

## PHP - Moștenire și modificador access protejat

În capitolul anterior am aflat că **proprietățile sau metodele protejate** pot fi accesate în cadrul **clasei** și prin **clase derivate** din acea **clasă**. Ce înseamnă asta?

Să ne uităm la un exemplu:

### Exemplu:

```
<?php
class Fruit {
    public $name;
    public $color;
    public function __construct($name, $color) {
        $this->name = $name;
        $this->color = $color;
    }
    protected function intro() {
        echo "Fructul este {$this->name} și culoarea este {$this->color}.";
    }
}

class Strawberry extends Fruit {
    public function message() {
        echo "Sunt un fruit sau o berry? ";
    }
}

// Încercați să apelați toate cele trei metode din afara clasei
$strawberry = new Strawberry("Strawberry", "red"); // OK. __construct() este public
$strawberry->message(); // OK. message() este public
$strawberry->intro(); // ERROR. intro() este protejat
?>
```

În exemplul de mai sus vedem că dacă încercăm să apelăm la o **metodă protejată (intro())** din afara **clasei**, vom primi o **eroare**. **Metodele publice** vor funcționa bine!

Să ne uităm la un alt exemplu:

### Exemplu:

```
<?php
class Fruit {
    public $name;
    public $color;
    public function __construct($name, $color) {
```

```
$this->name = $name;
$this->color = $color;
}
protected function intro() {
    echo "Fructul este{$this->name} și culoarea este {$this->color}.";
}
}
```

```
class Strawberry extends Fruit {
    public function message() {
        echo "Sunt un fruit sau o berry? ";
        // Apelați metoda protejată din cadrul clasei derivate - OK
        $this -> intro();
    }
}
```

```
$strawberry = new Strawberry("Strawberry", "red"); // OK. __construct() este public
$strawberry->message(); // OK. message() este public și apelează intro()
(care este protejat) din cadrul clasei derivate
?>
```

În exemplul de mai sus vedem că toate funcționează bine! Se datorează faptului că numim **metoda protejată (intro())** din interiorul **clasei derivate**.