



PROGRAMACION ORIENTADA A OBJETOS.

Paradigma de la Programación Orientada a Objetos

Paradigma de la programación orientación a objetos puede describirse como el conjunto de disciplinas (Ingeniería) que desarrollan y modelizan software que facilita la construcción de sistemas complejos a partir de componente.

El atractivo intuitivo de la orientación a objetos es que proporciona conceptos y herramientas con las cuales se modela y representa el mundo real tan fielmente como sea posible. Las ventajas de la orientación a objetos son muchas en programación y modelación de datos. Como apuntaba Ledbetter y Cox (1985):

La programación orientada a objetos permite una representación más directa del modelo del mundo real en el código. EL resultado es que la transformación radical normal de los requisitos del sistema (definido en términos de usuario) a la especificación del sistema (definido en términos del computador) se reduce considerablemente.

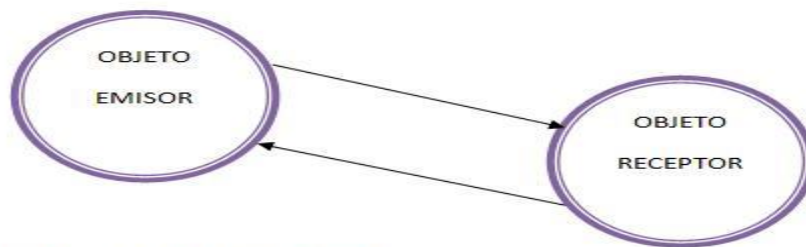
La orientación a objetos trata de cumplir las necesidades de los usuarios finales, así como las necesidades propias de los desarrolladores de los productos de software.

Estas tareas se realizan mediante la modelización del mundo real. El soporte fundamental es el modelo a objeto.

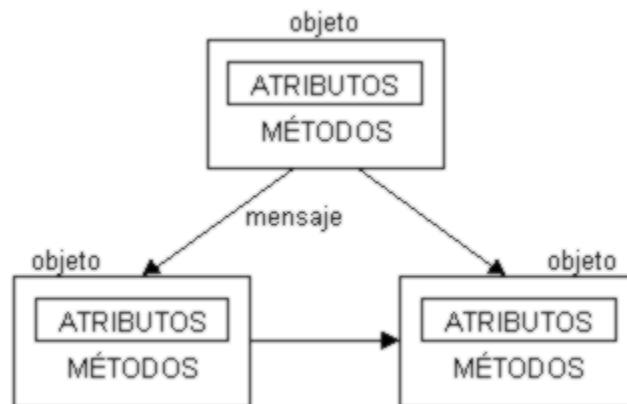
UNIDAD 1 : Introducción a la programación Orientada a Objetos



Dentro del programa de desarrollo de aplicaciones orientada a objetos cambia el enfoque de la solución. Lo importante para el paradigma procedimental o estructurado en el bloque de código principal (Instrucciones dentro de la marca un inicio y un fin). Para el paradigma orientado a objetos lo principal es entender y modelar el problema, y luego si definir en el bloque de código principal que empleando el modelo definido brinde la solución de un problema específico.



Paso de mensajes entre objetos



PILARES DE LA PROGRAMACIÓN ORIENTADA A OBJETOS



ABSTRACCIÓN: La abstracción es la propiedad que permite representar las características esenciales de un objeto, sin preocuparse de las restantes características (no esenciales).

Una abstracción se centra en la vista externa de un objeto, de modo que sirva para separar el comportamiento esencial de un objeto de su implementación. Definir una abstracción significa describir una entidad del mundo real, no importa lo compleja que pueda ser, y a continuación utilizar esta descripción en un programa.

ENCAPSULACIÓN: Es la propiedad que permite asegurar que el contenido de la información de un objeto está oculta al mundo exterior: el objeto A no conocer lo que hace el Objeto B y viceversa. La encapsulación también se conoce como la ocultación de la información, en esencia, es el proceso de ocultar todos los secretos de un objeto que no contribuyen a sus características esenciales.

UNIDAD 1 : Introducción a la programación Orientada a Objetos



La encapsulación permite la división de un programa en módulos. Estos módulos se implementan mediante clases, de forma que una clase representa la encapsulación de una abstracción.

HERENCIA : se define como relación entre clases, en donde una clase comparte una estructura o comportamiento definido en una o más clases (herencia Simple y Herencia Múltiple. El lenguaje de programación Java no soporta la herencia múltiple.

POLIMORFISMO: Polimorfismo es la capacidad que tienen los objetos de una clase de responder al mismo mensaje o evento en función de los parámetros utilizados durante su invocación. Por ejemplo, cuando se describe la clase mamífero se puede observar que la operación comer es una operación fundamental en la vida de los mamíferos, de modo de cada tipo de mamífero debe poder realizar la operación o función comer.

MODULARIDAD: La modularidad es la propiedad que permite subdividir una aplicación en partes mas pequeñas (llamadas módulos), cada una las cuales ser independiente como sea posible de la aplicación en sí y de las restantes partes.

Bibliografía:

Programación Orientada a Objetos: Conceptos, Modelado, Diseño y Codificación en C++, Autor: Luis Joyanes Aguilar.

Programación Orientada a Objetos en Java, Autor: Luis Fernando García Lilinas

**UNIDAD 1 : Introducción a la programación Orientada a
Objetos**

