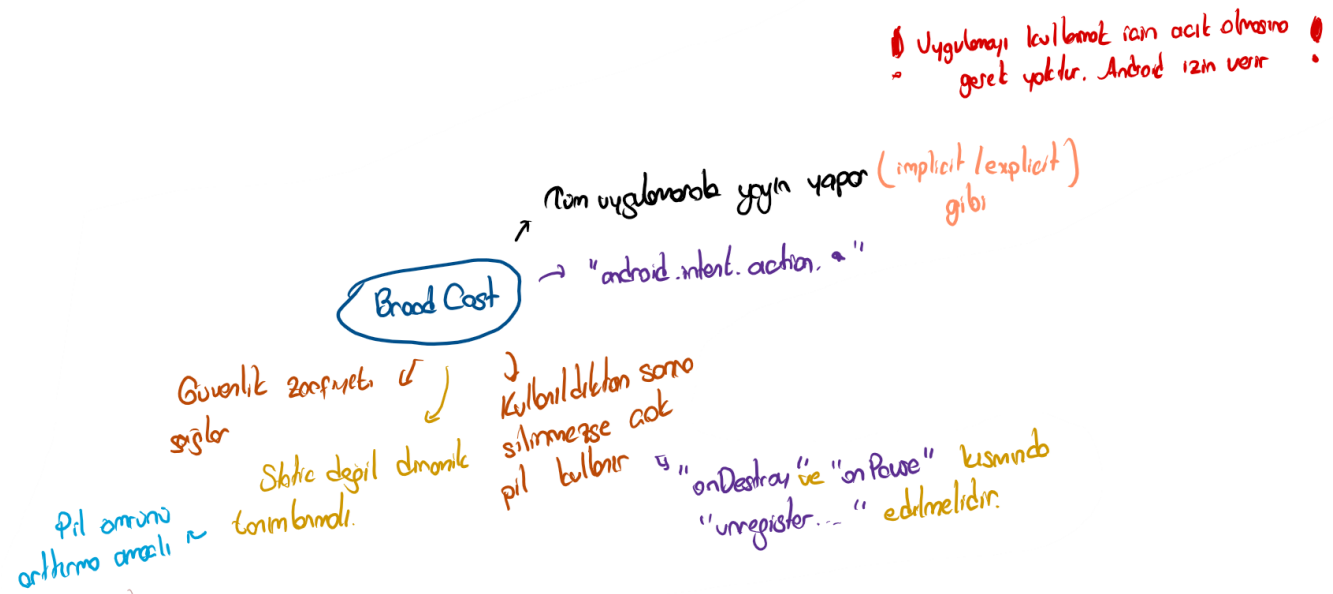


👣 Giriş | Broadcast

👁 Broadcast Hızlı Bakış

- 🔊 Haber salma olayı olarak ele alınabilir
- 📡 İşletim sistemindeki her uygulamaya bildirilir
- ☆ 📶 Wi-Fi'ya bağladım, 📶 Cihazı başlattım gibi örneklendirilebilir

📝 Yazılı Notlarım



description: Android üzerinde haber (Broadcast) oluşturma

🏠 Oluşturma | Broadcast

👁 Metotlara Göz Atma

```
m ➤ sendBroadcast(Intent intent) void
m ➤ sendBroadcast(Intent intent, String receiverPermission) void
m ➤ sendBroadcastAsUser(Intent intent, UserHandle user) void
m ➤ sendBroadcastAsUser(Intent intent, UserHandle user, Strin... void
m ➤ sendOrderedBroadcast(Intent intent, String receiverPermis... void
m ➤ sendOrderedBroadcast(Intent intent, String receiverPermis... void
m ➤ sendOrderedBroadcastAsUser(Intent intent, UserHandle user... void
m ➤ sendStickyBroadcast(Intent intent) void
m ➤ sendStickyBroadcastAsUser(Intent intent, UserHandle user) void
m ➤ sendStickyOrderedBroadcast(Intent intent, BroadcastReceiver... void
m ➤ sendStickyOrderedBroadcastAsUser(Intent intent, UserHandl...
```

🔍 Normal Broadcast

- 📡 Sırasız olarak tüm uygulamalara duyurulan haber yapısıdır

```
public void sendBroadcast() {  
    Intent intent = new Intent();  
    intent.setAction("com.example.myproject.ACTION_SHOW_TOAST");  
    // Set the optional additional information in extra field.  
    intent.putExtra("data", "This is a normal broadcast");  
    sendBroadcast(intent);  
}
```

📁 Ordered Broadcast (Sıralı)

- 🚩 XML üzerinde belirlenen `android:priority` sırasına göre uygulamalara haber verir
- 📦 Birden fazla aynı `android:priority` değerine sahip uygulamalara için seçim rastgele olur
- 👤 Her duyuru alan uygulama, `intent` verisini değiştirme hatta silme hakkına sahiptir

```
public void sendOrderedBroadcast() {  
    Intent intent = new Intent();  
  
    // Set a unique action string prefixed by your app package name.  
    intent.setAction("com.example.myproject.ACTION_NOTIFY");  
    // Deliver the Intent.  
    sendOrderedBroadcast(intent);  
}
```

🏠 Local Broadcast (Yerel)

- 🏠 Sadece uygulama içerisinde haber salınır
- 👤 Daha güvenlidir, çünkü diğer uygulamalar erişemez
- ✅ Daha verimlidir, tüm sisteme haber salmakla uğraşılmaz

```
LocalBroadcastManager.getInstance(this).sendBroadcast(customBroadcastIntent);
```

{% hint style="info" %} 📖 Detaylı bilgi için [Broadcasts](#) alanına bakabilirsin. {% endhint %}

👤 İzin Gerektirenler

- 📄 Manifest dosyası üzerinde `uses-permission` ile izin alınması gerekir
- 🚫 İzni olmayanlar uygulamaların erişmesi engellenir

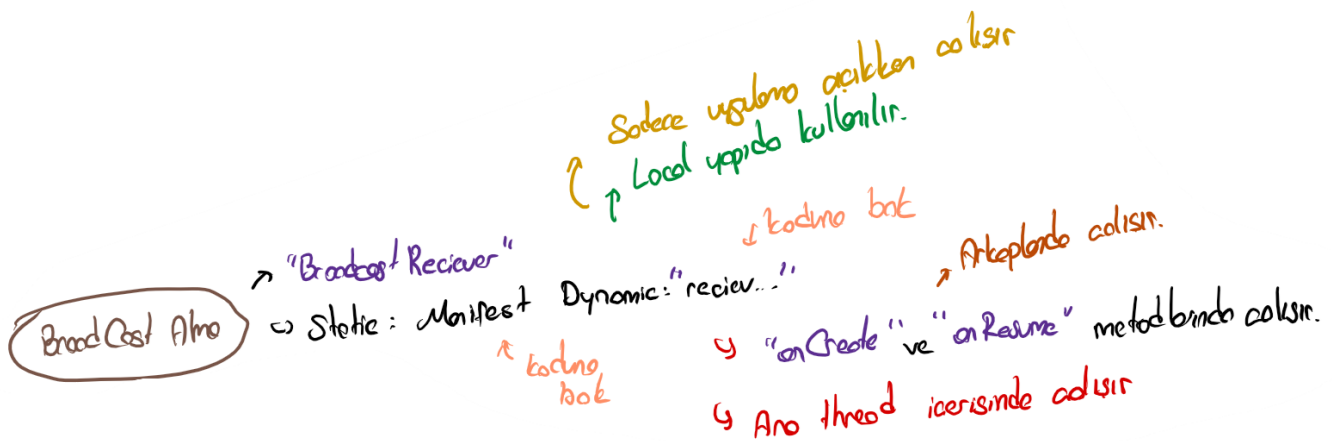
```
sendBroadcast(new Intent("com.example.NOTIFY"),Manifest.permission.SEND_SMS);
```

```
<uses-permission android:name="android.permission.SEND_SMS"/>
```

📡 Receiver | Broadcast

🔗 Broadcast Receiver Hakkında

- ⊗ UI thread üzerinden gerçekleştiğinden uzun işlemler yapılmamalı
- ⊖ `onReceive()` metodu içerisinde asenkron işlemler yapmayın
 - 🔧 Yapsanız bile `return` metodundan sonra broadcast işlemleri sonlandırılır
 - 🔧 Haliyle işlem asenkron olsa bile broadcast yapısına bağlı olduğundan ölecektir
- 🗨 `AlertDialog` gibi işlemler yerine `Notification` yapısı tercih edilmelidir



```
//Subclass of the BroadcastReceiver class.
private class myReceiver extends BroadcastReceiver {
    // Override the onReceive method to receive the broadcasts
    @Override
    public void onReceive(Context context, Intent intent) {
        //Check the Intent action and perform the required operation
        if (intent.getAction().equals(ACTION_SHOW_TOAST)) {
            CharSequence text = "Broadcast Received!";
            int duration = Toast.LENGTH_SHORT;

            Toast toast = Toast.makeText(context, text, duration);
            toast.show();
        }
    }
}
```

{% hint style="info" %} 📖 Detaylı bilgi için [Broadcast receivers](#) alanına bakabilirsin. {% endhint %}

📌 Receiver Türleri



📡 Static Receiver

- 📄 Manifest üzerinden kayıt edilmeleri gerekir
- 🤖 Uygulamamızı hedef almayan yayınlarını Android 8.0'dan itibaren alamaz
- 🔗 [implicit broadcast exceptions](#) yayınlarını hala alabilmektedir

```
<receiver
  android:name=".AlarmReceiver"
  android:exported="false">
  <intent-filter>
    <action android:name="
      "com.example.myproject.intent.action.ACTION_SHOW_TOAST"/>
  </intent-filter>
</receiver>
```

🌟 Dynamic Receiver

- 🔗 Uygulama üzerinden ilgilendiğimiz broadcast'e erişmek için [IntentFilter](#) kullanırız
- 📄 Genel kullanımı [onCreate](#) üzerinde yapılmaktadır (?)

```
IntentFilter intentFilter = new IntentFilter();
filter.addAction(Intent.ACTION_POWER_CONNECTED);
filter.addAction(Intent.ACTION_POWER_DISCONNECTED);
```

📋 Broadcast Kayıtları

- 🔗 İlk olarak [receiver](#) yapısını uygulamamıza [registerReceiver](#) ile kaydederiz
- 📄 Genelde [onResume](#) içerisinde [registerReceiver](#) işlemi yapılır
- 🔗 [onPause](#) içerisinde [unregisterReceiver](#) metodu ile kaldırırız

```
mReceiver = new AlarmReceiver();
this.registerReceiver(mReceiver, intentFilter);
```

```
unregisterReceiver(mReceiver);
```

{% hint style="info" %} 🕒 Detaylı bilgi için [Broadcast receivers](#) alanına bakabilirsin. {% endhint %}

🏠 Local Broadcast Alımı

- 🕒 [Local Broadcast](#), [Dynamic Receiver](#) ile alınmak zorundadır

```
LocalBroadcastManager.getInstance(this)
    .registerReceiver(mReceiver,
        new IntentFilter(CustomReceiver.ACTION_CUSTOM_BROADCAST));
```

```
LocalBroadcastManager.getInstance(this)
    .unregisterReceiver(mReceiver);
```

🔒 İzin Gerektirenlerin Alımı

```
<receiver android:name=".MyBroadcastReceiver"
    android:permission="android.permission.SEND_SMS">
    <intent-filter>
        <action android:name="android.intent.action.AIRPLANE_MODE"/>
    </intent-filter>
</receiver>
```

```
IntentFilter filter = new IntentFilter(Intent.ACTION_AIRPLANE_MODE_CHANGED);
registerReceiver(receiver, filter, Manifest.permission.SEND_SMS, null );
```

🕒 Broadcast Kısıtlamaları

! Paket adı bilgisi bulun !
"setPackage"

! İzinleri zorlayarak bulun !
✓ 5 izni olmayabilir erişemezler

"android.permission"
"<uses-permission>"

! Broadcast erişimi ile uygulanabilir
sıkıştırılabilir.

{% hint style="info" %} 📖 Detaylı bilgi için [Restricting broadcasts](#) alanına bakabilirsin. {% endhint %}

✳ Broadcast Tavsiyeleri

↑ "Best Practice"
↑ "com.yenercek.wiki"

→ Intent namespace'lerin eşit olması lazım.

→ Broadcast mesajlarını sınırlandır.
↳ Yetki sınırlaması etkle

↳ Broadcast
içerinde uzun işlem yapma
↳ main thread de çalışır.

↳ Dynamic receiver bulun

Güvenlik

🔗 Faydalı Bağlantılar

```
{% embed url="https://google-developer-training.github.io/android-developer-fundamentals-course-concepts-v2/unit-3-working-in-the-background/lesson-7-background-tasks/7-3-c-broadcasts/7-3-c-broadcasts.html#broadcast_receivers" %}
```