DATA HIDING MEANS --

- declare class variables/attributes as `private`
- provide public **get** and **set** methods to access and update the value of a `private` variable

## Get and Set

You learned from the previous chapter that `private` variables can only be accessed within the same class (an outside class has no access to it). However, it is possible to access them if we provide public **get** and **set** methods.

The `get` method returns the variable value, and the `set` method sets the value.

Syntax for both is that they start with either `get` or `set`, followed by the name of the variable, with the first letter in upper case:

## Example

```
public class Person {
  private String name; // private = restricted access

  // Getter
  public String getName() {
    return name;
  }

  // Setter
  public void setName(String newName) {
    this.name = newName;
  }
}
```

Example explained

The `get` method returns the value of the variable `name`.

The `set` method takes a parameter (`newName`) and assigns it to the `name` variable. The `this` keyword is used to refer to the current object.

However, as the `name` variable is declared as `private`, we **cannot** access it from outside this class:

## PRACTISING DATA HIDING –

```
1  package objectorientiedprograming;
2
3  class Rectangle123
4  {
5      private double length;
6      private double breadth;
7
8      public double getLength()
9      {
10         return length;
11     }
12
13     public double getBreadth()
14     {
15         return breadth;
16     }
17
18     public void setLength(double l)
19     {
20         if(l>0)
21         {
22             length=l;
23         }
24         else
25         {
26             length=0;
27         }
28     }
29
30     public void setBreadth(double b)
31     {
32         if(b>0)
33         {
34             breadth=b;
35         }
36         else
37         {
38             breadth=0;
39         }
40     }
```

```
    public void area()
    {
        double area=getLength()*getBreadth();
        System.out.println("Area of the Rectangle is : "+area);
    }

    public void perimeter()
    {
        double perimeter=2*(getLength()+getBreadth());
        System.out.println("Perimeter of the Rectangle is :"+perimeter);
    }
}

public class DataHiding_area_perimeter_example {
    public static void main(String arg[])
    {
        Rectangle123 r = new Rectangle123();
        r.setLength(10);
        r.setBreadth(10);

        r.area();
        r.perimeter();

        System.out.println("Length of the rectangle is :"+r.getLength());
        System.out.println("Perimeter of the Rectangle is : "+r.getBreadth());
    }

}
```

## OUTPUT—

```
<terminated> DataHiding_area_perimeter_example [Java Application]
Area of the Rectangle is : 100.0
Perimeter of the Rectangle is :40.0
Length of the rectangle is :10.0
Perimeter of the Rectangle is : 10.0
```