

به نام خدا



برنامه نویسی شی گرا در PHP

Object-Oriented Programming in PHP

Mostafa Jahangir (PhD)

www.mjahangir.ir

mjahangir@varastegan.ac.ir

برنامه نویسی شی گرا در PHP

Object-oriented Programming in PHP

مفهوم برنامه نویسی شی گرا

- یک شیوه برنامه نویسی که کدهای مشابه را در قالب یک کلاس **Class** جمع آوری و در قسمت های مختلف برنامه در قالب شی **Object** از آن استفاده می کند.
- فرض کنید یک وب سایت طراحی کرده اید که برای قسمت های مختلف آن مانند اخبار، مقالات و ... امکان ارسال نظرات برای کاربر وجود دارد.
- خب در برنامه نویسی به شیوه معمولی شما می بایست برای هر قسمت سایت نظردهی جداگانه ای بنویسید.
- اما در برنامه نویسی شی گرایی شما یک بار در قالب یک کلاس **class** سیستم نظردهی را می نویسید و برای صفحات مختلف یک شی **object** از کلاس نظردهی ایجاد می کنید.
- به این ترتیب شما فقط یک بار سیستم نظردهی را نوشته اید و از آن در تمامی قسمت های سایت استفاده می کنید.

مفهوم برنامه نویسی شی گرا

- در طبیعت نیز از مدل شی گرایی برای تمامی پدیده ها نیز استفاده شده است.
- مثلاً یک کلاس کلی از انسان وجود دارد که دارای اجزای مشترک مانند قلب، چشم، گوش و ... می باشد.
- و شی های مختلف از این کلاس ساخته می شوند.

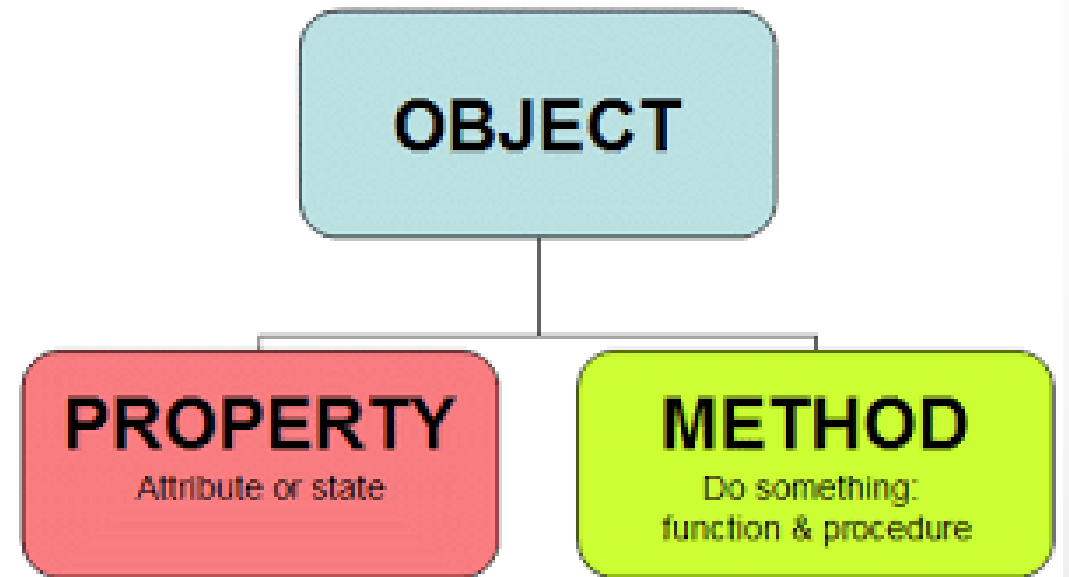
چرا برنامه نویسی شی گرا؟

- پرهیز از تکرار کدنویسی
- توسعه آسان تر برنامه های نوشته شده
- اگر خطایی در برنامه ایجاد شود و یا در آینده قصد توسعه برنامه را داشته باشید، فقط کافیست کدهای کلاس مربوطه را تغییر دهید و مجبور به تغییر کل برنامه از ابتدا تا انتها نخواهید بود.

مفهوم و ساختار Class

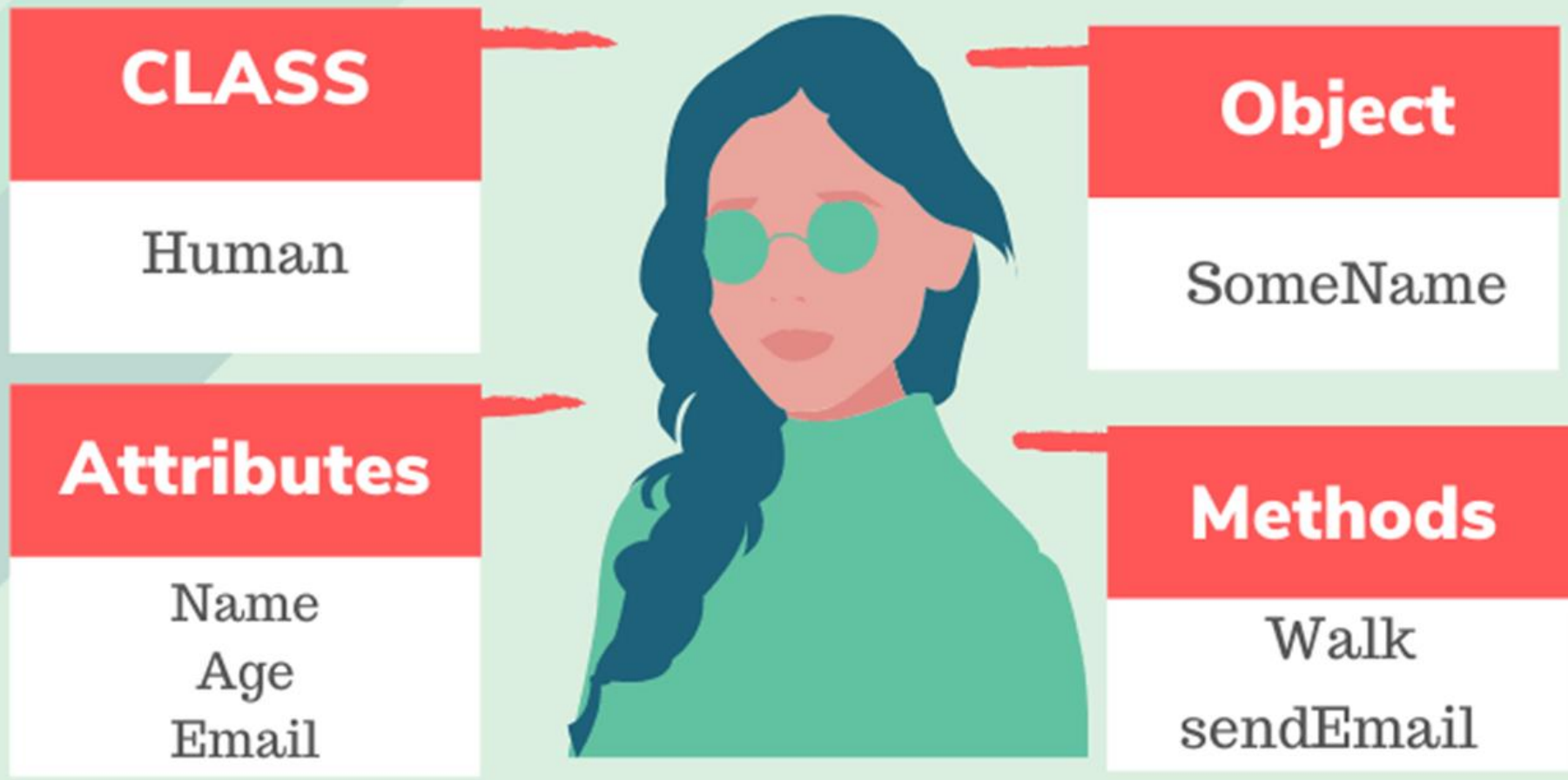
- Class شامل دو مورد زیر است:
- Property: ویژگی ها، خصوصیات و متغیرهای Class
- Method: رفتار و کارهایی است که یک Class انجام می دهد
- ساختار Class

```
class classname{  
    Property1;  
    Property2;  
    ...  
    method1;  
    method2;  
    ...  
}
```



Object Oriented Programming

HolyCoders.com



مثالی از یک Class

```
class Car {  
    public $color = "آبی";  
    public $sunroof = true;  
}
```

- کلاس خودرو
- کلیدواژه public را قبل از نام property می نویسیم تا بگوییم از این property عمومی و در جاهای مختلف قابل استفاده است.
- یک Property می تواند مقدار پیش فرض داشته باشد که در همان خط تعریف و به آن اختصاص داده می شود. مانند رنگ آبی که به Property، \$color تخصیص داده شده است.

نحوه ساخت object از روی یک Class

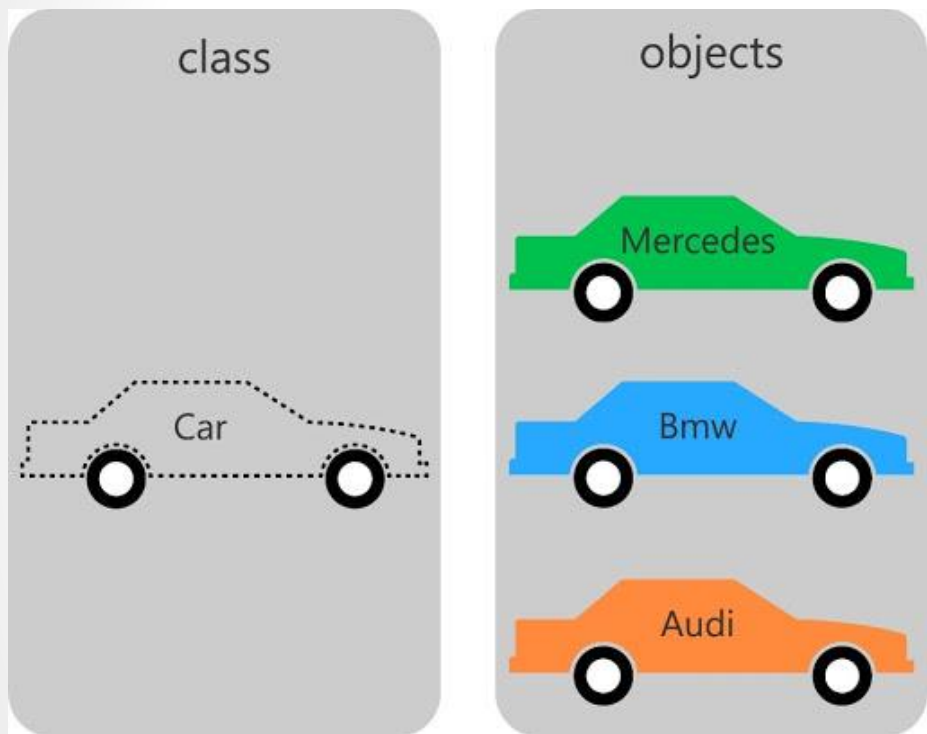
- به منظور استفاده از یک Class می بایست ابتدا یک شی object از آن ایجاد کنید.
- می توانید از روی یک Class چندین شی object ایجاد کنید.
- جهت ساخت شیء از روی کلاس از کلیدواژه new استفاده کنید:

```
$bmw = new Car ();
```

```
$mercedes = new Car();
```

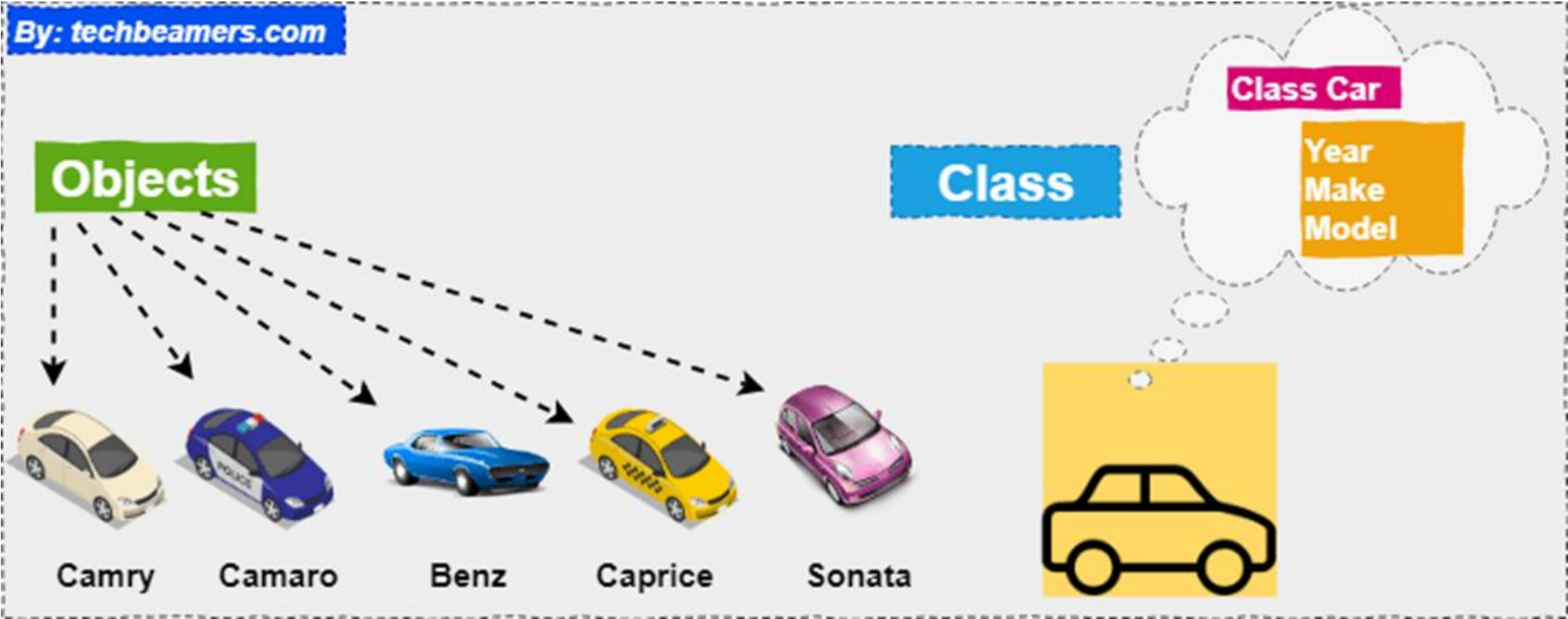
```
$car1=new Car();
```

مثالی از Class و Object

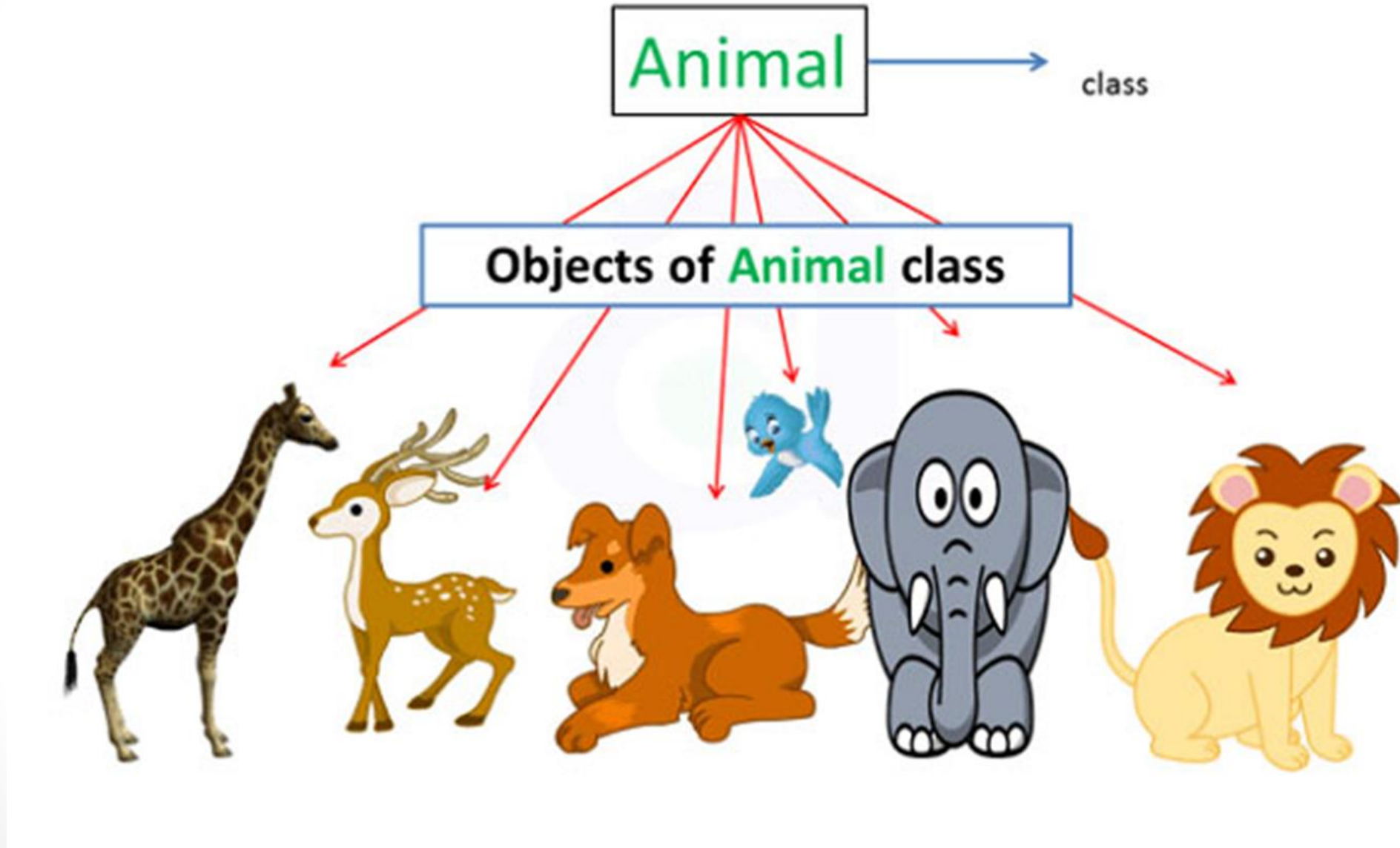


- از کلاسی واحد به نام Car، سه Object مستقل به ترتیب به نام های Mercedes، Bmw و Audi ایجاد کردیم.
- تمامی شی‌های فوق Method و property های کلاس Car را دارا می باشند.
- با این وجود هر سه شیء از یکدیگر متفاوت و مستقل هستند.
- در واقع نه تنها دارای اسم متفاوت اند، بلکه به property های هر یک مقدار متفاوتی تخصیص داده شده است.
- در نمونه فوق، سه شیء از نظر رنگ با یکدیگر متفاوت هستند.

Object و Class



مثالی از Class و Object



چگونگی دسترسی به مقدار property های یک شی

- پس از ایجاد شیء از روی کلاس، می توان به property های آن دسترسی داشت. برای مثال:

```
echo $bmw -> color;
```

```
echo $mercedes -> color;
```

- به منظور دسترسی به مقدار یک property، ابتدا اسم شی، عملگر `->` و سپس اسم property را ذکر می کنیم.

- همان طور که مشاهده می کنید، اسم property با علامت `$` آغاز نمی شود، بلکه این علامت فقط در ابتدای اسم شی درج می گردد.

نحوه مقداردهی به property های شی

- به منظور مقداردهی property رنگ از کد زیر استفاده کنید

```
$bmw -> color = "آبی";
```

- به منظور نمایش مقدار متغیر color از شی \$bmw، از دستور زیر استفاده کنید:

```
echo $bmw -> color;
```

نحوه تعریف متد در کلاس

- کلاس ها معمولا علاوه بر Property، Method هایی نیز دارند که کارهایی را انجام داده و رفتارهای آن کلاس محسوب می شوند.
- در کد زیر می بینید که متد hello() را با سطح دسترسی عمومی public تعریف کردیم.

```
class Car {  
    public $color = 'آبی';  
    public $sunroof = true;  
    public function hello() {  
        return "beep";  
    }  
}
```

شیوه دسترسی به Method

- شیوه دسترسی به Method نیز مانند دسترسی به property های کلاس می باشد.
- اما قبل از هر چیز لازم است یک شیء از روی کلاس ایجاد کنیم.

```
$bmw = new Car ();
```

```
$mercedes = new Car ();
```

```
echo $bmw -> hello();
```

```
echo $mercedes -> hello();
```


مثال: نمایش قد و وزن به روش شی گرا

```
1 ▼ <html dir="rtl">
2 ▼ <head>
3     <meta charset="utf-8">
4     <title>Define class & object</title>
5 </head>
6 ▼ <body>
7 <h4>محاسبه BMI:</h4>
8 <?php
9 ▼     class bmi{
10         public $weight=76;
11         public $height=174;
12     }
13
14     $bmi1=new bmi;
15
16     echo 'وزن برابر است با ' . $bmi1->weight . '<br>';
17     echo 'قد برابر است با ' . $bmi1->height . '<br>';
18     ?>
19 </body>
20 </html>
```

مثال: محاسبه BMI به روش شی گرا

```
1 <html dir="rtl">
2 <head>
3     <meta charset="utf-8">
4     <title>Initialize to objects</title>
5
6 </head>
7 <body>
8 <h4>محاسبه BMI:</h4>
9 <?php
10     class bmi{
11         public $weight=87;
12         public $height=1.35;
13         public $bmi_calc;
14         public function bmi_calc (){
15             return 87/(1.35*1.35);
16         }
17     }
18
19     $bmi1=new bmi;
20
21     echo 'وزن برابر است با ' . $bmi1->weight . '<br>';
22     echo 'قد برابر است با ' . $bmi1->height . '<br>';
23     echo 'شاخص توده بدنی برابر است با ' . $bmi1->bmi_calc() . '<br>';
24 >
25 </body>
26 </html>
```