

# Créer ses propres filtres et fonctions

Pour créer notre propre filtre ou notre propre fonction, nous allons créer dans le dossier "**src**" un dossier "**Twig**" contenant un fichier que l'on nommera comme on le souhaite, ici je l'appellerais "**AppExtension.php**".

Celui ci devra avoir un namespace correspondant à son emplacement, ici "**App\Twig**" Ainsi qu'une classe portant le même nom que le fichier, ici "**AppExtension**", Cette classe devra hériter de "**Twig\Extension\AbstractExtension**".

```
namespace App\Twig;

use Twig\Extension\AbstractExtension;

class AppExtension extends AbstractExtension
{}
```

Ensuite on devra créer une méthode "**getFilters**" si on veut créer de nouveaux filtres et "**getFunctions**" si on veut créer de nouvelles fonctions.

```
// en haut du fichier :
use Twig\TwigFilter;
// Dans la classe :
# Cette fonction va retourner les filtres nouvellement créé.
public function getFilters()
{
    return [
        /*
            new TwigFilter prend en premier paramètre le nom du filtre
            en second un tableau contenant la classe où trouver le filtre, et
            le nom de la fonction contenant le filtre. (Fonction que l'on déclare
plus bas)
            En troisième un tableau d'option.
        */
        new TwigFilter('price', [$this, 'formatPrice'], [
            // Cette option permet de ne pas filtrer le html
            'is_safe' => [
                'html'
            ]
        ]),
    ];
}

public function getFunctions()
{
    return [
        // fonctionne de même que pour les filtres.
        new TwigFunction('area', [$this, 'calculateArea']),
    ];
}
```

```
    ];  
}
```

Chaque filtre ou fonction correspondra à une nouvelle méthode créée dans ce fichier, pour plus de détail, rendez vous dans ce fichier.

```
// Le premier paramètre correspondra à ce qui se trouve à gauche du filtre.  
public function formatPrice(  
    float $number,  
    string $sign = "€",  
    bool $before = true,  
    string $decPoint = ',',  
    string $thousandsSep = ' ',  
    int $decimals = 0,  
): string  
{  
    // Formate un nombre.  
    $price = number_format($number, $decimals, $decPoint, $thousandsSep);  
  
    if($before) $price = "<sup>$sign</sup>".$price;  
    else $price .= "<sup>$sign</sup>";  
  
    return $price;  
}  
  
public function calculateArea(int $width, int $length): int  
{  
    return $width * $length;  
}
```