



Introduzione OOP

Modulo 226A

Implementare in base alle classi (senza ereditarietà)

Paradigmi di programmazione

- **Programmazione Procedurale (o imperativa)**
- Programmazione Funzionale
- Programmazione Logica
- **Programmazione Object-Oriented**

Procedurale

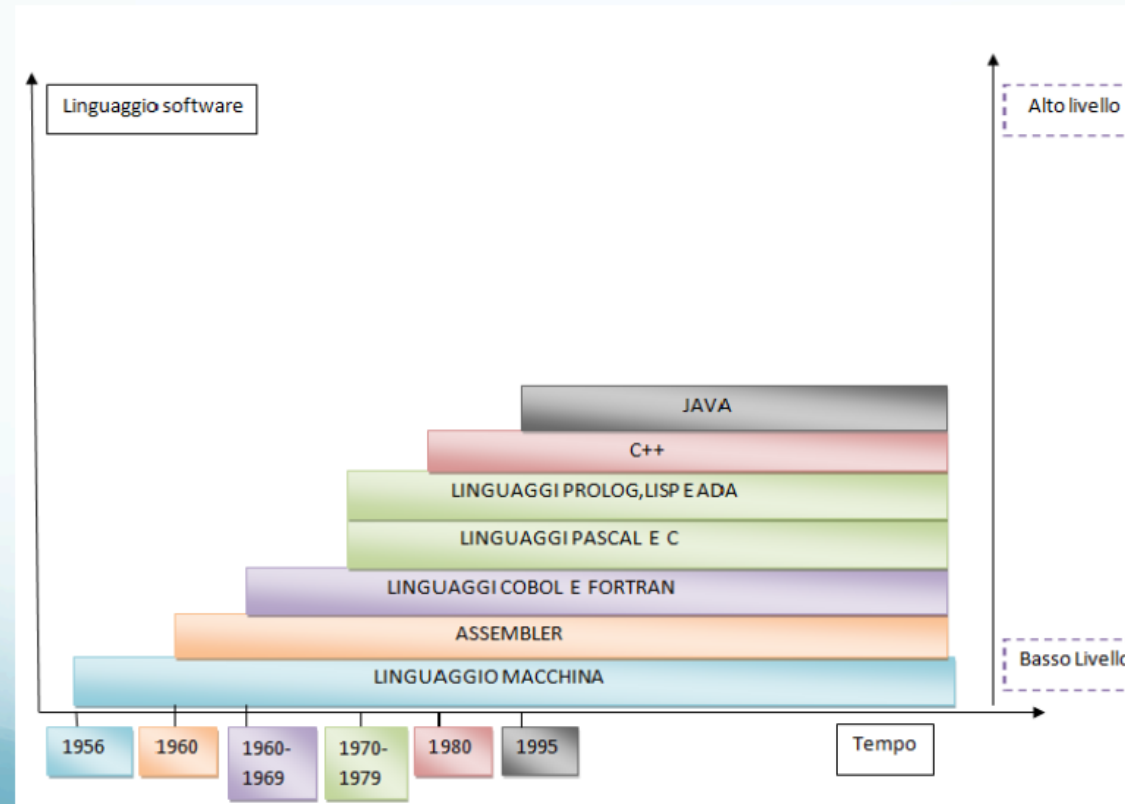
- Programmazione basata sul principio “divide et impera”
- Si focalizza sul problema, definendo procedure che interagiscono tra di loro.
- Ad esempio il C è un linguaggio "puramente" procedurale

Procedurale vs OOP

Procedurale	OOP
Si progettano le procedure	Si progettano gli oggetti
Si impone loro di scambiare dati	Si impone loro di interagire tramite procedure

Evoluzione

Evoluzione dei linguaggi di programmazione



Un pò di storia

- Il paradigma computazionale object oriented (OOP) nacque nel 1969 ad opera del norvegese Nygaard
- simulazione del movimento delle navi nei fiordi
- Difficolta nel simulare maree, movimenti delle navi, forme delle linee di costa con i tradizionali metodi di programmazione
- Più facile concepire gli elementi da modellare come *oggetti* dotati di dati e funzioni proprie

Definizione OOP

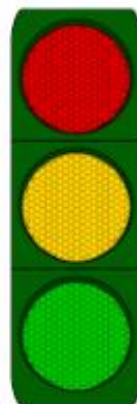
- Un programma è costituito da oggetti che interagiscono tra loro.
- Il dato viene visto come un tipo di dato astratto (**ADT**) caratterizzato da un insieme di valori che lo definiscono e un insieme di operazioni che possono essere applicate a esso.
- Modularità e riusabilità del software
- Facilita la gestione e la manutenzione di progetti.
- Qualsiasi modello include gli aspetti più importanti o essenziali di qualcosa mentre ignora i dettagli meno importanti.
- Enfatizzare gli aspetti comuni.

Oggetti

- Gli oggetti sono entità (reali o astratte) che troviamo nel mondo reale che ci circonda.
- Componenti hardware/software, documenti, esseri umani, veicoli, alberi, ogni elemento del mondo può essere rappresentato attraverso un oggetto.

Durante un viaggio in macchina quali oggetti interagiscono tra di loro?





Un altro esempio

Un ingegnere automobilistico vedrebbe come oggetti le ruote, il motore, il volante e gli altri componenti delle automobili che progetta.

OGGETTO



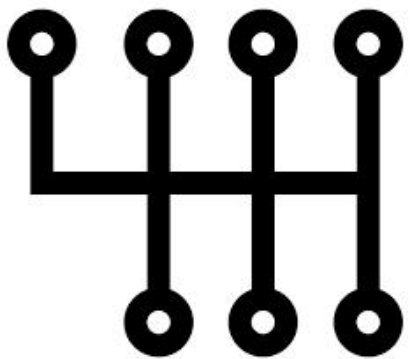
OGGETTO



OGGETTO



OGGETTO



OGGETTO



Classi

L'elemento principale nella OOP è la classe, una descrizione astratta di un tipo di dato (**ADT**) che descrive una famiglia di oggetti con caratteristiche e comportamenti simili.

Generalmente le classi sono costituite da due categorie di membri: gli **attributi** e i **metodi**.

Esempio Automobile

Car class



Automobile
marca
modello
colore
potenza
marce
peso
<u>accellera()</u>
<u>frena()</u>
<u>cambiaMarcia()</u>
<u>rivernicia()</u>

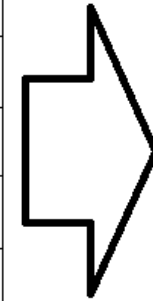
Attributi

- Una classe è costituita da attributi che specificano le caratteristiche che tutti gli oggetti della classe devono avere.
- I valori che assumono gli attributi in un certo istante determinano lo stato del singolo oggetto della classe. Se cambia anche un solo valore di una proprietà di un oggetto il suo stato varierà di conseguenza.

Car class



Automobile
marca
modello
colore
potenza
marce
peso



ATTRIBUTI

Metodi

I metodi costituiscono le azioni che possono essere compiute da un oggetto appartenente a quella classe.

Car class



Automobile

marca

modello

colore

potenza

marce

peso

accelera()

frena()

cambiaMarcia()

rivernicia()

ATTRIBUTI

METODI

Esempio

Nella classe *Automobile* i seguenti metodi definiscono le azioni che un oggetto può compiere:

- accelera()
- frena()
- cambiaMarcia()
- cambiaColore()

Caratteristiche

I metodi di un oggetto possono modificare lo stato dell'oggetto stesso.

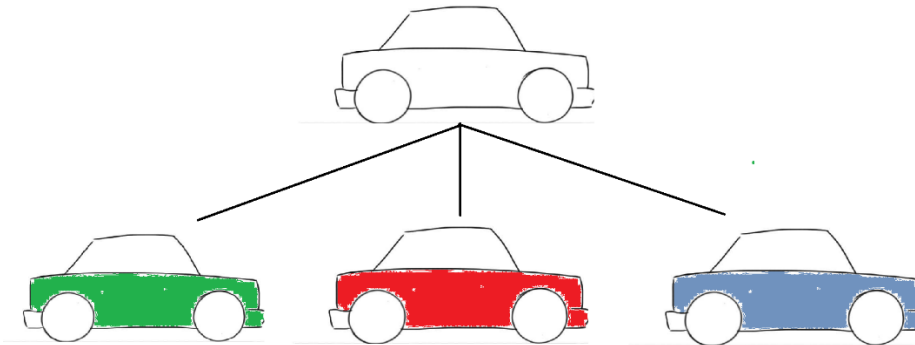
Nel caso dell'automobile, quando riverniciamo l'auto il colore cambia.

Possiamo realizzare questa regola cambiando il valore dell'attributo colore ogni volta che viene eseguito il metodo `cambiaColore()`.

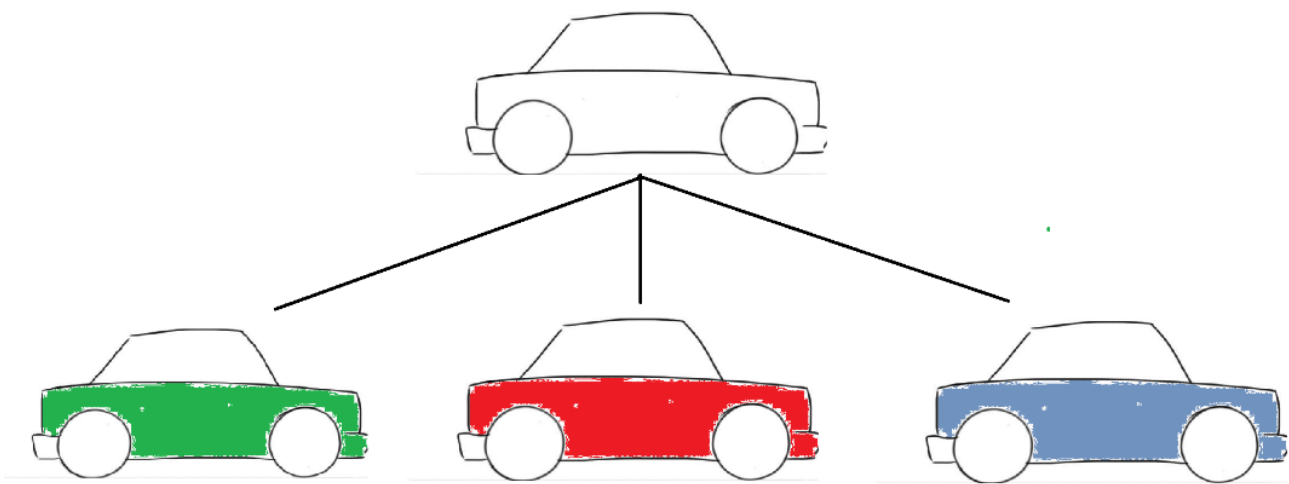
Istanze

Tutti gli oggetti di uno stesso tipo formano una classe di oggetti.
Tecnicamente ogni oggetto viene definito istanza di una classe.

Class Automobile



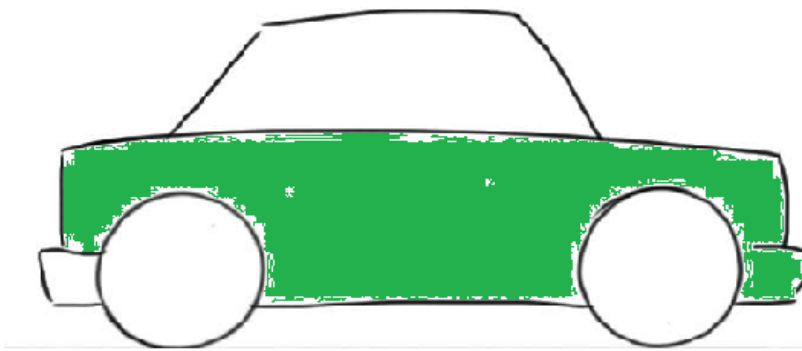
CLASS *Automobile*



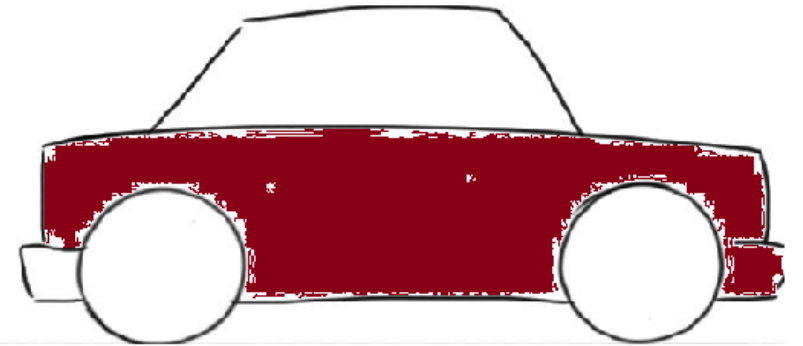
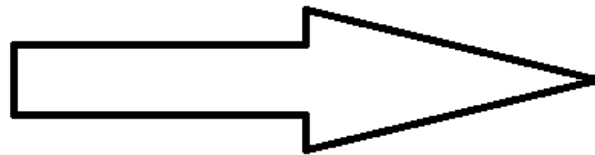
Istanza	Istanza	Istanza
Marca: Ford	Marca: Ferrari	Marca: Toyota
Modello: Sierra	Modello: GTO	Modello: Prius
Colore: verde	Colore: Rosso	Colore: Blu
Potenza: 204cv	Potenza: 400cv	Potenza: 180cv
Marce: 5	Marce: 5	Marce: 5
Peso: 1100 Kg	Peso: 1100 Kg	Peso: 1100 Kg

Esempio cambiaColore

Riprendendo l'esempio della riverniciatura dell'automobile possiamo eseguire il metodo `cambiaColore()` sull'oggetto (istanza) *Ford Sierra*.



cambiaColore()



Esercizio in classe

Descrivi la classe Rettangolo che rappresenta il modello astratto di un rettangolo.

La classe deve possedere gli attributi pubblici latoMaggiore e latoMinore (float) per descrivere la dimensione del rettangolo.

La classe deve implementare i metodi di istanza:

- calcolare l'area del rettangolo;
- calcolare il perimetro del rettangolo;
- disegnare a video il rettangolo; sotto forma di asterischi;
- stampare la dimensione del rettangolo (es. 5x5);
- stampare a video l'area e il perimetro del rettangolo.