

# OGGETTI

## CREARE UN OGGETTO

La base dell'Object Programming e' classificare degli oggetti in base a delle caratteristiche comuni.

*La classificazione si effettua tramite il comando **class***

```
class Person{
    protected $name;
    public $surname;
    public $age;
    public $height;
    public static $count = 0;

    public function __construct($nome, $cognome, $eta){
        $this->name = $nome;
        $this->surname = $cognome;
        $this->age = $eta;
        $this->height = rand(160, 200);
        self::$count++;
    }

    public function presentati(){
        echo "Ciao sono $this->surname, $this->name $this->surname";
    }

    public function dimmiLaTuaEta(){
        echo "sono vecchietto, perche ho $this->age anni";
    }
    protected function changeName(){
        $this->name = "Marco";
    }
}

$francesco = new Person("Francesco", "Talamona", 35);
$francesco->presentati();
print_r($francesco);
print_r($francesco->name);
$francesco->dimmiLaTuaEta();
print_r(Person::$count);

//      Ciao sono Talamona, Francesco Talamona

//      Person Object
//(
//      [name] => Francesco
//      [surname] => Talamona
//      [age] => 35
//)

//      Francesco
//      sono vecchietto perchè ho 35 anni
//      1 (count)
```

Queste caratteristiche le avranno tutte le **new Person** di questa classe.  
Queste caratteristiche si chiamano “arguments”, caratteristiche intrinseche di ogni persona.

**class Person{ }** il nome della classe richiede sempre la prima lettera Maiuscola.

**public** specifica che possiamo modificare e accedere ad una proprietà dall'esterno.

**protected** specifica che non possiamo modificare la proprietà dall'esterno, ma solo dall'interno o dalle classi figlie. Stessa cosa vale per le funzioni dichiarate così'.

Devo istruire la classe a darmi degli oggetti. Attraverso la funzione  
**\_\_construct**(parametroFormale1, parametroFormale2, parametroFormale3)  
indichiamo dei parametri formali, e successivamente utilizziamo  
**\$this->** per assegnare il parametro ad una proprietà della classe.

=> chiave valore  
-> proprietà this dell'**oggetto**

**\$nomeVariabileObj = new Person(...)** per creare un nuovo oggetto, assegnamo ad una variabile la funzione tramite il termine **new**.

**\$nomeVariabileObj->presentati();** possiamo creare funzioni all'interno della classe che e' possibile richiamare con specificando l'oggetto creato in precedenza

**\$nomeVariabileObj->name = "Marco";** possiamo modificare un valore della classe dall'esterno se la proprietà e' **public**

**print\_r(\$nomeVariabileObj->name);** posso accedere alle proprietà o metodo di un oggetto

**\$this->height = rand(160, 200);** possiamo anche assegnare delle proprietà che non necessitano di essere esplicitate tramite un parametroFormale. In questo caso dichiariamo un valore random che verra' assegnato ad ogni oggetto che creiamo.

**public static \$count = 0;** con static dichiariamo che la proprietà e' della classe e non dell'oggetto

**self::\$count++;** con self:: richiamiamo nella funzione una proprietà statica nella funzione, e in questo caso aumentiamo di 1 il count della proprietà statica ad ogni creazione di un oggetto.

**print\_r(Person::\$count);** in questo modo si richiama a Terminale una proprietà statica