HEINSOHN
BUSINESS TECHNOLOGY

Programación Orientada a Objetos

Microsoft Partner
Gold Software Development
Gold Web Development
Gold Software Asset Management

www.heinsohn.com.co

HEINSOHN
BUSINESS TECHNOLOGY

Agenda

Programación Orientada a Objetos

- Conceptos Fundamentales
- Herencia, Encapsulamiento, Polimorfismo

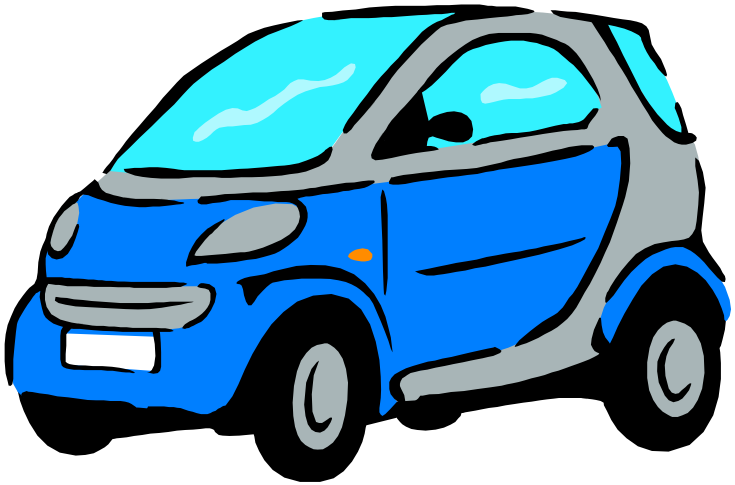
HEINSOHN
BUSINESS TECHNOLOGY

¿Qué es la Programación Orientada a Objetos?

- Es una manera de construir Software basada en un nuevo paradigma.
- Propone resolver problemas de la realidad a través de identificar objetos y relaciones de colaboración entre ellos.
- El Objeto y el mensaje son sus elementos fundamentales.

HEINSOHN
BUSINESS TECHNOLOGY

¿Qué es lo que ves?

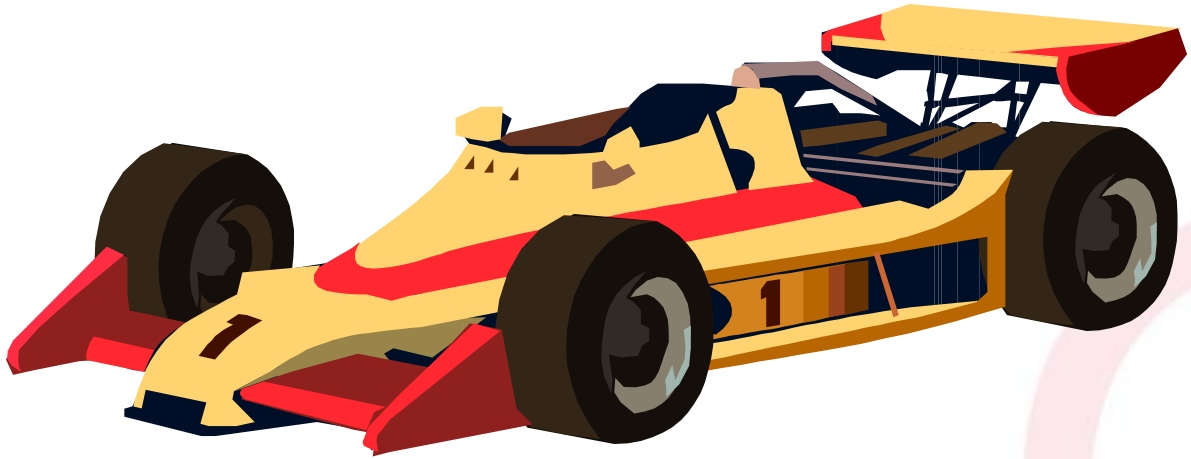


HEINSOHN
BUSINESS TECHNOLOGY

¿Qué es lo que ves?

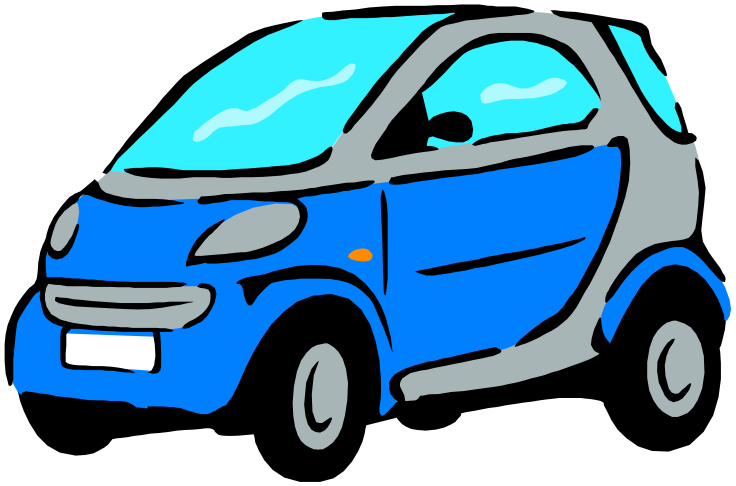
HEINSOHN
BUSINESS TECHNOLOGY

¿Qué es lo que ves?



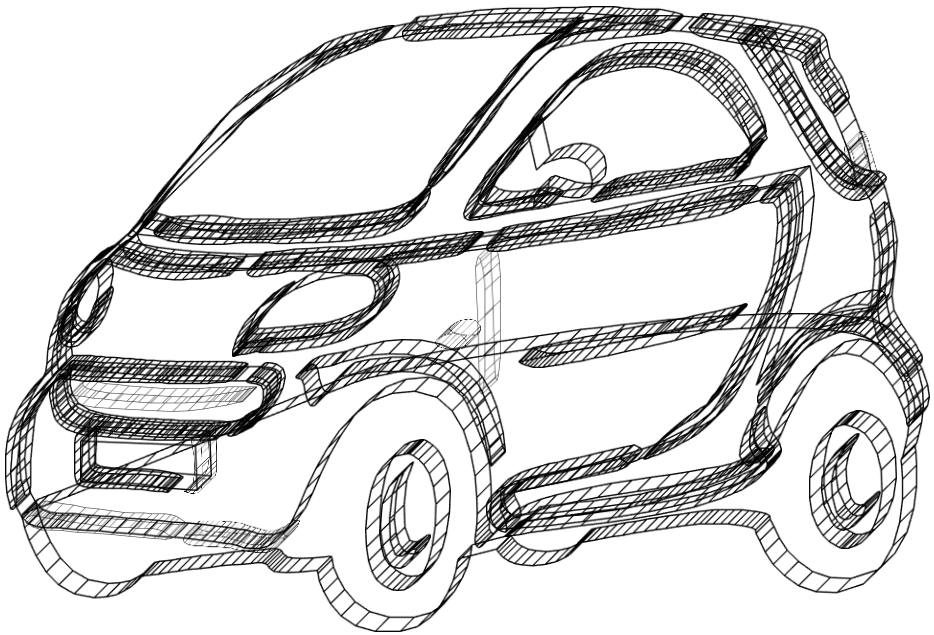
HEINSOHN
BUSINESS TECHNOLOGY

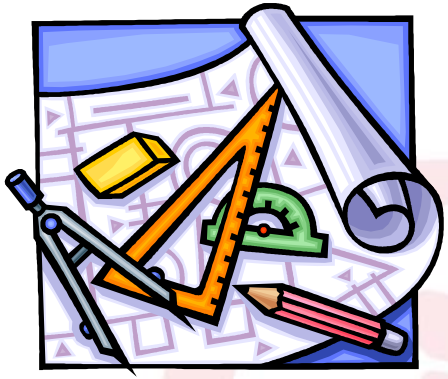
¿Qué es lo que tienen en común?



HEINSOHN
BUSINESS TECHNOLOGY

¿Qué es lo que tienen en común?





HEINSOHN
BUSINESS TECHNOLOGY

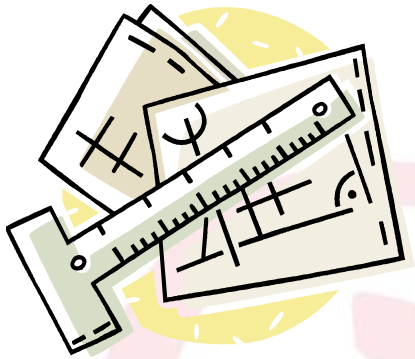
¿Qué es una clase?

- Clase → Clasificación
- Clasificación en base a comportamiento y atributos comunes
- Crea un vocabulario
  - La forma en que nos comunicamos
  - La forma en que pensamos

HEINSOHN
BUSINESS TECHNOLOGY

¿Qué es una clase?

- Construcción Estática
- Describe:
  - Comportamiento común
  - Atributos [estado]
- Estructura de datos
- Incluye:
  - Datos
  - Funciones o métodos



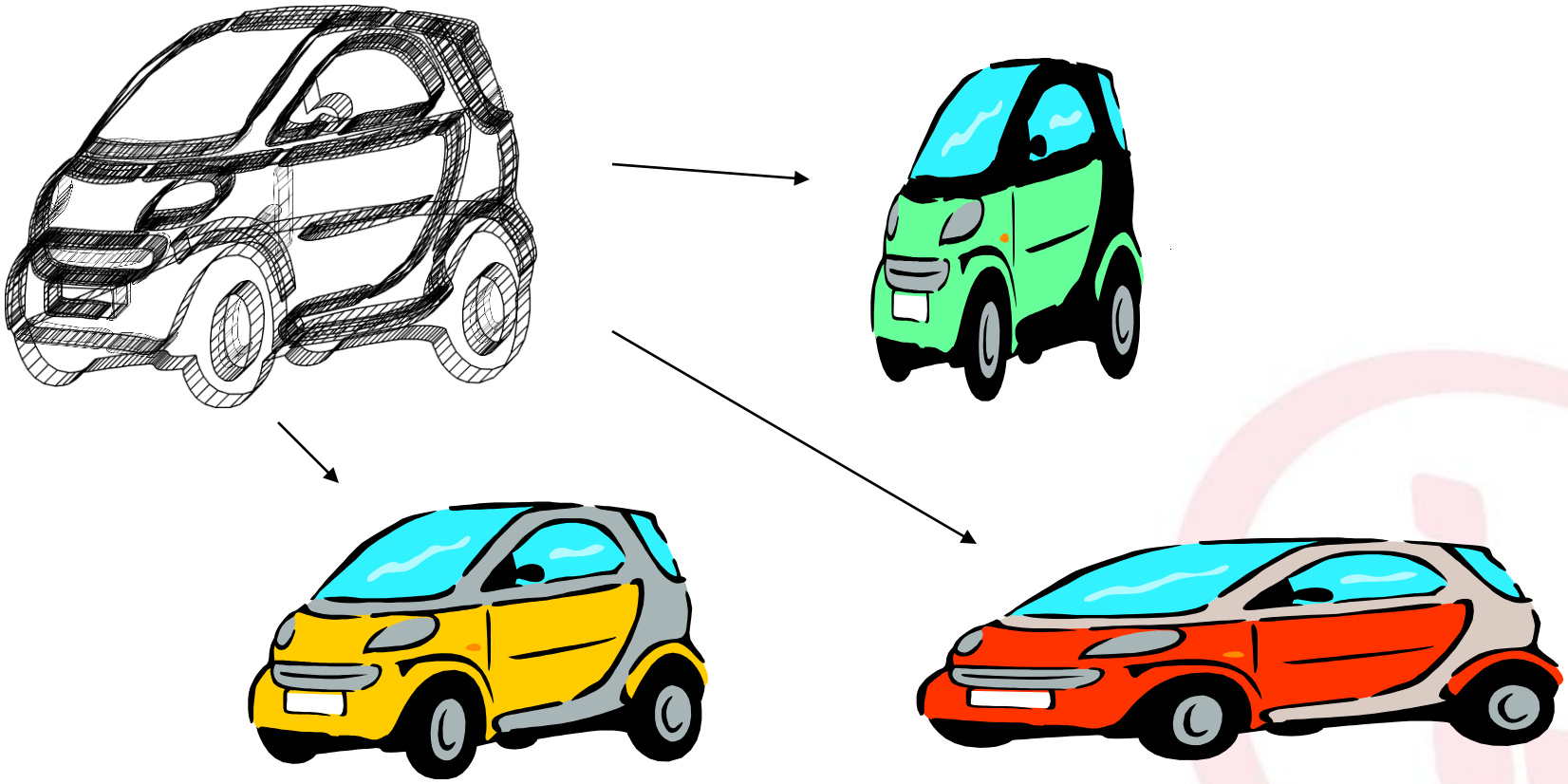
HEINSOHN
BUSINESS TECHNOLOGY

Constructor y Destructor

- Dos métodos de las clases
- Existen por defecto
- Constructor, inicializa valores
- Destructor, libera recursos al finalizar la vida de una instancia de una clase creada en memoria
- Existen constructores y destructores por defecto

HEINSOHN
BUSINESS TECHNOLOGY

¿Qué es un objeto?



HEINSOHN
BUSINESS TECHNOLOGY

¿Qué es un objeto?

- Instancia de una clase
- Un objeto posee:
  - Identidad: Relación única entre el objeto del modelo y el ente de la realidad que representa. Se implementa a través de un id único en el modelo.
  - Comportamiento: Resuelve un conjunto particular de problemas a través de su protocolo.
  - Estado: Almacena información
    - Fija
    - Variable

HEINSOHN
BUSINESS TECHNOLOGY

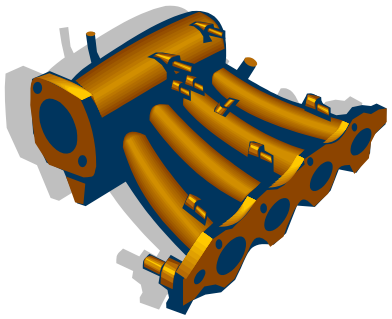
Pilares de POO

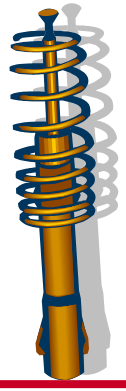
- Herencia
- Polimorfismo
- Encapsulamiento
- Abstracción

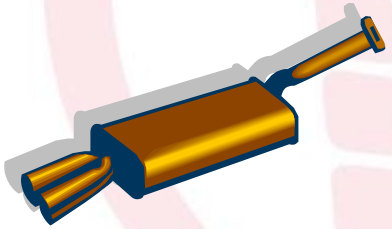
HEINSOHN
BUSINESS TECHNOLOGY

Abstracción

- Ignorancia selectiva
- Decide que es importante y que no lo es
- Se enfoca [depende] en lo que es importante
- Ignora [no depende] de lo que no es importante
- Utiliza la encapsulación para reforzar la abstracción

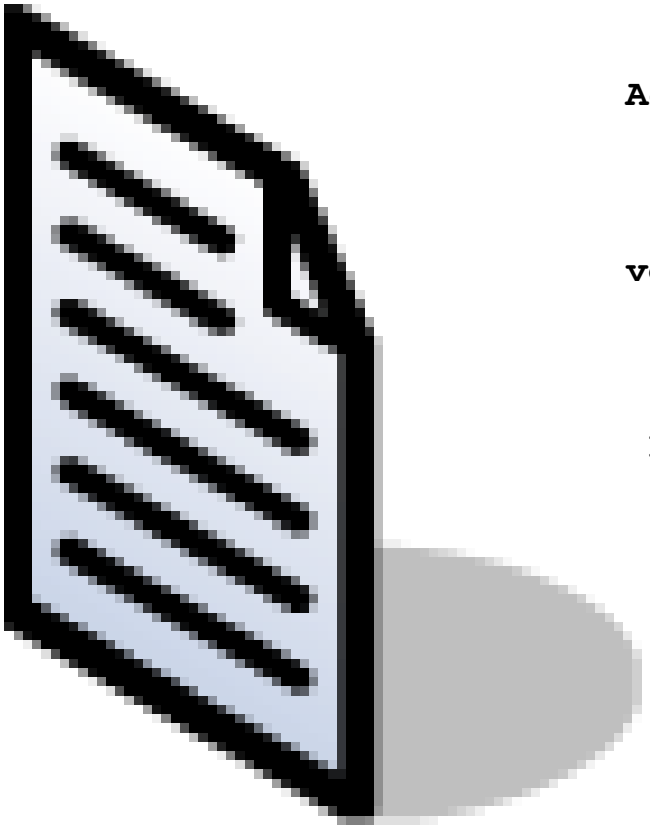






HEINSOHN
BUSINESS TECHNOLOGY

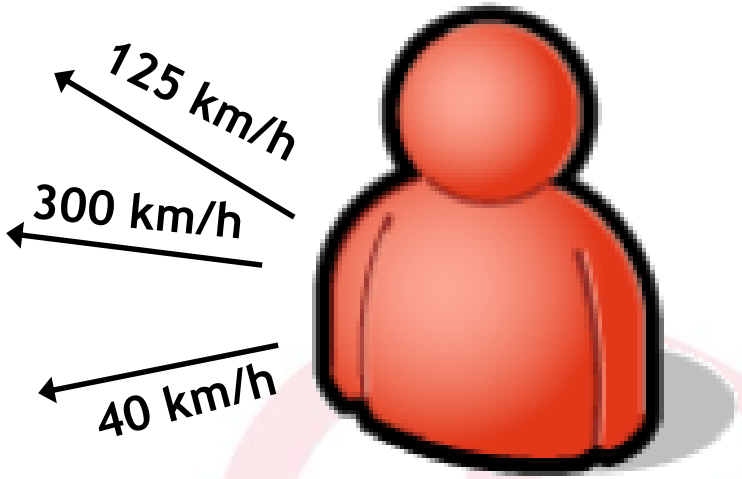
Encapsulamiento



`Acelera()`

`velocidad`

`Frena()`



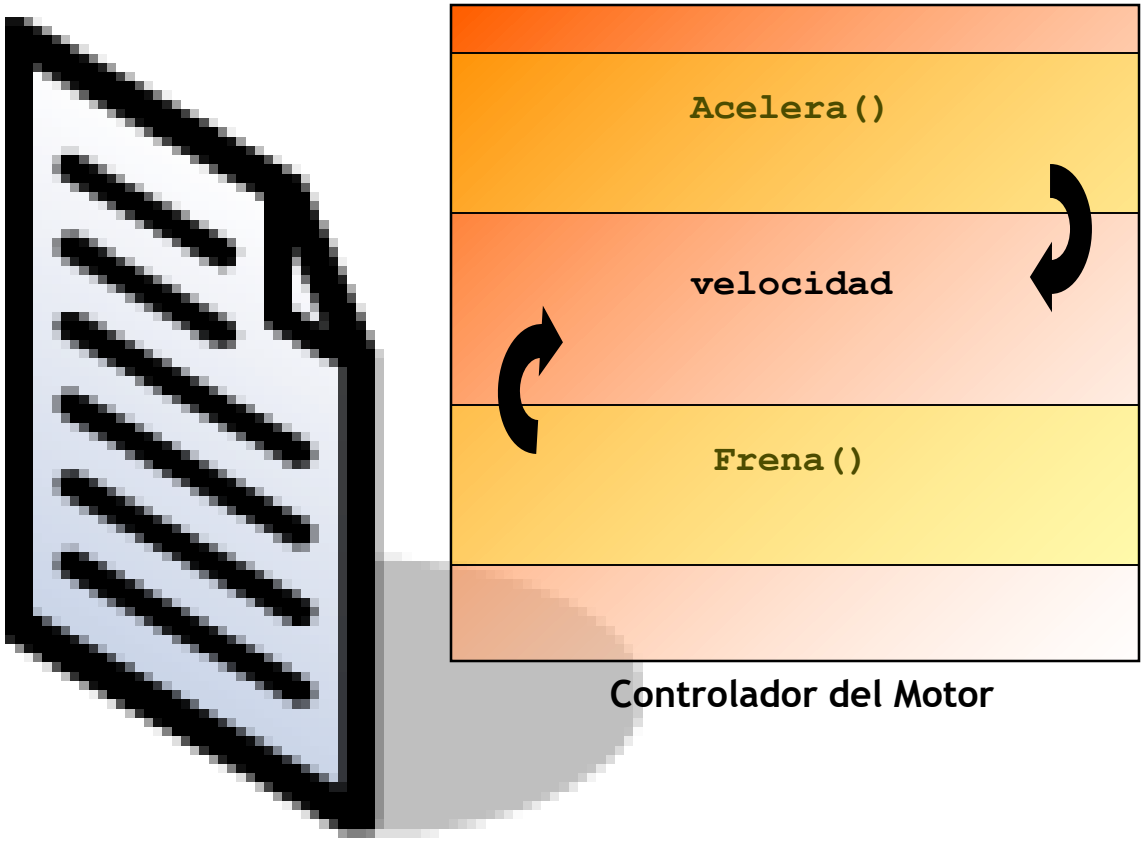
HEINSOHN
BUSINESS TECHNOLOGY

¿Por qué utilizar encapsulamiento?

- Dos grandes razones
  - Privacidad
  - Control

HEINSOHN
BUSINESS TECHNOLOGY

Encapsulamiento



- Los métodos son públicos: son accesibles desde fuera
- Los datos son privados: accesibles desde dentro

Datos del Objeto

- Describe los objetos de forma individual

- **Color: Azul**
- **Color 2: Gris**
- **Instrumental Digital**
- **Ventanas eléctricas**
- **4 asientos**

- **Color: Verde**
- **Color 2: Negro**
- **Instrumental digital**
- **2 asientos**

Datos Estáticos

- Describen información para todos los objetos

- **Número de llantas: 4**
- **Capacidad del tanque: 40 litros**
- **Tipo de Transmisión: Automática**

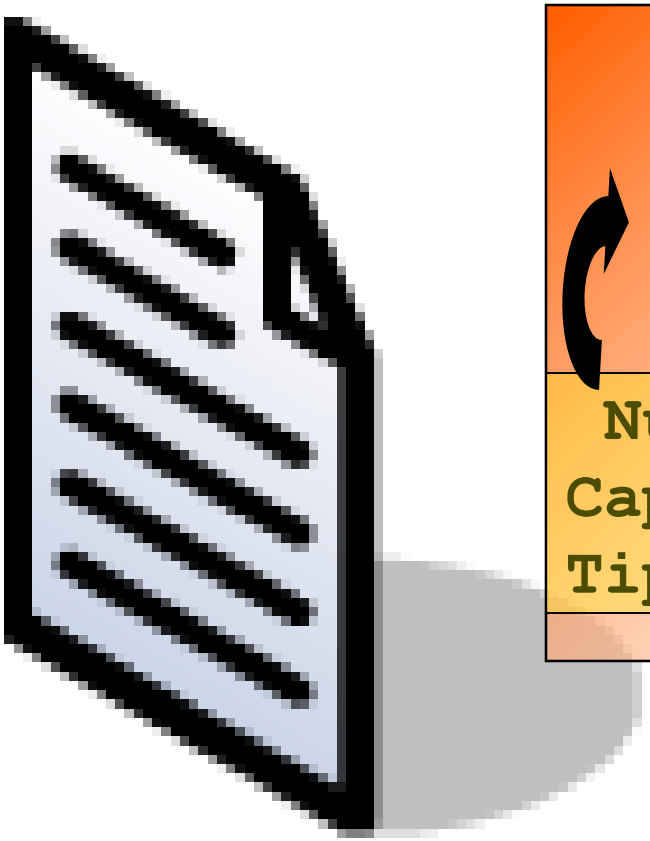
HEINSOHN
BUSINESS TECHNOLOGY

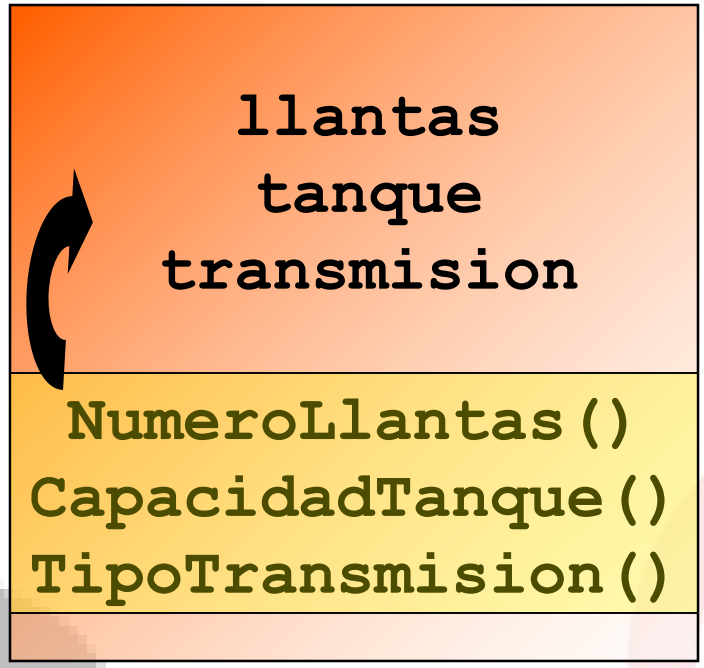
Métodos Estáticos

- Solo pueden acceder a datos compartidos por todas las instancias de la clase.
- Encapsula los datos estáticos
- Son invocados en la clase, no en el objeto
  - No es necesaria la creación de una instancia para invocarlos

HEINSOHN
BUSINESS TECHNOLOGY

Métodos Estáticos



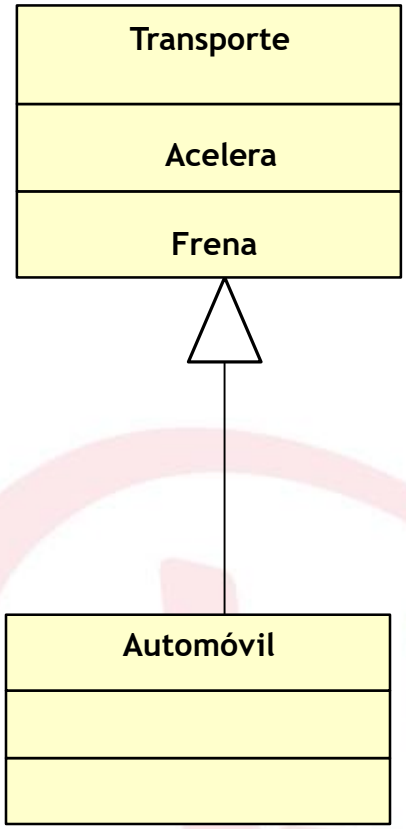


Automóvil

HEINSOHN
BUSINESS TECHNOLOGY

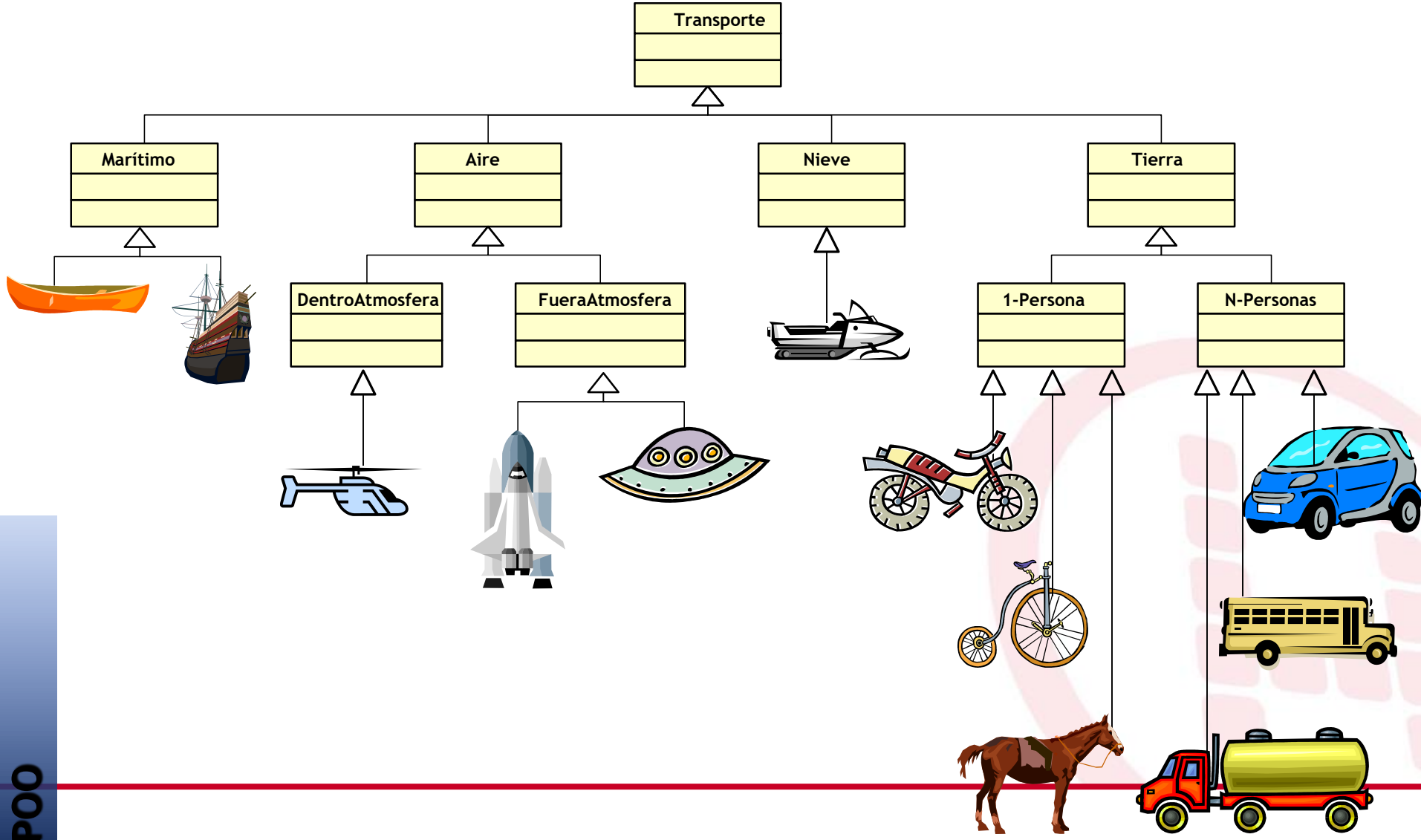
Herencia

- Es “un tipo de” relación
  - Relación “es un”
- Entre Clases
- Clase base
- Clase derivada
- Hereda la implementación



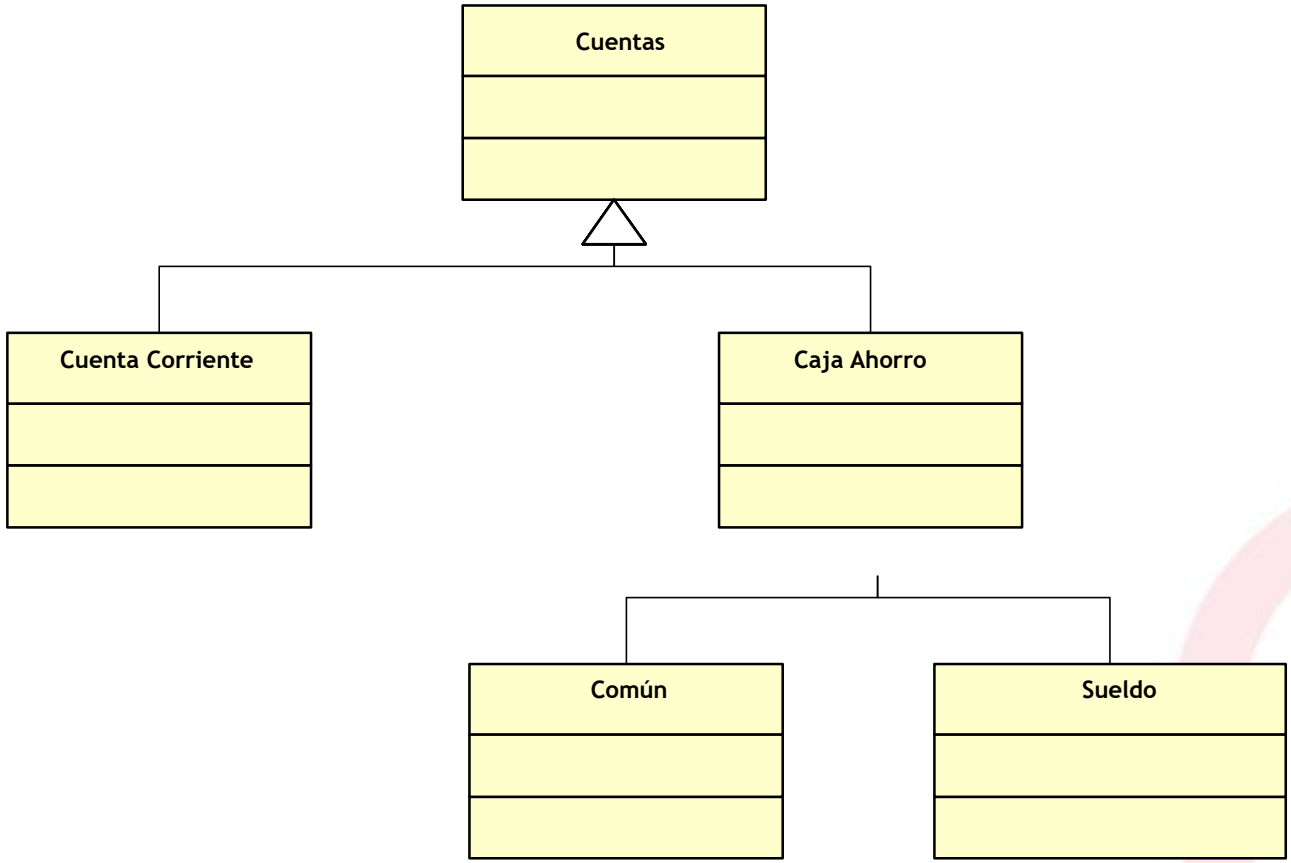
HEINSOHN
BUSINESS TECHNOLOGY

Jerarquías de Clase



HEINSOHN
BUSINESS TECHNOLOGY

Jerarquías de Clase - Otro ejemplo



HEINSOHN
BUSINESS TECHNOLOGY

Polimorfismo y sobrecarga

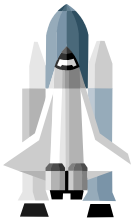
- Polimorfismo: desconocimiento del cliente sobre la clase concreta del objeto que brinda el servicio. Esto me da libertad de intercambiar libremente el objeto servidor.
- Sobrecarga: Definir más de un método por cada mensaje, los tipos de los argumentos ayudan a decidir a qué mensaje se invoca.
- Tareas similares son realizadas por métodos con mismo nombre
  - Suma
    - Enteros
    - Decimales
    - Fracciones
- Simplifican la tarea del desarrollador, al no tener que recordar distintos nombres para comportamientos iguales.

Polimorfismo

- La definición del método reside en la clase base
- La implementación del método reside en la clase derivada
- La invocación es resuelta al momento de ejecución

Auto
Acelera
Frena

Transporte
Acelera
Frena

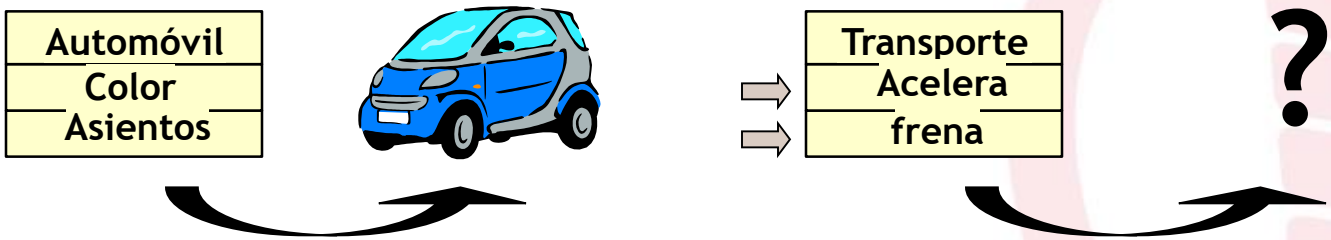


Cohete
Acelera
Frena

Caballo
Acelera
Frena

Clases Base Abstractas

- Existen solamente para que se deriven de ellas
  - No tiene sentido crear una instancia de este tipo de clases
- Métodos abstractos
- Clases abstractas – Concrete classes



HEINSOHN
BUSINESS TECHNOLOGY

Resumen

- ¿Qué es la Programación Orientada a Objetos?
- Clases y Objetos
- Métodos y Atributos
- Encapsulamiento
- Herencia
- Polimorfismo

HEINSOHN
BUSINESS TECHNOLOGY