

DAHILI SINIFLAR

Neden Dahili Sınıf

- Diğer programlama dillerinde olan çoklu kalıtım (**multiple inheritance**) özelliği Java programlama dilinde yoktur.
- Java programlama dilinde çoklu kalıtım desteğinden faydalanmak için arayüz (**interface**) ve dahili sınıflar (inner classes) kullanılır

Dahili Sınıflar

- Sınıf içerisinde tanımlanmış sınıflardır.
- Birbiriyle bağlantılı olan yapıları bir bütün halinde tanımlamamıza yardımcı olurlar.
- Okunabilirlik ve yazılabilirlik kurallarına aykırı olduklarından pek tercih edilmemektedirler.
- Dahili sınıfları 3 grupta inceleyebiliriz:
 - Üye dahili sınıflar (member classes)
 - Yerel sınıflar (local classes)
 - İsimsiz sınıflar (anonymous classes)

Üye Dahili Sınıflar

- Nesneler farklı nesnelerin bir araya gelmesiyle oluşabilmektedir. (Örn. araba)
- Bir araya gelerek bir bütün oluşturan bu nesnelerin hepsini dahili sınıflar olarak ana sınıf içerisinde tanımlayabiliriz.

```
class CevreliYiciSınıf {  
  
    class DahiliSınıf {  
        //....  
    }  
  
    //...  
}
```

Örnek

```
public class Hesaplama {  
  
    public class Toplama {  
        public int toplamaYap(int a, int b) {  
            return a+b;  
        }  
    }  
  
    public static void main(String args[]) {  
        Hesaplama.Toplama ht = new Hesaplama().new Toplama();  
        int sonuc = ht.toplamaYap(3,5);  
        System.out.println("Sonuc = " + sonuc );  
    }  
}
```

- Toplama sınıfına ait bir nesne oluşturmak için, önce Hesaplama sınıfına ait bir nesne oluşturmamız gerekir.
- **Hesaplama.Toplama ht = new Hesaplama().new Toplama();**

Üye Dahili Sınıflar

- Dâhili üye sınıflara, public, friendly, protected veya private erişim belirleyicileri atanabilir.
- Dikkat edilmesi gereken diğer bir husus ise bir dâhili üye sınıf private erişim belirleyicisine sahip olsa dahi, çevreleyici sınıf içerisindeki tüm yordamlar tarafından erişilebilir olmasıdır.
- Dâhili üye sınıflar, içerisinde bulundukları çevreleyici sınıfların
 - tüm alanlarına (statik veya değil private dâhil)
 - yordamlarına (statik veya değil-private dâhil) erişebilirler.

Üye Dahili Sınıflar

- İççe üye dahili sınıflar yaratılabilir.
- Dâhili üye sınıflar, soyut (abstract) sınıf olarak tanımlanabilir. Bu soyut dâhili üye sınıflardan türeyen sınıflar, soyut dâhili üye sınıfların içerisindeki gövdesiz (soyut) yordamları iptal etmeleri gerekmektedir
- Üye dahili sınıflar kendisini çevreleyen sınıflarla aynı adı taşıyamazlarç

Yerel Dahili Sınıflar

- Yerel sınıflar, yalnızca içinde tanımlandıkları, metodun veya bloğun içerisinde geçerlidir.
- Dahili üye sınıfların çevreleyici sınıfları yerine, yerel sınıfların çevreleyici yordamları veya blokları vardır.
- Yerel sınıflar tanımlandıkları bu metodların veya blokların dışarısından erişilemezler.

```
public class Sinif {  
    public void metod() {  
        public class YerelSinif {  
            //...  
        }  
    }  
}
```


Yerel Dahili Sınıflar

- Yerel sınıflar başka sınıflardan türetilebilir veya arayüzlere (interface) erişebilir.
- Yerel sınıfların yapılandırıcıları olabilir.
- Yerel sınıflar, içinde bulundukları yordamın sadece final olan değişkenlerine ulaşabilirler.
- Yerel sınıflar, statik veya statik olmayan yordamların içerisinde tanımlanabilirler.

Yerel Dahili Sınıflar

- Yerel sınıflar, private, protected ve public erişim belirleyicisine sahip olamazlar sadece friendly erişim belirleyicisine sahip olabilirler.
- Yerel sınıflar, statik olarak tanımlanamaz.
- Yerel sınıf kendisini çevreleyen sınıfla aynı ismi taşıyamaz

İsimsiz Dahili Sınıflar

- İsimsiz ifade edilebilen sınıflardır.
- Özellikle olay dinleyicilerin (event listeners) devreye sokulduğu uygulamalarda sıkça kullanılırlar.
- Diğer dâhili sınıf çeşitlerinde olduğu gibi, isimsiz sınıflar direk extends ve implements anahtar kelimelerini kullanarak, diğer sınıflardan türetilemez ve arayüzlere erişemez.
- İsimsiz sınıfların herhangi bir ismi olmadığı için, yapılandırıcısı da (constructor) olamaz.

İsimsiz ve yerel dahili sınıfların aynı is için uygulanması

İSİMSİZ SAHİLİ SINIF:

```
return new Toplayici() {  
    public int hesaplamaYap() {
```

```
        // final olan yerel degiskenlere ulasabilir.  
        return a + b ;  
    }  
}; // noktali virgul sart
```

YEREL DAHİLİ SINIF:

```
public Toplayici topla(final int a, final int b) {  
    public class BenimToplayicim implements Toplayici {  
        public int hesaplamaYap() {
```

```
            // final olan yerel degiskenlere ulasabilir.  
            return a + b ;  
        }  
    } // yordam sonu  
    return new BenimToplayicim();
```

İsimsiz sınıfları, yerel sınıfların kısaltılmışı gibi düşünebilirsiniz. Yerel sınıflarda `return new BenimToplayicim()` yerine, isimsiz sınıflarda hangi sınıf tipinde değer döndürüleceği en başta belirtilir

Örnek

İsimsiz Dahili Sınıf	Yerel Dahili Sınıf
<pre>return new Toplayici() { public int hesaplamaYapO { return a + b ; } }; // noktali virgul sart</pre>	<pre>public Toplayici topla(final int a, final int b) { public class BenimToplayicim implements Toplayici { public int hesaplamaYapO { return a + b ; } } return new BenimToplayicimO; }</pre>

- Yerel sınıflarda return new BenimToplayicim() yerine, isimsiz sınıflarda hangi sınıf tipinde değer döndürüleceği en başta return new Toplayici() ile belirtildi.