

# PROGRAMMATION ORIENTÉE OBJET : VISIBILITÉ

<https://www.php.net/manual/fr/oop5.intro.php>



## Visibilité : public, private ou protected

- En programmation orientée objet (POO), la visibilité des propriétés et des méthodes détermine comment elles peuvent être accessibles dans le code. En PHP, il existe trois niveaux de visibilité : **public**, **protected** et **private**.
- Ces niveaux de visibilité permettent de contrôler l'accès aux données et aux fonctionnalités d'une classe, favorisant ainsi l'encapsulation et la protection des données.

## Visibilité : public, private ou protected

- **Public** : Accessible partout (dans la classe, les classes enfants, et à l'extérieur).
- **Protected** : Accessible dans la classe et les classes enfants.
- **Private** : Accessible uniquement dans la classe où ils sont définis.

## Visibilité : public

- Les membres (propriétés et méthodes) déclarés avec le mot-clé public sont accessibles partout. Cela signifie qu'ils peuvent être accédés depuis n'importe où, y compris à l'extérieur de la classe.

# Visibilité : public

## ■ Classe : Plante.class.php

```
Plante.class.php X
PUBLIC > Plante.class.php > PHP Intelephense > Plante
1 <?php
2
3 class Plante{
4     public $nom;
5     public $type;
6     public $hauteur;
7     public $dureeDeVie;
8     public $famille;
9
10    public function __construct($nom,$type,$hauteur,$dureeDeVie,$famille){
11        $this->nom = $nom;
12        $this->type = $type;
13        $this->hauteur = $hauteur;
14        $this->dureeDeVie = $dureeDeVie;
15        $this->famille = $famille;
16    }
17
18    //Méthode afficher les plantes
19    public function afficher(){
20        echo "Nom : $this->nom <br>";
21        echo "Nom : $this->nom " . "<br>";
22        echo "Type : $this->type " . "<br>";
23        echo "Hauteur : $this->hauteur " . "<br>";
24        echo "Durée de vie : $this->dureeDeVie " . "<br>";
25        echo "Famille : $this->famille " . "<br>";
26        echo "*****<br>";
27    }
28
29 }
```

# Visibilité : public

- index.php

```
php index.php X
1  <?php
2  require_once "Plante.class.php";
3
4  $chene = new Plante("Le chêne","arbre","20 mètres","100 ans ", "Fagacées");
5  $rose = new Plante("La rose","fleur","1 mètre","2 ans","Rosacées");
6  $tournesol = new Plante("La tournesol","fleur","3 mètres","1 an","Astéracées");
7
8  $chene->afficher();
9  $rose->afficher();
10 $tournesol->afficher();
11
12 |
```

## Visibilité : public

- Pour apporter des modifications sur les attributs (public) d'une classe : On doit ajouter des setters

```
1 reference | 0 overrides  
public function setHauteur($hauteur){  
    $this->hauteur = $hauteur;  
}
```

```
index.php X  
PUBLIC > index.php > ...  
1  <?php  
2  require_once "Plante.class.php";  
3  
4  $chene = new Plante("Le chêne","arbre","20 mètres","100 ans ", "Fagacées");  
5  $rose = new Plante("La rose","fleur","1 mètre","2 ans","Rosacées");  
6  $tournesol = new Plante("La tournesol","fleur","3 mètres","1 an","Astéracées");  
7  
8  $chene->afficher();  
9  $rose->afficher();  
10 $tournesol->afficher();  
11  
12 //Apporter une modification  
13 $chene->setHauteur("25 mètres");  
14 $chene->afficher();  
15
```

## Visibilité : `protected`

- Les attributs déclarés avec le mot-clé **`protected`** sont accessibles uniquement depuis l'intérieur de la classe elle-même et des classes qui en héritent (les classes enfants).



# Visibilité : protected



```
Plante.class.php X
PROTECTED > Plante.class.php > PHP Intelephense > Plante
1  <?php
   6 references | 1 implementation
2  class Plante{
   7 references
3      protected $nom;
   4 references
4      protected $type;
   6 references
5      protected $hauteur;
   4 references
6      protected $dureeDeVie;
   4 references
7      protected $famille;
8
   5 references | 1 override
9      public function __construct($nom,$type,$hauteur,$dureeDeVie,$famille){
10         $this->nom = $nom;
11         $this->type = $type;
12         $this->hauteur = $hauteur;
13         $this->dureeDeVie = $dureeDeVie;
14         $this->famille = $famille;
15     }
16
17     //Méthode afficher les plantes
   5 references | 0 overrides
18     protected function afficher(){
19         echo "Nom : $this->nom <br>";
20         echo "Nom : $this->nom " . "<br>";
21         echo "Type : $this->type " . "<br>";
22         echo "Hauteur : $this->hauteur " . "<br>";
23         echo "Durée de vie : $this->dureeDeVie " . "<br>";
24         echo "Famille : $this->famille " . "<br>";
25         echo "*****<br>";
26     }
27
   1 reference | 0 overrides
28     public function setHauteur($hauteur){
29         $this->hauteur = $hauteur;
30     }
31
32 }
```

# Visibilité : protected

```
Plante.class.php  Chene.class.php X
PROTECTED > Chene.class.php > PHP Intelephense > Chene
1  <?php
2
3  require_once "Plante.class.php";
   1 reference | 0 implementations
4  class Chene extends Plante
5  {
6
   4 references
7      private $couleur;
8
   2 references | 0 overrides | prototype
9      public function __construct($nom, $type, $hauteur, $dureeDeVie, $famille, $couleur)
10     {
11         parent::__construct($nom, $type, $hauteur, $dureeDeVie, $famille);
12         $this->couleur = $couleur;
13     }
   0 references | 0 overrides
14     public function getCouleur()
15     {
16         return $this->couleur;
17     }
   0 references | 0 overrides
18     public function setCouleur($couleur)
19     {
20         $this->couleur = $couleur;
21     }
22
23
   1 reference | 0 overrides
24     public function afficherChene()
25     {
26         $this->afficher();
27         echo "Couleur : $this->couleur<br>";
28     }
29
30
```

```
Plante.class.php  Chene.class.php  index.php X
PROTECTED > index.php > ...
1  <?php
2  require_once "Chene.class.php";
3
4  $chene = new Chene("Le chêne", "arbre", "20 mètres", "100 ans ", "Fagacées", "brun");
5
6  $chene->afficherChene();
7
8
```

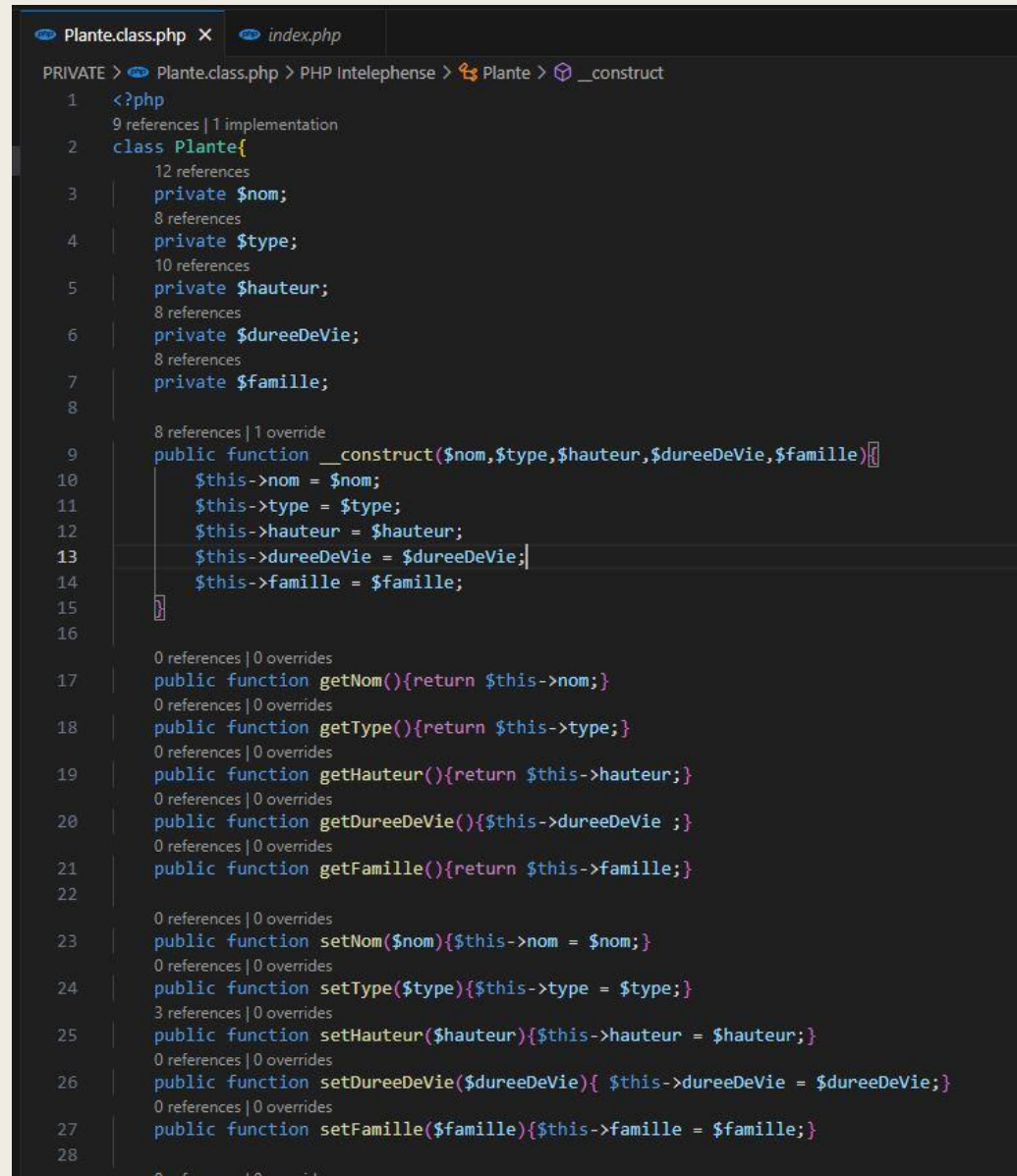
## Visibilité : protected

```
Plante.class.php  Chene.class.php  index.php X
PROTECTED > index.php > ...
2  require_once "Chene.class.php";
3
4  $chene = new Chene("Le chêne","arbre","20 mètres","100 ans ", "Fagacées","brun");
5
6  echo "Avant modification de la hauteur : <br>";
7  $chene->afficherChene();
8
9  //Modification de la hauteur
10 $chene->setHauteur("25 mètres");
11
12 echo "Après modification de la hauteur : <br>";
13 $chene->afficherChene();
14
15 |
```

## Visibilité : private

- Les attributs déclarés avec le mot-clé **private** sont accessibles uniquement depuis l'intérieur de la classe où ils sont définis. Ils ne sont pas accessibles dans les classes enfants ou de l'extérieur.

# Visibilité : private



The screenshot shows a code editor with two tabs: 'Plante.class.php' and 'index.php'. The 'Plante.class.php' tab is active, displaying the following PHP code:

```
1 <?php
2 class Plante{
3     private $nom;
4     private $type;
5     private $hauteur;
6     private $dureeDeVie;
7     private $famille;
8
9     public function __construct($nom,$type,$hauteur,$dureeDeVie,$famille){
10         $this->nom = $nom;
11         $this->type = $type;
12         $this->hauteur = $hauteur;
13         $this->dureeDeVie = $dureeDeVie;
14         $this->famille = $famille;
15     }
16
17     public function getNom(){return $this->nom;}
18     public function getType(){return $this->type;}
19     public function getHauteur(){return $this->hauteur;}
20     public function getDureeDeVie(){return $this->dureeDeVie;}
21     public function getFamille(){return $this->famille;}
22
23     public function setNom($nom){$this->nom = $nom;}
24     public function setType($type){$this->type = $type;}
25     public function setHauteur($hauteur){$this->hauteur = $hauteur;}
26     public function setDureeDeVie($dureeDeVie){$this->dureeDeVie = $dureeDeVie;}
27     public function setFamille($famille){$this->famille = $famille;}
28 }
```

The code defines a class 'Plante' with five private attributes: \$nom, \$type, \$hauteur, \$dureeDeVie, and \$famille. It includes a constructor method \_\_construct() and five getter methods (getNom(), getType(), getHauteur(), getDureeDeVie(), getFamille()) and five setter methods (setNom(), setType(), setHauteur(), setDureeDeVie(), setFamille()).

## Visibilité : private



```
PRIVAT > index.php > ...
1  <?php
2  require_once "Plante.class.php";
3
4  $chene = new Plante("Le chêne","arbre","20 mètres","100 ans ", "Fagacées");
5  $rose = new Plante("La rose","fleur","1 mètre","2 ans","Rosacées");
6  $tournesol = new Plante("La tournesol","fleur","3 mètres","1 an","Astéracées");
7
8  $chene->afficher();
9  $rose->afficher();
10 $tournesol->afficher();
11
12 //Apporter une modification
13 $chene->setHauteur("25 mètres");
14 $chene->afficher();
15
16
```