Sınıf (class) yapısının bu sınıftan üretilecek nesneler için bir şablon görevi gördüğünü söyleyebiliriz.

Nesneleri oluşturmanın ilk adımı, o nesnenin özelliklerini ve eylemlerini belirleyen sınıf yapısını tanımlamaktır. Java'da sınıf yapısı,

**class** **S**ı**n**ı**f**İ**smi**

    {

        Özellik Tanımları;

        Metod Tanımları;

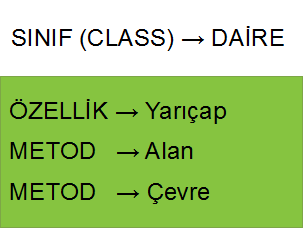
    }

şeklinde oluşturulur.

Sınıf içinde özellik tanımlamak için değişkenler, metot tanımlamak için de çeşitli yordamlar bulunabilir.

**ÖRNEK:**

Aşağıdaki **Daire** adlı sınıf, **Yaricap** adlı bir özelliği ve **Alan** ile **Cevre** adlı iki metodu ile gösterilmiştir:



Bu sınıfı oluşturmak için Java'da aşağıdaki kod yazılabilir:

**class** **Daire**

{

**public** **double** yariCap;

**public** **double** **alan**()

    {

**return** 3.14159 \* yariCap \* yariCap;

    }

**public** **double** **cevre**()

    {

**return** 2 \* 3.14159 \* yariCap;

    }

}

**SINIFTAN NESNE OLUŞTURMA**

Sınıf yapısının bu sınıftan üretilecek nesneler için bir şablon görevi gördüğünü söyleyebiliriz.

Yukarda tanımladığımız **Daire** adlı sınıftan **d** adlı bir nesne oluşturmak için,

Daire d = **new** Daire();

şeklinde bir ifade kullanmamız gerekir. **d**, artık daire sınıfından bir nesne olduğu için bu sınıfın metotlarını kullanabilir. Metotları çağırmak için,

**d.cevre();  
d.alan();**

ifadelerini kullanırız.

Aynı şekilde özelliklere erişmek için de "**.**" sembolünü kullanırız:

**d.yariCap;**

**ÖRNEK UYGULAMA: SINIF TANIMLAMA VE SINIFTAN NESNE TÜRETME**

I. Bir yarıçap bilgisinden faydalanarak dairenin çevresini ve alanını bulan örneği yazalım. Bunun için aşağıdaki Java kodunu yazınız:

//Dosya Adı: Daire.java

**package** com.turkcell.and101;

**public** **class** **Daire** {

**public** **double** yariCap;

**public** **double** **alan**() {

**return** 2 \* 3.14 \* yariCap;

}

}

​

II. Programı çalıştırdığınızda önce yarıçap sorulacak ve bu veri girilince çevre ve alan hesaplanıp ekrana basılacaktır:

//Dosya Adı: Main.java

**package** com.turkcell.and101;

**import** java.util.Scanner;

**public** **class** **Main** {

**public** **static** **void** **main**(String[] args) {

Scanner giris=**new** Scanner(System.in);

**double** yariCap=giris.nextDouble();

Daire daire=**new** Daire();

daire.yariCap=yariCap;

System.out.println(daire.alan());

}

}