

SINIFLAR VE NESNELER

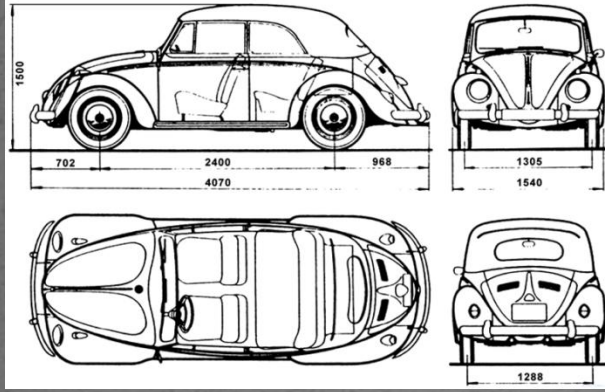


SINIF

- En genel haliyle sınıf kullanıcının tanımladığı bir veri tipidir. Nesne ise o tipte oluşturulmuş bir değişken gibi düşünülebilir.
- Java Sınıfı = veri + metotlar

NESNE

SINIF



JAVA

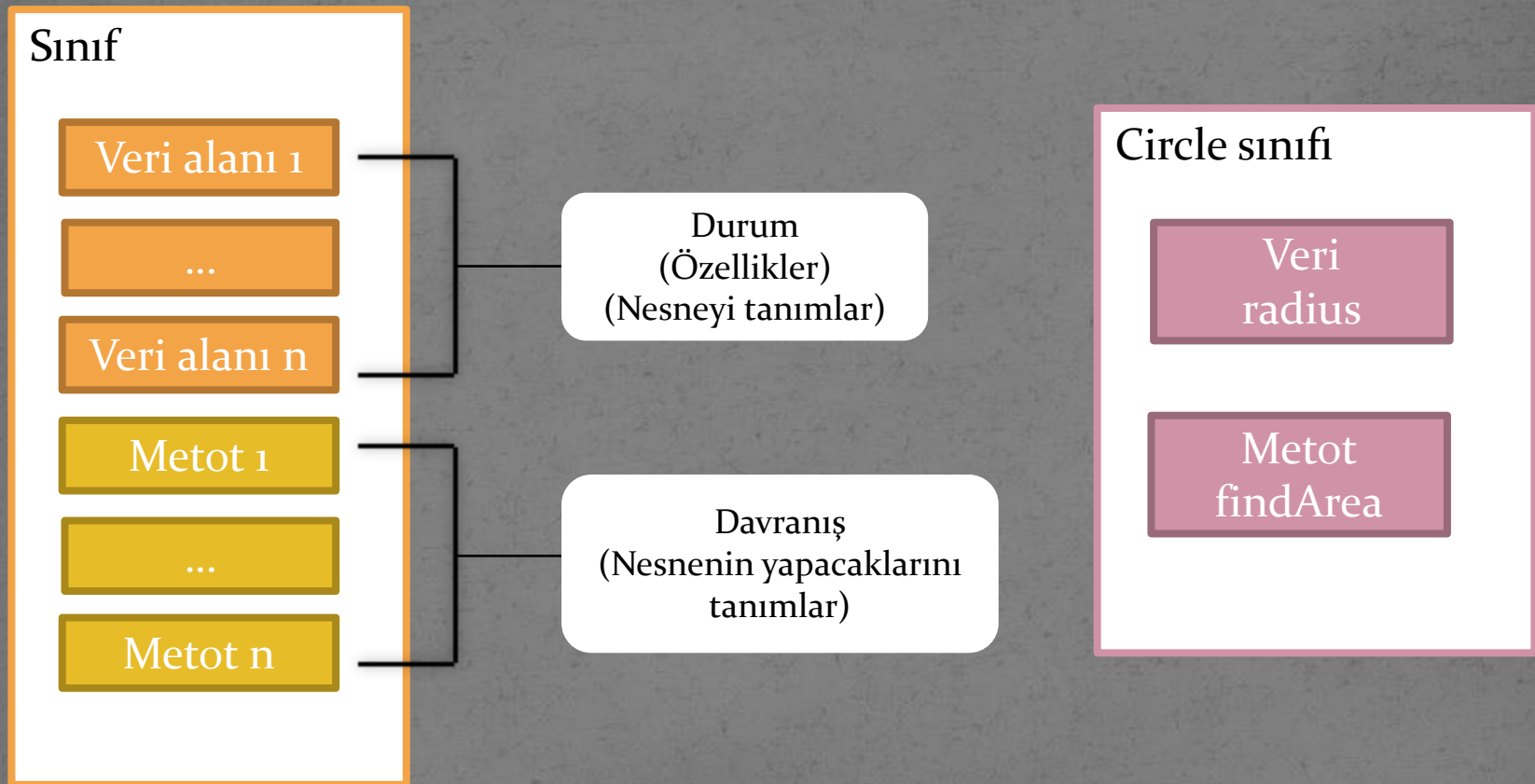


NESNE



Sınıflarla nesneler arasındaki ilişki otomobil tasarımıyla otomobil arasındaki ilişki gibidir. Tasarımda tanımlananlarla otomobilleri yaparsınız. Ama tasarımı değil otomobili sürebilirsiniz.

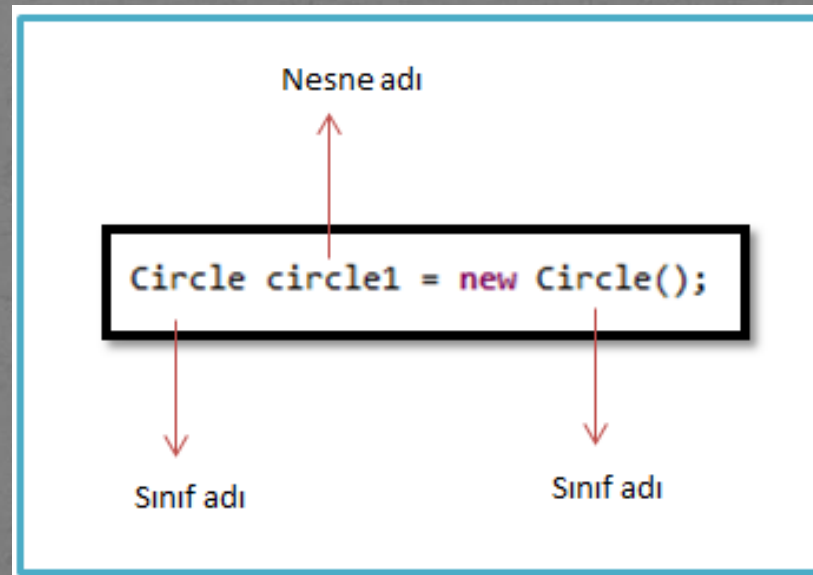
SINIF



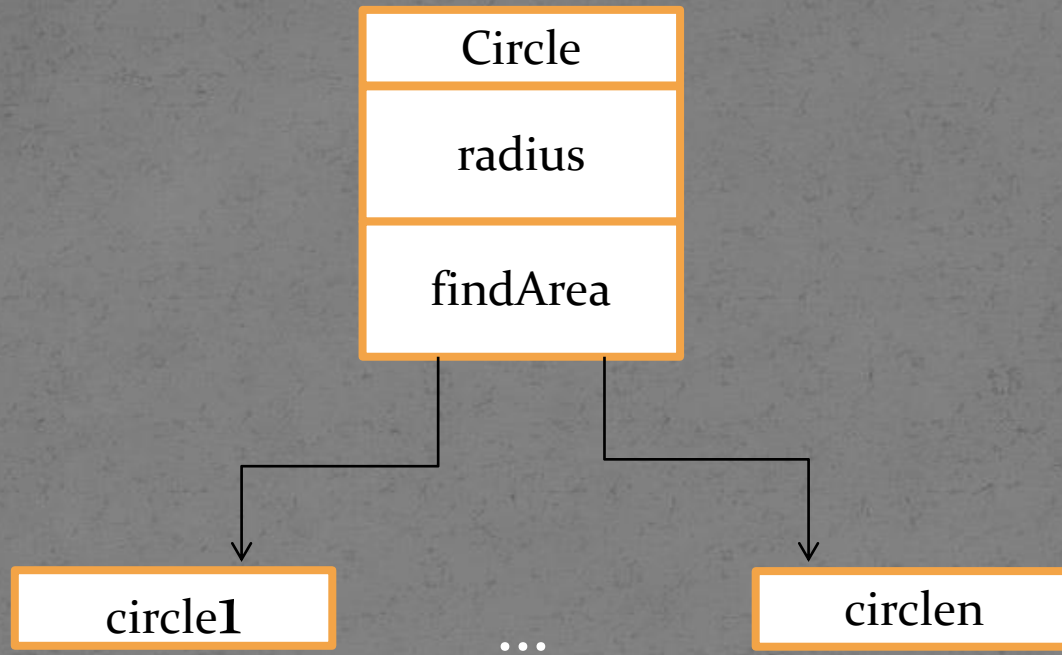
```
class Circle {  
    double radius = 1.0;  
  
    double findArea()  
    {  
        return radius*radius*Math.PI;  
    }  
}
```

- Bu sınıf şimdiye kadar gördüğümüz sınıflardan farklıdır: **Bu sınıfta main yoktur.** Dolayısıyla bu sınıfı çalıştıramazsınız.
- Bu sınıf yalnızca Circle nesnelerini açıklamak ve üretmek için kullanılan tanımlamalardır. Yani sınıf bir nesnenin verilerinin ve metotlarının ne olacağını tanımlar.

Nesne oluşturma

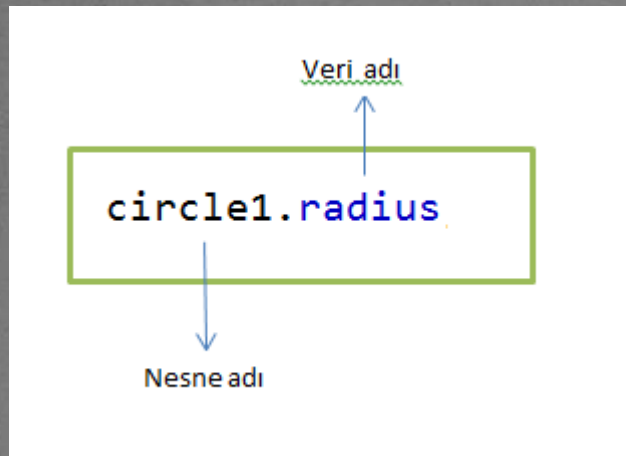


NESNE

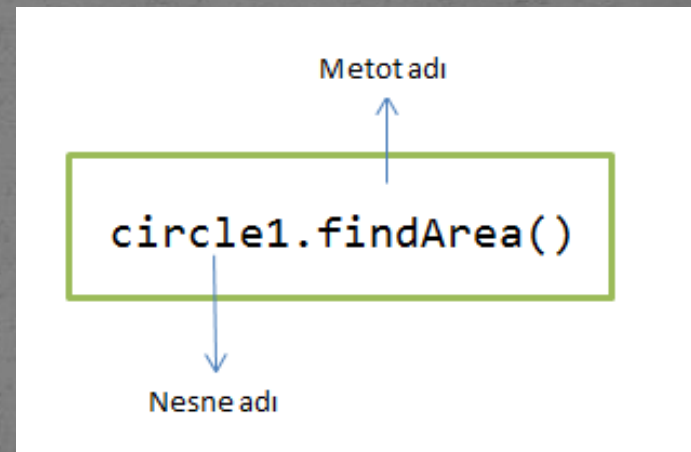


Nesnelerin veri ve metotlarına ulaşma

Veriye ulaşma



Metoda ulaşma



YAPILANDIRICILAR (CONSTRUCTORS)

- Tanımladığımız Circle sınıfından üreteceğimiz her nesne aynı yarıçapa(1.0) sahip olacaktır. Ancak aynı Circle sınıfından yarıçapları farklı nesneler üretebilmek daha kullanışlı değil midir?
- Java, sınıflarında, sınıfın verilerine ilk değer vermek için kullanılabilen, *yapılandırıcı* olarak adlandırılan, özel bir metot tanımlamaya izin verir.
- Yapılandırıcının ismi sınıfinkiyile aynı olmalıdır.

YAPILANDIRICILAR (CONSTRUCTORS)

Circle sınıfının, radius verisine ilk değer veren yapılandırıcısının tanımlanması

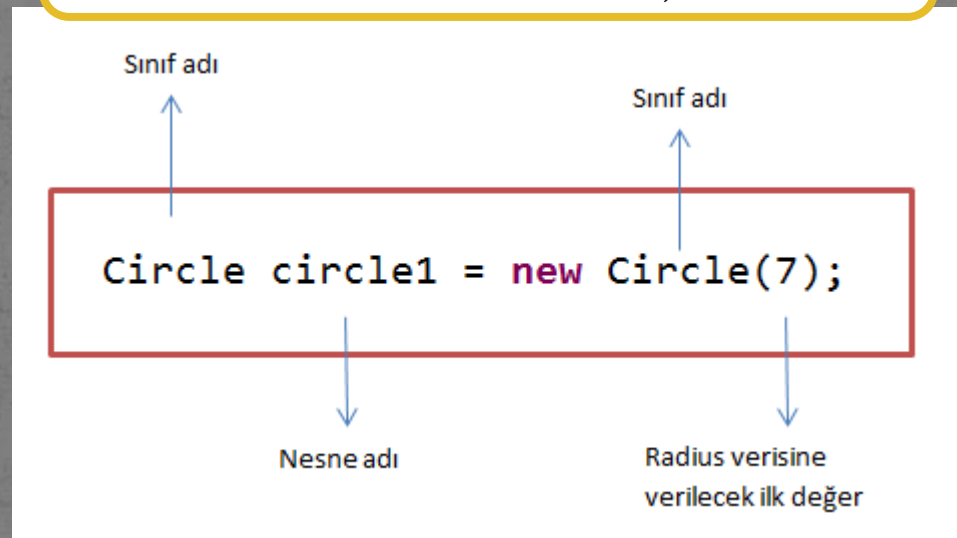
```
class Circle {  
  
    double radius;  
  
    Circle(double r)  
    {  
        radius = r;  
    }  
  
    double findArea()  
    {  
        return radius*radius*Math.PI;  
    }  
}
```

Sınıf adı

Yapılandırıcı

YAPILANDIRICI İLE NESNE OLUŞTURMA

radius verisinin değeri 7 olan, circle1 isimli Circle nesnesinin oluşturulması



YAPILANDIRICILAR (CONSTRUCTORS)

- Yapılandırıcılar iki yönden diğer metotlardan farklıdırlar:
 - Yapılandırıcıların dönüş tipi yoktur.
 - Yapılandırıcılar başlarına *new* koyularak çağırılırlar.
- radius verisine başka bir metotla da değer verilemez miydi? Ya da değerini sonradan main'in içinde belirleyemez miydim? Fark ne?
- Fark yapılandırıcının nesneyi radius değerini belirleyerek oluşturması. Herhangi başka bir metot kullanılarak yapılırsa, önce nesne başka bir radius değeriyle oluşturulur, sonra metot çağırıldığında radius istediğimiz değer yapılır.

VARSAYILAN YAPILANDIRICI (DEFAULT CONSTRUCTOR)

- Sınıftaki verilere default (varsayılan) bir ilk değer vermek için Varsayılan Yapılandırıcılar oluşturulur.
- Varsayılan Yapılandırıcılar parametre almaz, onun dışında bildiğimiz yapılandırıcılarla aynı biçimde tanımlanır ve kullanılırlar.

VARSAYILAN YAPILANDIRICININ TANIMLANMASI

Circle sınıfının, radius verisine default ilk değerini veren varsayılan yapılandırıcısının tanımlanması

```
class Circle {
```

```
    double radius;
```

Sınıf adı

```
    Circle()
```

```
    {
```

```
        radius = 1.0;
```

```
    }
```

Varsayılan
Yapılandırıcı

```
    Circle(double r)
```

```
    {
```

```
        radius = r;
```

```
    }
```

Radius verisine ilk
değerin verilmesi

```
    double findArea()
```

```
    {
```

```
        return radius*radius*Math.PI;
```

```
    }
```

```
}
```

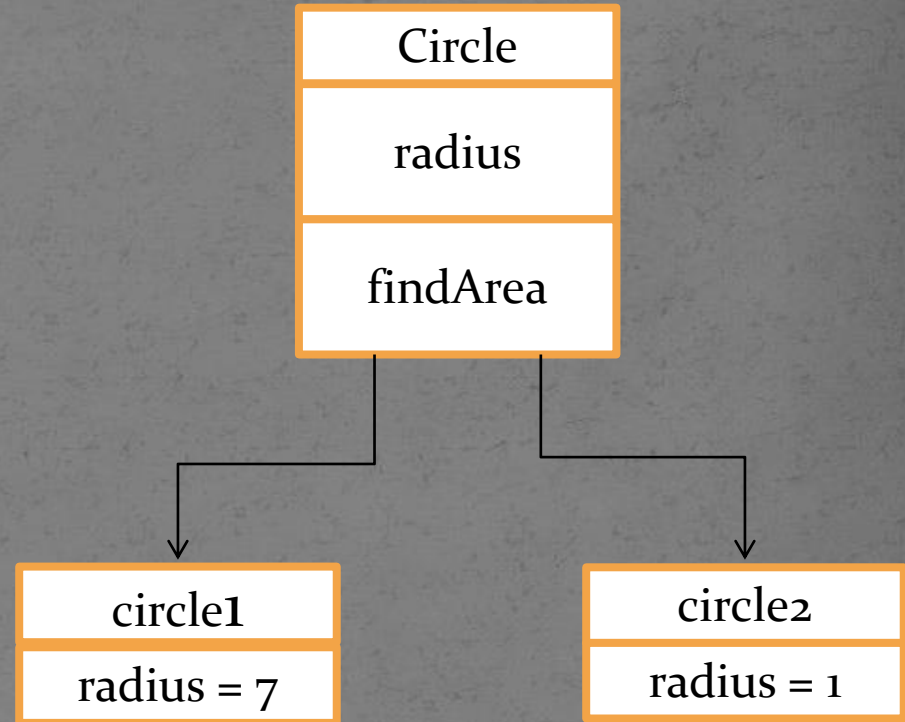

VARSAYILAN YAPILANDIRICIYLA NESNE OLUŞTURMA

radius verisinin değeri 1 olan circle2
isimli Circle nesnesinin oluşturulması

```
Circle circle2 = new Circle();
```

YAPILANDIRICIYLA NESNE OLUŞTURMA

```
Circle circle1 = new Circle(7);  
Circle circle2 = new Circle();
```



UYARI:

1. Eğer bir sınıfta hiç yapılandırıcı (varsayılan ya da değil) tanımlanmadıysa, sınıfın bir nesnesi **new SınıfAdı()** komutuyla oluşturulabilir. Ancak eğer sınıfta sadece varsayılan olmayan yapılandırıcı tanımlandıysa sınıfın **new SınıfAdı()** komutuyla bir nesnesi oluşturulamaz.
2. Bir sınıf verisi, nümerik bir tipteyse o, boolean tipteyse false, char tipindeyse '\u0000' ve bir nesneyse null ilk değerini alır.

- Birlikte bir Rectangle sınıfı oluřturalım...

