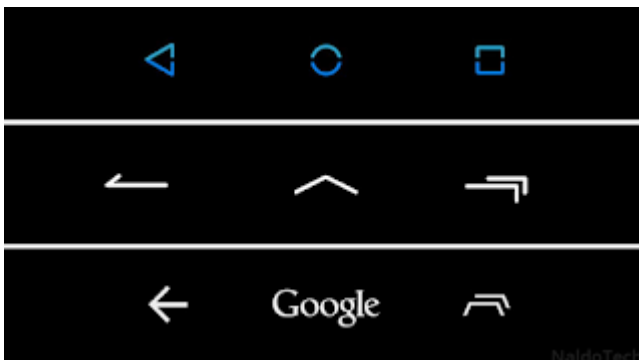
SeekBar



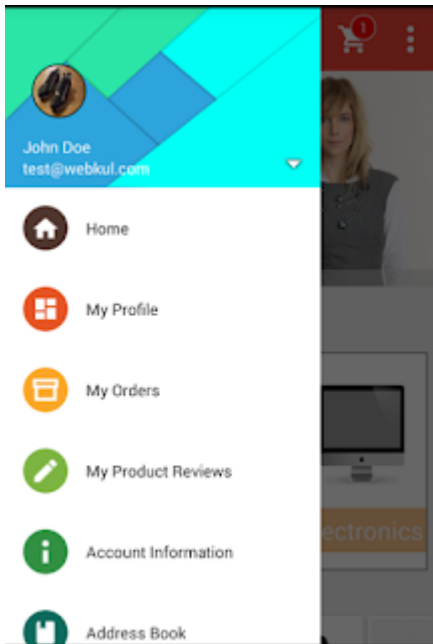
ProgressBar



Navigation Bar



Navigation View



AlertDialog



Gecikmeli Activity Başlatma

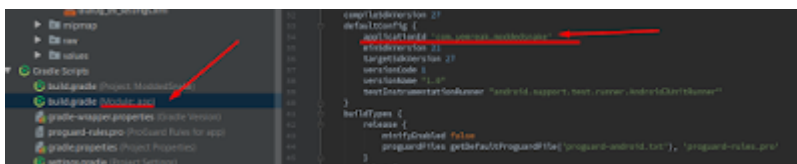
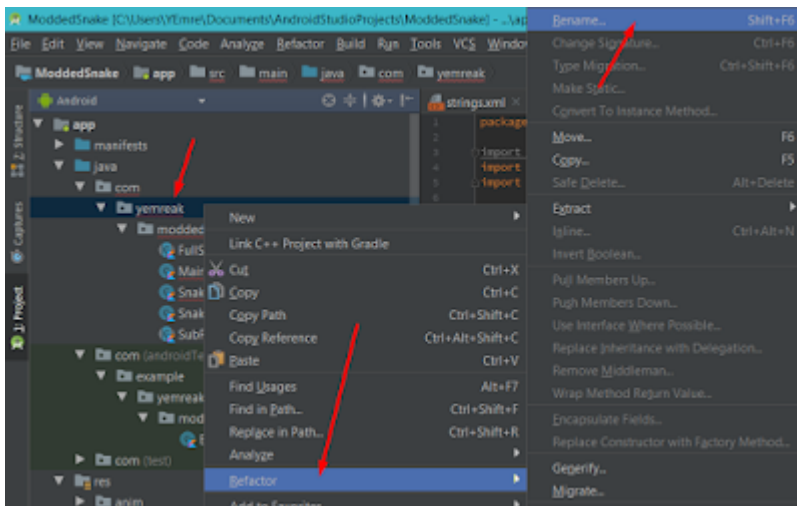
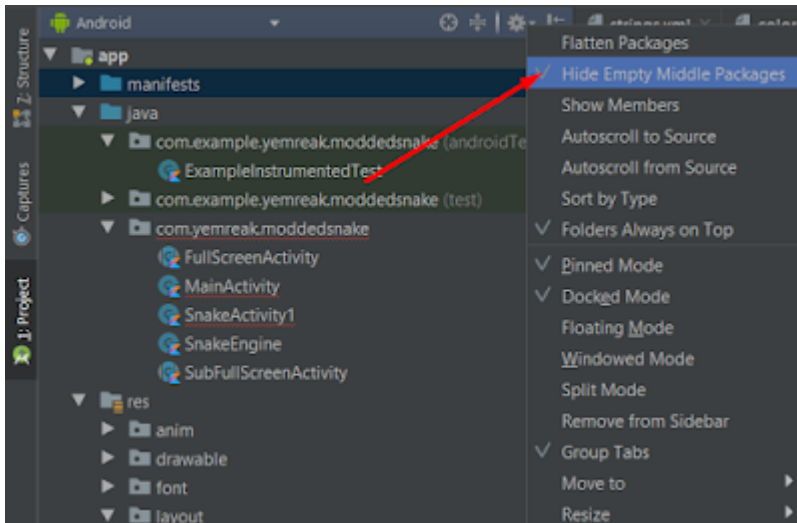
```
Handler().postDelayed({      startActivity(Intent(this, SnakeActivity1::class.java))
}, 400)
```

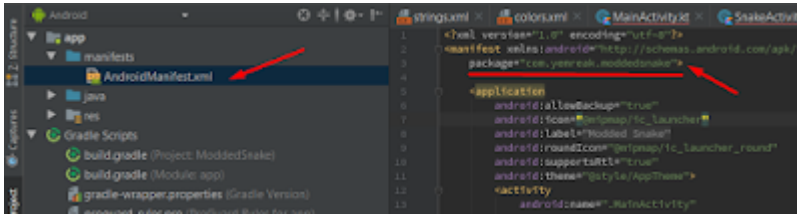
Bütün Eski Activity'leri Sonlandırıp Yeni Activity Açma

```
val intent = Intent(this, MainActivity::class.java)
intent.flags = Intent.FLAG_ACTIVITY_NEW_TASK or Intent.FLAG_ACTIVITY_CLEAR_TASK //
Tüm işlemleri bitirme
finish() // İşlemi sonlandırma
startActivity(intent)
```

Projeyi Oluşturduktan Sonra Package İsmi Değiştirme

- Project sekmesinden "Ayarlar" ikonundan "Hide Empty Middle Package"a tıklıyoruz.
- Değiştirmek istediğimiz klasöre sağ tıklayıp, "Refactor" -> "Rename" diyoruz ve adlandırıyoruz. (3. klasöre de aynı yapıyoruz)
- Eğer klasörü kaldırmak istiyorsak; Kalkacak olan klasörün içindekileri, onun üstündeki klasöre (yani onun yanına) yapıştırıyoruz.
 - Örn; "moddedsnake"i kesip, "com" adlı klasöre yapıştırıyoruz ve "yemreak" ı siliyoruz.
- Gradle'imizdeki build.gradle (Module: app) 'a giriyoruz ve applicationId'yi güncelliyoruz. Bizim örneğimizde oraya "com.yemreak.moddedsnake" yazıyoruz.
- Son olarak; Manifestimize girip "package name"i güncelliyoruz.
- Artık başarıyla "package name"i güncellemiş olduk. Emin olmak için projenizi Build etmeyi unutmayın





Online Android Emulator

APKOnline adlı sitenin *emulator*'ü için [buraya](#) bakabilirsiniz.

Google Play Store

Uygulamalarınızı **google play store**'a yüklemek için *develeport* hesabı açmanızı gerekmektedir.

- Google tek seferlik **25\$**'lık bir ücret almaktadır
 - Hesabınızı kapatmanız durumunda bu ücret **iade** edilecektir
 - Hesabınızdaki uygulamaları başka hesaplara aktarmak için [buraya](#) bakabilirsiniz
 - Detaylı bilgi için [buraya](#) bakabilirsiniz.
- Uygulama satışlarının **%30**'u Google'a gitmektedir

Harici Bağlantılar

- [Android Tasarım Bloğu](#)
- [Rename Package Name](#)
- [Finish All Previous Activity](#)
- [ADB komut listesi](#)
- [ADB ile telefona bağlanma](#)
- [Android Git Bağlantısı](#)

Kotlin için Harici Bağlatılar

- Java kodunu Kotlin'e çevirmek için CTRL + SHIFT + ALT + K
- [Kotlin Trick for Android](#)
- [Android Webview](#)
- [WebView'i appte kalarak kullanmak](#)
- [MediaPlayer \(.mp3\)](#)
- [Videoplayer](#)
- [TabLayout](#)
- [Navigation Drawer \(Türkçe / Video\)](#)
- [Material Drawer \(Navigation Drawer gelişmiş\)](#)
- [ViewPager \(Kaydırmalı Görünüm\) Tutorial](#)
- [Alert Dialog](#)
- [Paylaş seçeneğine uygulamayı ekleme](#)

Karma Notlar

- Swap mode ile her değişiklik durumunda snap alınır, emulator hızlı çalışır
- Visual Studio'nun emulatorü hızlı çalışıyor dendi (?)