



# Manual de Prácticas de Técnicas de Programación

Secretaría/División:DIMEI

Área/Departamento: Mecatrónica

## Encapsulamiento y herencia

N° de práctica: 2

Nombre completo del alumno		Firma
N° de brigada:	Fecha de elaboración:	Grupo:

Elaborado por:	Revisado por:	Autorizado por:	Vigente desde:
M.F. Gabriel Hurtado Chong M.I. Jorge Armando Rodríguez Vera	M.F. Gabriel Hurtado Chong	Colegio de Programación	18-Feb-16



# Manual de Prácticas de Técnicas de Programación

Secretaría/División:DIMEI

Área/Departamento: Mecatrónica

## 1. Seguridad en la ejecución

	Peligro o Fuente de energía	Riesgo asociado
1	Tensión alterna	Electrocución

## 2. Objetivos de aprendizaje

**I. Objetivo general:** Entender la reutilización de código a partir de una clase base, obviando la implementación de los métodos y propiedades, y ocultando al usuario la estructura interna de programación.

### II. Objetivos específicos:

- ❖ Implementar una clase base.
- ❖ Establecer restricciones de acceso al código.
- ❖ Aprender las ventajas que ofrece el uso de la herencia y el encapsulamiento.

## 3. Introducción

El paradigma de la programación orientada a objetos surge debido a la necesidad de realizar programas que puedan ser reutilizados para evitar tener que iniciar cada proyecto nuevo desde cero. En otras palabras, no reinventar la rueda. Por lo que las bases de dicho paradigma son dos conceptos conocidos como la herencia y el encapsulamiento.

La herencia consiste en desarrollar una clase, denominada clase base, pensando en que se cederán todos sus atributos, propiedades y métodos a otra nueva clase, conocida como clase derivada, con la finalidad de que ésta última pueda reutilizar las características de la clase base sin necesidad de reescribir todo el código nuevamente.

El encapsulamiento consiste en desarrollar métodos y propiedades dentro de un programa que puedan ser utilizados sin la necesidad de analizar el código fuente original, de tal modo que la estructura interna de programación queda oculta al usuario; es decir que el usuario podrá utilizar el programa y sabrá cómo hacerlo, pero sin saber a ciencia cierta cómo se implementó su funcionamiento. Esto sería análogo a cómo un usuario común puede utilizar un televisor sin saber a ciencia cierta cómo se genera la imagen en la pantalla o qué se requiere internamente para poder cambiar de canal, modificar el volumen, el brillo, etc.

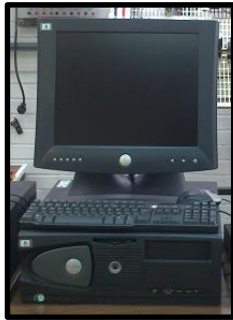


# Manual de Prácticas de Técnicas de Programación

Secretaría/División:DIMEI

Área/Departamento: Mecatrónica

## 4. Equipo y Material



Computadora

## 5. Desarrollo

### I. Actividad 1

#### Creación de una clase base

El alumno definirá y programará una clase que cuente con al menos tres atributos y los métodos necesarios para la lectura y/o escritura de los valores de cada atributo.

### II. Actividad 2

#### Creación de una clase derivada

El alumno definirá una clase derivada a partir de la clase base que definió previamente y deberá realizar un programa que demuestre que efectivamente fueron heredados todos los atributos y métodos de la clase base, sin necesidad de verificar ni de modificar el código fuente original.

Una vez comprobado lo anterior, deberá añadirse al menos un atributo nuevo a la clase derivada, así como los métodos de lectura/escritura de valores que sean necesarios.



# Manual de Prácticas de Técnicas de Programación

Secretaría/División:DIMEI

Área/Departamento: Mecatrónica

## 6. Bibliografía

- ❖ CEBALLOS SIERRA, Francisco Javier. **Microsoft C#. Curso de programación.** México, Alfaomega, 2007
- ❖ DEITEL, Harvey y Deitel, PAUL. **C# Cómo programar.** España, Pearson, 2007
- ❖ LÓPEZ ROMÁN, Leobardo. **Metodología de la programación orientada a objetos.** México, Alfaomega, 2007