Wrapping up a data member and a method together into a single unit (in other words class) is called Encapsulation.

Encapsulation is like enclosing in a capsule. That is enclosing the related operations and data related to an object into that object.

Encapsulation means hiding the internal details of an object, in other words how an object does something.

Encapsulation prevents clients from seeing its inside view, where the behaviour of the abstraction is implemented.

Encapsulation is a technique used to protect the information in an object from another object.

Hide the data for security such as making the variables private, and expose the property to access the private data that will be public.

Exercise:

Suppose you are a Mobile Phone Manufacturer and you have designed and developed a Mobile Phone design (a class). Now by using machinery you are manufacturing Mobile Phones (objects) for selling, when you sell your Mobile Phone the user only learns how to use the Mobile Phone but not how the Mobile Phone works.

This means that you are creating the class with functions and by with objects (capsules) of which you are making available the functionality of your class by that object and without the interference in the original class.

abstract class Mobile\_Phone {

private int number;

private int toNumber;

private String message;

public void Calling();

public void SMS();

}

public class N\_142: Mobile\_Phone {

}

public class N\_271: Mobile\_Phone {

   private int[] fmList[];

public void FMRadio();

public void MP3();

public void Camera();

}

public class B: Mobile\_Phone {

private int[] fmList[];

private String emailBody;

public void FMRadio();

public void MP3();

public void Camera();

public void Video\_Recording();

public void ReadAndSendEmails();

}