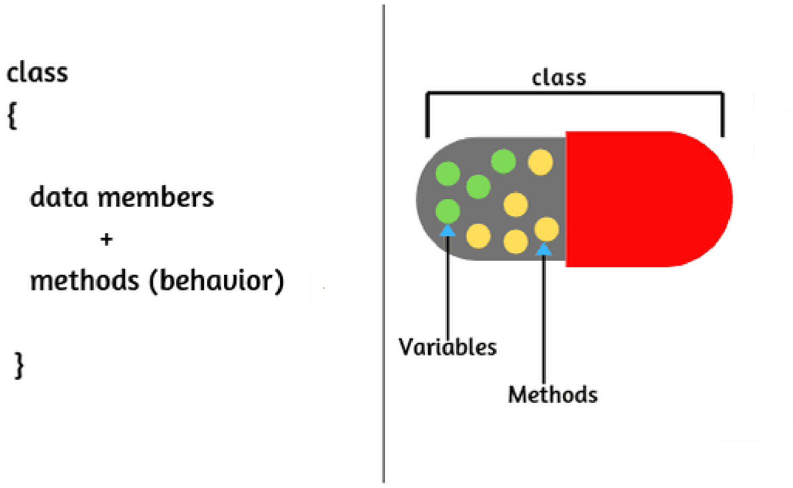
Encapsulation

============

Encapsulation in java is a mechanism of wrapping the data (variables) and data related methods together as a single unit.



E.x

1. Medicine enclosed inside the capsule which are hidden from the patient.
2. Engine, brake and gear system are hidden from the driver.

Class without encapsulation concept

—---

E.x

```
class employee                        // first class
{
        int empid = 101;
}

class company                         // second class
{
Public static void main(String[ ] args)
        {
        employee ref = new employee();   // creating object
        System.out.println(ref.empid);    // initialising the instance variable
        }
}
```

Note :

1. In the above example we can access the “ int empid ” from anywhere, hence the privacy is affected.
2. If the data is sensitive then the data is not secure in this concept.

How to perform encapsulation

1. If we declare the integer with a private keyword then only we can secure the data.
2. To call or access the data we need to create some methods called “getter” and “setter” methods.

Class with encapsulation concept
—---
E.x

```
class employe                                   // first class
{
        private int empid;                      // it is called as data hiding

        public void setEmpID(int eid)           // setter method
        {
                empid = eid;
        }

        public int getEmpID()                   // getter method
        {
                return empid;                   // return statement
        }
}
```

Note : To set the values under the data we use the **getter** and **setter** methods to
This concept is known as encapsulation.

```
class company                           // second class
{
Public static void main(String[ ] args)
        {
        employee e = new employee();    // creating object
        e.setEmpID(101);                // to set the value of variable
        System.out.println(e.getEmpID); // to get the value of variable
        }
}
```

—----------
First E.x
—-----------

```
package testJava;
class sclass
{
        private int Id;                 // data or variable

        public void setId(int eid)      // setter method
        {
                Id = eid;
```

```
    }
    public int getId()                      // getter method
    {
        return Id;
    }
}
```

—---------

```
package testJava;

public class aclass
{
    public static void main(String[] args)
    {
        sclass sc = new sclass();

        sc.setId(123);  // setting value to instance variable by using setter method
        System.out.println(sc.getId()); // getting value by using getter method
    }
}
```

—--------

Second e.x

—-------------

```
package Testing_Package;


public class Javatesting
{
    private String Name;
    private int ID;

    public void setID(String setName, int setID)        // setter method
    {
        ID = setID;
        Name = setName;
    }

    public int getID()                                          // getter method
    {
        return ID;
    }

    public String getName()                                     // getter method
    {
        return Name;
    }
}
```

—------------

```
package Testing_Package;

public class NewClass
{
        public static void main(String[] args)
        {
        Javatesting ref = new Javatesting();

        ref.setID("qwerty",001);                                // setter method call
        System.out.println(ref.getID());                        // getter method call
        System.out.println(ref.getName());                      // getter method call
        }
}
```