

Afficher par catégorie

Modification de [index.php](#)

Nous commençons par modifier [index.php](#) :

```
<?php
$filename = __DIR__ . '/data/articles.json';
$articles = [];
$categories = [];

if (file_exists($filename)) {
    $articles = json_decode(file_get_contents($filename), true) ?? [];
    $cattmp = array_map(fn ($a) => $a['category'], $articles);
    $categories = array_reduce($cattmp, function ($acc, $cat) {
        if (isset($acc[$cat])) {
            $acc[$cat]++;
        } else {
            $acc[$cat] = 1;
        }
        return $acc;
    }, []);
    $articlePerCategories = array_reduce($articles, function ($acc,
$article) {
        if (isset($acc[$article['category']])) {
            $acc[$article['category']] = [...$acc[$article['category']],
$article];
        } else {
            $acc[$article['category']] = [$article];
        }
        return $acc;
    }, []);
}

?>

<!DOCTYPE html>
<html lang="fr">
    <head>
        <?php require_once 'includes/head.php' ?>
```

```

    <link rel="stylesheet" href="/public/css/index.css">
    <title>Blog</title>
</head>

<body>
    <div class="container">
        <?php require_once 'includes/header.php' ?>
        <div class="content">
            <div class="category-container">
                <?php foreach ($categories as $cat => $num) : ?>
                    <h2><? $cat ?></h2>
                    <div class="articles-container">
                        <?php foreach ($articlePerCategories[$cat] as
$a) : ?>

                            <div class="article block">
                                <div class="overflow">
                                    <div class="img-container"
style="background-image:url(<? $a['image'] ?>)"></div>
                                </div>
                                <h3><? $a['title'] ?></h3>
                            </div>
                        <?php endforeach; ?>
                    </div>
                <?php endforeach; ?>
            </div>
        </div>
        <?php require_once 'includes/footer.php' ?>
    </div>

</body>
</html>

```

`$cattmp = array_map(fn ($a) => $a['category'], $articles);` permet de retourner un tableau contenant uniquement les catégories.

```

$categories = array_reduce($cattmp, function ($acc, $cat) {
    if (isset($acc[$cat])) {
        $acc[$cat]++;
    } else {
        $acc[$cat] = 1;
    }
}, []);

```

```

    }
    return $acc;
}, []);

```

Nous parcourons le tableau des catégories et partons d'un tableau vide [].

Pour chaque élément `$cat` du tableau des catégories. Nous vérifions s'il y a déjà une clé pour la catégorie dans le tableau `$acc`. S'il y en a une nous incrémentons la valeur. S'il n'y en a pas nous créons un élément avec comme clé la catégorie et comme valeur 1.

Nous obtenons un tableau associatif de la forme `['category' => nombre]`.

```

$articlePerCategories = array_reduce($articles, function ($acc, $article)
{
    if (isset($acc[$article['category']])) {
        $acc[$article['category']] = [...$acc[$article['category']],
$article];
    } else {
        $acc[$article['category']] = [$article];
    }
    return $acc;
}, []);

```

Nous parcourons le tableau des articles et partons d'un tableau vide [].

Pour chaque élément `$article` du tableau des articles. Nous vérifions s'il y a déjà une clé pour la catégorie dans le tableau `$acc`. S'il y en a une, nous ajoutons l'article à la fin du tableau des articles. S'il n'y en a pas nous créons un élément avec comme clé la catégorie de l'article et comme valeur un nouveau tableau contenant l'article.

Nous obtenons un tableau associatif de la forme `['category' => [article1, article 2...]]`.

`$acc[$article['category']] = [...$acc[$article['category']], $article];` peut également être remplacé par la notation `array_push : $acc[$article['category']][] = $article;`.

Modification de `index.css`

Nous modifions `index.css` pour utiliser `h3` et plus `h2` :

```
.articles-container {  
  display: flex;  
  flex-flow: row wrap;  
  justify-content: center;  
  align-items: flex-start;  
}
```

```
.article {  
  flex: 30%;  
  margin: 1rem;  
  min-height: 41rem;  
  min-width: 30rem;  
  cursor: pointer;  
}
```

```
.article:hover .img-container {  
  transform: scale(1.1);  
}
```

```
.category-container h2 {  
  font-size: 1.9rem;  
  padding: 0 1rem;  
}
```

```
.article h3 {  
  font-size: 1.7rem;  
  padding: 1rem 1.5rem;  
}
```

```
.img-container {  
  height: 30rem;  
  background-size: cover;  
  transition: transform 1s;  
}
```