



10) Require et Include

Dans ce chapitre nous allons voir quelque chose de simple, c'est le **require** et le **include**. Vous avez déjà vu un truc du style en python et en javascript. Jusqu'à maintenant on a travaillé toujours dans un même fichier, mais il faut savoir qu'on peut lier plusieurs fichiers différents. On peut par exemple créer un fichier où l'on écrira des fonctions, et il suffira d'appeler ce fichier dans un autre pour qu'on puisse utiliser les fonctions présentes. En gros **require** et **include** vont vous permettre d'inclure un fichier php dans le fichier php courant. On peut le retrouver dans la documentation officielle de php (tapez require).

On va créer un nouveau fichier php que l'on va appeler « **require_include.php** » et un deuxième fichier que l'on va appeler « **fonctionsReq.php** ».

Dans le fichier **fonctionsReq.php** on va y mettre toutes nos fonctions dont on aura besoin et dans le fichier **require_include.php** on fera l'appelle de ces fonctions.

Dans **fonctionsReq.php** copiez-collez vos fonctions : addition, soustraction, division, multiplication, moyenne2nombre, estFemme et estAdulte.

Et créer celle-là :

```
function ouiOuNon(string $phrase = null) : bool{
    while(true){
        $reponse = readline($phrase . " - (o)ui/(n)on : " );
        if($reponse === "o"){
            return true;
        }elseif($reponse === "n"){
            return false;
        }
    }
}
```

Cette fonction retourne un booléen en fonction de si on tape « o » ou « n » pour oui ou non.

Maintenant que nous avons définis plusieurs fonctions dans le fichier **fonctionsReq.php**, on peut aller dans le fichier **require_include.php** et faire l'appel :

La syntaxe c'est de mettre le mot clé :

- **include 'cheminVersLeFichierAInclure' ;**

Dans notre cas les deux fichiers sont au même niveau, point de vue arborescence, donc pour le chemin on mettra juste le nom du fichier :

```
<?php
    include "fonctionsReq.php";
    echo addition(5,6);
?>
```

Et si maintenant vous exécutez le fichier **require_include.php**, il fera bien l'addition et affichera bien comme résultat 11.

J'aurai pu aussi combiner plusieurs fonctions du fichier :

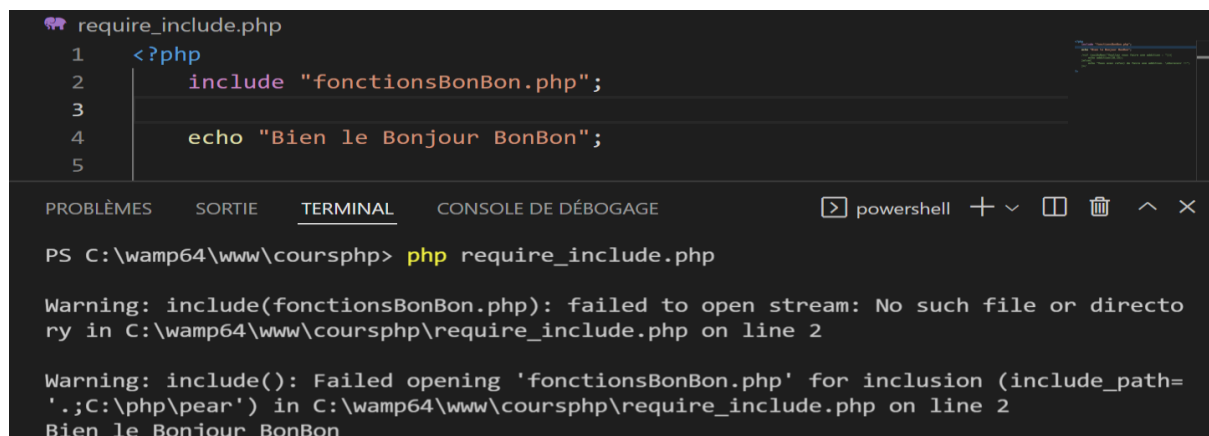
```
if (ouiOuNon("Voulez vous faire une addition : ")){
    echo addition(10,15);
}else{
    echo "Vous avez refusé de faire une addition. \nAurevoir !!";
}
```

Ici j'appelle d'abord la fonction pour vérifier si oui ou non on veut faire une addition, en fonction de la réponse on fait l'addition ou on renvoi un message.

Vous verrez que c'est pratique pour séparer vos codes proprement. Vous pourrez aussi inclure plusieurs fichiers.

A savoir que notre mot clé **include** peut être remplacé par le **require**, rappelez-vous dans la documentation il y était mis la différence principale entre les deux.

Le **require** demande que le fichier existe. Par exemple si j'appelle un fichier non existant avec le **include**, il va quand même exécuter ce qu'il y a dans mon fichier, alors qu'avec le **require** il exécutera pas.



```
require_include.php
1  <?php
2      include "fonctionsBonBon.php";
3
4      echo "Bien le Bonjour BonBon";
5

PROBLÈMES  SORTIE  TERMINAL  CONSOLE DE DÉBOGAGE  powershell + - [] 🗑 ⤴ ✕

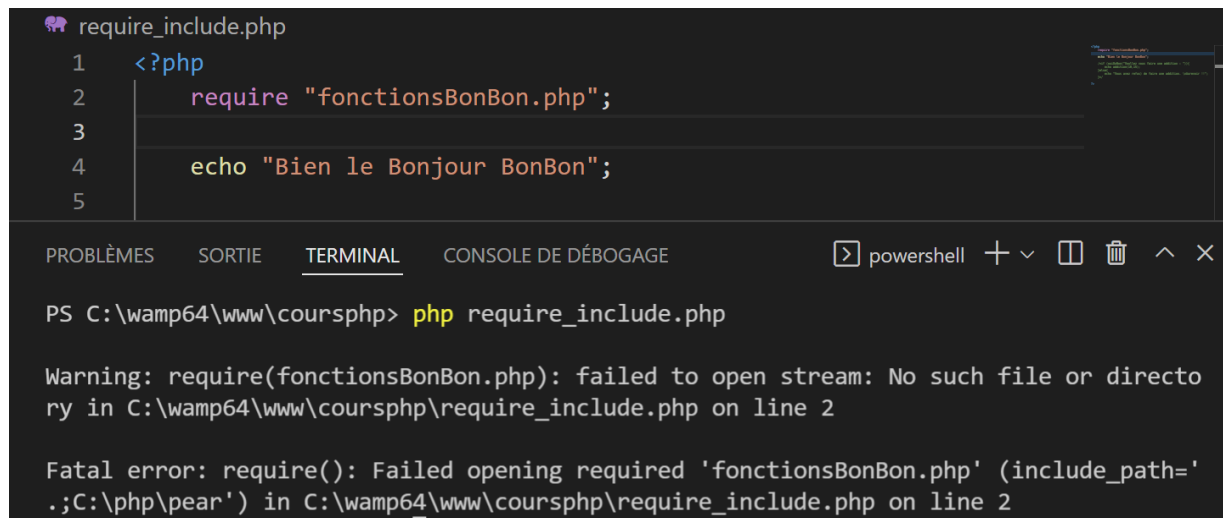
PS C:\wamp64\www\coursphp> php require_include.php

Warning: include(fonctionsBonBon.php): failed to open stream: No such file or directory in C:\wamp64\www\coursphp\require_include.php on line 2

Warning: include(): Failed opening 'fonctionsBonBon.php' for inclusion (include_path='.;C:\php\pear') in C:\wamp64\www\coursphp\require_include.php on line 2
Bien le Bonjour BonBon
```

Nous voyons bien qu'il nous a sorti des warning (attentions), ensuite il a quand même affiché le message.

Si je remplace le include par require il affichera ceci :



```
require_include.php
1  <?php
2      require "fonctionsBonBon.php";
3
4      echo "Bien le Bonjour BonBon";
5
PROBLÈMES  SORTIE  TERMINAL  CONSOLE DE DÉBOGAGE  powershell + - □ □ ⌵ ×
PS C:\wamp64\www\coursphp> php require_include.php

Warning: require(fonctionsBonBon.php): failed to open stream: No such file or directory in C:\wamp64\www\coursphp\require_include.php on line 2

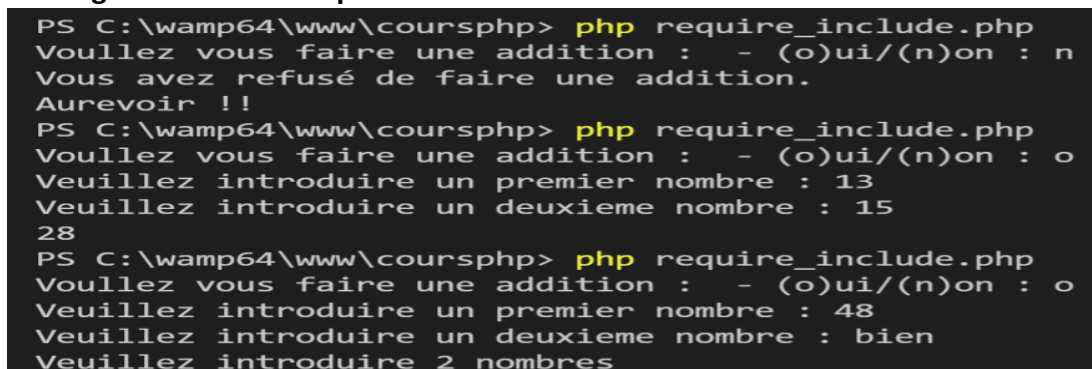
Fatal error: require(): Failed opening required 'fonctionsBonBon.php' (include_path='.;C:\php\pear') in C:\wamp64\www\coursphp\require_include.php on line 2
```

Vous voyez bien qu'ici il n'affichera pas le message et il dira qu'il y a une fatal error (erreur fatale).

Le **require** va donc complètement planter votre script si le fichier n'existe pas ou que vous vous êtes trompé de chemin alors que le **include** lui va juste dire qu'il y a un problème mais le reste du script va pouvoir continuer à s'exécuter. Vous comprendrez que nous utiliserons principalement des **require** pour être sûr que nous appelons bien les fichiers.

(Exos)

- 1) Créez un script qui demande si on veut faire une soustraction et qui le fait si on dit oui et qui renvoi un message si on dit non.
- 2) Créez un script qui demande si on veut faire une multiplication et qui le fait si on dit oui et qui renvoi un message si on dit non.
- 3) Créez un script qui demande si on veut faire une division et qui le fait si on dit oui et qui renvoi un message si on dit non. Pas de boucle.
- 4) Améliorez moi l'exemple avec l'addition. On nous demande bien si on veut faire une addition, mais cette fois ci si on dit oui, on nous demandera d'introduire au clavier nos 2 nombres. Si on introduit autre chose que des nombres, il me renvoi un message : « Vous n'avez pas introduit 2 nombres ». Pas de boucle.



```
PS C:\wamp64\www\coursphp> php require_include.php
Voulez vous faire une addition : - (o)ui/(n)on : n
Vous avez refusé de faire une addition.
Aurevoir !!
PS C:\wamp64\www\coursphp> php require_include.php
Voulez vous faire une addition : - (o)ui/(n)on : o
Veuillez introduire un premier nombre : 13
Veuillez introduire un deuxieme nombre : 15
28
PS C:\wamp64\www\coursphp> php require_include.php
Voulez vous faire une addition : - (o)ui/(n)on : o
Veuillez introduire un premier nombre : 48
Veuillez introduire un deuxieme nombre : bien
Veuillez introduire 2 nombres
```

- 5) Améliorez la soustraction et la multiplication comme pour l'addition.
- 6) (Difficile) Améliorez la division comme pour l'addition. **Attention** la division, on ne peut pas diviser par zéro. Si on introduit 0 comme deuxième nombre, il me redemande d'introduire un nombre. Et si j'introduis autre chose qu'un nombre quand on me demande d'introduire un deuxième nombre après une tentative de division par 0, je veux qu'il m'envoie un message : « Vous avez introduit autre chose qu'un nombre »

```
PS C:\wamp64\www\coursphp> php require_include.php
Veuillez vous faire une division : - (o)ui/(n)on : n
Vous avez refusé de faire une division.
Aurevoir !!
PS C:\wamp64\www\coursphp> php require_include.php
Veuillez vous faire une division : - (o)ui/(n)on : o
Veuillez introduire un premier nombre : 45
Veuillez introduire un deuxième nombre : 0
Vous avez tenter une division par 0, recommencer
Veuillez introduire un deuxième nombre : pok
Vous avez introduit autre chose qu'un nombre
Veuillez introduire un deuxième nombre : 3
15
```

- 7) (Très difficile pour les plus rapides) **Avant de commencer, je vais vous montrer comment il fonctionne.** Créez une fonction « calcule » dans votre fichier fonctionsReq.php qui permet de choisir si on veut faire une addition, une soustraction, une multiplication ou une division. La fonction « calcule », recevra 3 paramètres (nombre1, nombre2, fct). Ici « fct » représentera un chiffre entre 1 et 4. En fonction de ce chiffre pour 1 il fait l'addition, 2 c'est soustraction, 3 c'est multiplication et 4 c'est division. Elle retournera l'appelle de la fonction demandé. La première chose que votre programme demandera c'est si on veut faire un calcule, si oui on continue sinon on envoie un message. La deuxième chose que votre programme demandera c'est de choisir un chiffre entre 1 et 4. Si on met autre chose qu'un nombre ou un chiffre n'étant pas entre 1 et 4, il enverra comme message « Vous avez introduit autre chose qu'un chiffre entre 1 et 4 ».

```
PS C:\wamp64\www\coursphp> php require_include.php
Veuillez vous faire un calcule : - (o)ui/(n)on : n
Vous avez refusé de faire un calcule.
Aurevoir !!
PS C:\wamp64\www\coursphp> php require_include.php
Veuillez vous faire un calcule : - (o)ui/(n)on : o
Veuillez introduire un chiffre entre 1 et 4 (1 = Addition, 2 = Soustraction, 3 = Multiplication, 4 = Divisio
n) : p
Vous avez introduit autre chose qu'un chiffre entre 1 et 4
PS C:\wamp64\www\coursphp> php require_include.php
Veuillez vous faire un calcule : - (o)ui/(n)on : o
Veuillez introduire un chiffre entre 1 et 4 (1 = Addition, 2 = Soustraction, 3 = Multiplication, 4 = Divisio
n) : 45
Vous avez introduit autre chose qu'un chiffre entre 1 et 4
```

Il vous demandera ensuite d'introduire 2 nombres avec lesquelles il fera le calcul que vous aurez demandé. Mais si on introduit autre chose qu'un nombre il enverra comme message « veuillez introduire 2 nombres ».

```
PS C:\wamp64\www\coursphp> php require_include.php
Veuillez vous faire un calcul : - (o)ui/(n)on : o
Veuillez introduire un chiffre entre 1 et 4 (1 = Addition, 2 = Soustraction, 3 = Multiplication, 4 = Division) : 3
Veuillez introduire un premier nombre : 45
Veuillez introduire un deuxième nombre : h
Vous devez introduire 2 nombres
```

Je veux que vous gériez si on met division par zéro, je veux qu'il redemande d'introduire un nombre. Et si j'introduis autre chose comme deuxième nombre, il m'envoie un message « vous avez introduit autre chose qu'un nombre ».

```
Veuillez vous faire un calcul : - (o)ui/(n)on : o
Veuillez introduire un chiffre entre 1 et 4 (1 = Addition, 2 = Soustraction, 3 = Multiplication, 4 = Division) : 1
Veuillez introduire un premier nombre : 12
Veuillez introduire un deuxième nombre : 14
26
PS C:\wamp64\www\coursphp> php require_include.php
Veuillez vous faire un calcul : - (o)ui/(n)on : o
Veuillez introduire un chiffre entre 1 et 4 (1 = Addition, 2 = Soustraction, 3 = Multiplication, 4 = Division) : 4
Veuillez introduire un premier nombre : 56
Veuillez introduire un deuxième nombre : 0
Vous avez tenté une division par 0, recommencer
Veuillez introduire un deuxième nombre : m
Vous avez introduit autre chose qu'un nombre
Veuillez introduire un deuxième nombre : 2
28
```

Pour cet exercice basez-vous sur les précédents 🤖

Require_once et include_once

Les deux fonctionnent un peu comme le **require** et **include**, sauf que comme son nom l'indique (once en anglais veut dire une fois), il n'inclut qu'une seule fois le fichier. Dans la documentation officielle vous pouvez le retrouver.

Si je veux créer **sep.php** un fichier qui me permet de séparer juste avec des « ===== », et ensuite l'appeler plusieurs fois, là j'utiliserai alors avec **include** ou **require**. Par contre si c'est un fichier de fonctions, on n'aura pas besoin de l'appeler à chaque fois, un seul appel est suffisant, donc on utilisera **require_once** ou **include_once**.

```
PS C:\wamp64\www\coursphp> php require_include.php
=====
ok
=====
blabla
=====
```

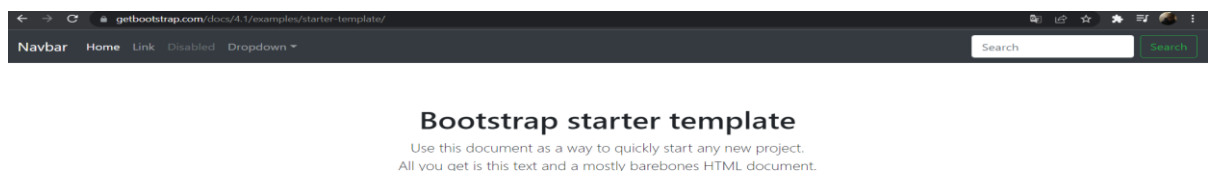
```
echo "\n ok \n";
require "sep.php";
echo "\n blabla \n";
require "sep.php";
```

11) PHP et HTML

Comme promis, je vous avais dit qu'on allait un moment donner faire le lien entre nos codes PHP et les pages web. Tous ce qu'on a fait jusqu'à présent c'était bien pour comprendre le langage PHP et un peu d'algorithmie (logique informatique). A la base vous êtes formé pour devenir développeur web, donc savoir faire des pages web. Vous avez déjà eu plusieurs cours de frontend, ici vos connaissances seront plus qu'utile, vous pourrez les mettre en pratique. Vous avez déjà compris que PHP permet d'afficher quelques choses sur nos pages web. Comme on l'a dit au tout début du cours, par défaut ce qu'on met, sera affiché. Donc si j'exécute du code qui contient de l'HTML, automatiquement il va générer la même page en HTML.

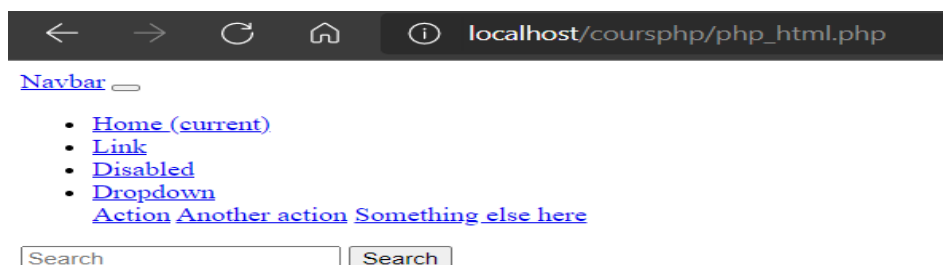
Créons un **fichier** php qu'on va appeler « **php_html.php** ». On va y mettre un templates. On va commencer par aller prendre un templates de twitter bootstrap. (version 4.3)

Allez tous sur ce lien : <https://getbootstrap.com/docs/4.3/examples/starter-template/>



Cliquez droit sur cette page ensuite cliquez sur afficher le code/page source. Ce code source vous allez le copier-coller dans votre fichier php_html.php.

Si vous sauvegardez et exécutez votre fichier via le navigateur, vous devez avoir un résultat comme ceci grâce au serveur web de wamp (apache) :



Bootstrap starter template

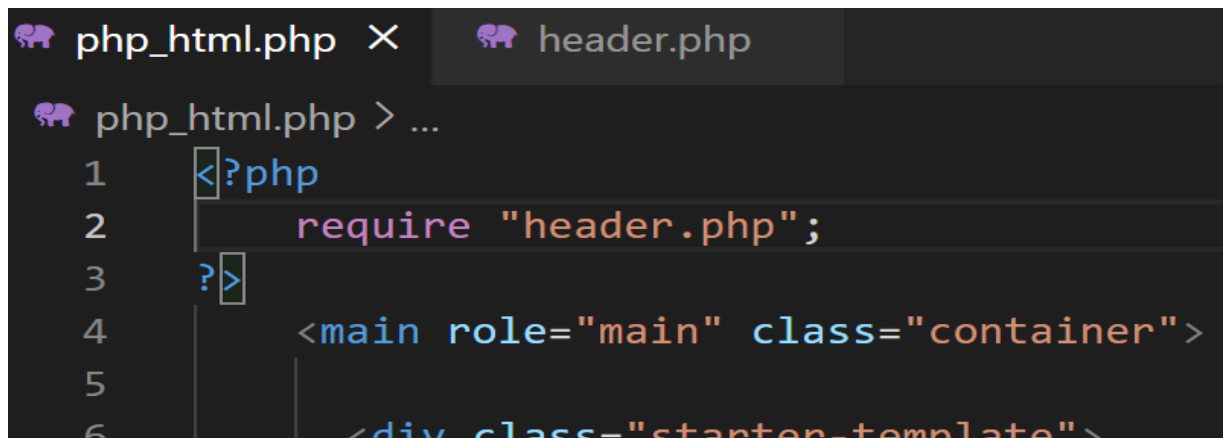
Use this document as a way to quickly start any new project.
All you get is this text and a mostly barebones HTML document.

On n'a pas encore le même rendu mais ça va venir. Vous vous rappelez que si on lance ce fichier dans notre terminal, il va juste afficher ce qu'il y a dans ce fichier. Il faudrait du code php pour qu'il le traite.

On va un peu séparer notre code pour y voir plus clair. On sait que généralement une barre de navigation ou un entête, est toujours repris dans plusieurs pages d'un même site internet.

On va séparer toute la partie du haut jusqu'à la fermeture de `</nav>`. Mettre tout ça dans un fichier que l'on va appeler « **header.php** ».

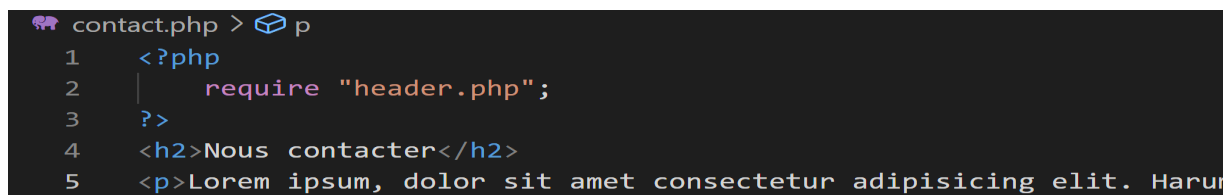
Ensuite il faudra retourner dans votre fichier « **php_html.php** » et ouvrir une balise php et y introduire un **require** suivi du chemin du fichier **header.php**.



```
php_html.php X header.php
php_html.php > ...
1  <?php
2      require "header.php";
3  ?>
4  <main role="main" class="container">
5
6  <div class="starter-template">
```

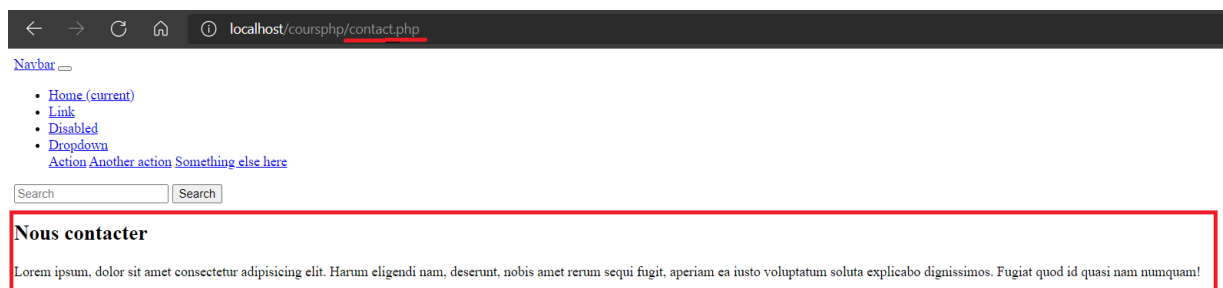
Rechargez votre page web, vous verrez que rien n'aura changé pourtant on a utilisé cette fois ci deux fichiers en utilisant du php, grâce au « **require** ».

Si maintenant je veux créer un troisième fichier que je vais appeler « **contact.php** » qui servira à nous contacter, je peux y mettre quelques choses comme cela :



```
contact.php > p
1  <?php
2      require "header.php";
3  ?>
4  <h2>Nous contacter</h2>
5  <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Harum
```

J'ai mis un titre et un paragraphe lorem ipsum. Ensuite allez dans le navigateur et lancez notre fichier **contact.php** :



On voit bien qu'on a gardé notre barre de navigation, c'est juste le contenu de la page qui a changé.

Ce que j'ai fait pour le header, je peux le faire aussi pour le footer bien entendu. On va prendre toute la partie après la fin du `</main>`, et on va créer un fichier « **footer.php** » qui recevra tous ce code.

```
🐘 footer.php > ...
1      <!-- Bootstrap core JavaScript
2      ===== -->
3      <!-- Placed at the end of the document so the pages load faster
4      <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" :
5      <script>window.jQuery || document.write('<script src="../../asse
6      <script src="../../assets/js/vendor/popper.min.js"></script>
7      <script src="../../dist/js/bootstrap.min.js"></script>
8      </body>
9      </html>
```

Et dans mes fichiers **php_html.php** et **contact.php**, on mettra un **require** à la fin, pour faire appel à ce bout de code :

```
🐘 php_html.php > ...
1      <?php
2      require "header.php";
3      ?>
4      <main role="main" class="container">
5      <div class="starter-template">
6      <h1>Bootstrap starter template</h1>
7      <p class="lead">Use this document as a way to quickly start
8      </div>
9      </main><!-- /.container -->
10
11 <?php
12 require "footer.php";
13 ?>
```

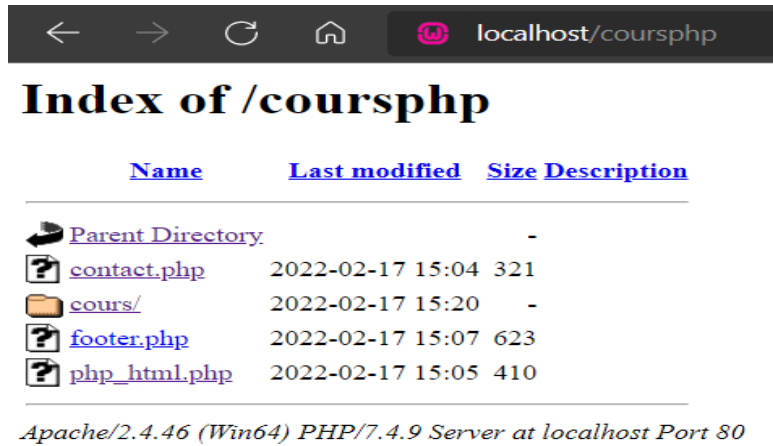
Vous comprenez ici l'utilité de faire cela, ça nous permettra de juste appeler toujours une même navbar ou un même footer etc. Et en plus ça rend le code plus lisible.

On va un peu restructurer notre dossier de travail dans `www/coursphp`. Tous les fichiers d'exercices ou de cours que vous avez, vous aller les mettre dans un dossier « cours » que vous créerez. Dans « coursphp » il ne devra rester que ces 4 fichiers ci :

« Windows-SSD (C:) > wamp64 > www > coursphp > »				Rechercher dans :
Nom	Modifié le	Type	Taille	
📁 cours	17-02-22 15:20	Dossier de fichiers		
📄 contact.php	17-02-22 15:04	Fichier source PHP	1 Ko	
📄 footer.php	17-02-22 15:07	Fichier source PHP	1 Ko	
📄 header.php	17-02-22 14:42	Fichier source PHP	3 Ko	
📄 php_html.php	17-02-22 15:05	Fichier source PHP	1 Ko	

Maintenant si vous allez dans votre navigateur et que vous taper : <http://localhost/coursphp>

Vous atterrissez sur quelque chose comme cela :



The screenshot shows a web browser window with the address bar displaying 'localhost/coursphp'. The main content area shows a directory listing titled 'Index of /coursphp'. The listing has columns for Name, Last modified, Size, and Description. It lists a Parent Directory, contact.php, cours/, footer.php, and php_html.php. At the bottom, it says 'Apache/2.4.46 (Win64) PHP/7.4.9 Server at localhost Port 80'.

Name	Last modified	Size	Description
Parent Directory		-	
contact.php	2022-02-17 15:04	321	
cours/	2022-02-17 15:20	-	
footer.php	2022-02-17 15:07	623	
php_html.php	2022-02-17 15:05	410	

Apache/2.4.46 (Win64) PHP/7.4.9 Server at localhost Port 80

Si vous cliquez sur cours, vous aurez tous vos fichiers sur lesquels nous avons travaillé jusqu'ici.

Ici il nous reste qu'une seule chose à faire pour qu'il exécute notre page web. Il faut renommer notre fichier php_html.php en **index.php**.

Maintenant vous aurez directement votre page d'accueil :



The screenshot shows a web browser window with the address bar displaying 'localhost/coursphp/'. The main content area shows a Bootstrap starter template. It has a Navbar with links for Home (current), Link, Disabled, and Dropdown. Below the navbar is a search bar with the text 'Search' and a 'Search' button. The main heading is 'Bootstrap starter template'. Below the heading is a paragraph: 'Use this document as a way to quickly start any new project. All you get is this text and a mostly barebones HTML document.'

Navbar

- [Home \(current\)](#)
- [Link](#)
- [Disabled](#)
- [Dropdown](#)

[Action](#) [Another action](#) [Something else here](#)

Search Search

Bootstrap starter template

Use this document as a way to quickly start any new project.
All you get is this text and a mostly barebones HTML document.

Nous contacter

Lorem ipsum, dolor sit amet consectetur adipisicing elit. Harum eliget

Pour l'instant nous avons un petit problème de mise en page, il vient du fait que dans la partie bootstrap CSS on essaye de charger un fichier que nous n'avons pas.

On peut retourner sur ce lien : <https://getbootstrap.com/docs/4.3/getting-started/introduction/>

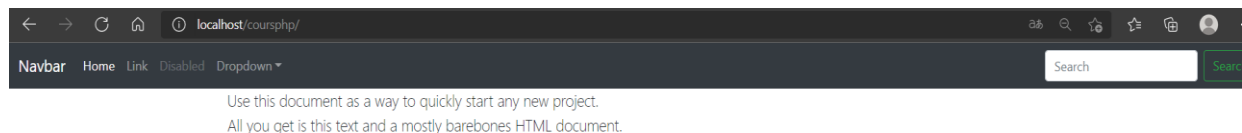
On aura besoin de copier le lien du CSS. Et on ira remplacer dans notre fichier header.php toute la partie qu'on a sur le bootstrap CSS.

```
10
11     <title>Starter Template for Bootstrap</title>
12
13     <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com
14
15 </head>
```

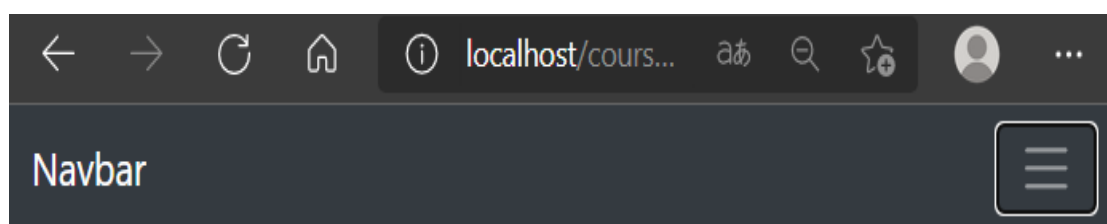
Et toujours sur le même site on a aussi la partie du JS à copier, qu'on mettra dans notre fichier footer.php.

```
footer.php > ...
1     <!-- Bootstrap core JavaScript
2     ===== -->
3     <!-- Placed at the end of the document so the pages load faster
4     <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
5     <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1
6     <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3
7     </body>
8 </html>
```

Normalement après ces deux phases, vous pouvez déjà voir que notre site affiche bien la barre de navigation :



Et vous remarquerez qu'il est « responsive » :



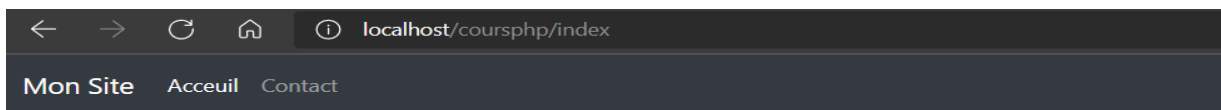
Use this document as a way to quickly start any new project.
All you get is this text and a mostly barebones HTML
document.

Dans le fichier header.php on va un peu le modifier pour qu'on ait ce résultat.

```
19 <nav class="navbar navbar-expand-md navbar-dark bg-dark mb-4">
20   <a class="navbar-brand" href="/coursphp/index">Mon Site</a>
21   <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarCollapse">
22     <span class="navbar-toggler-icon"></span>
23   </button>
24
25   <div class="collapse navbar-collapse" id="navbarCollapse">
26     <ul class="navbar-nav mr-auto">
27       <li class="nav-item active">
28         <a class="nav-link" href="/coursphp/index">Accueil <span class="sr-only">
29         </li>
30       <li class="nav-item">
31         <a class="nav-link" href="/coursphp/contact">Contact</a>
32       </li>

```

Ce qui donnera :



Bootstrap starter template

Use this document as a way to quickly start any new project.
All you get is this text and a mostly barebones HTML document.

On est clairement pour l'instant dans le côté frontend, mais ne vous inquiétez pas on va y retourner assez vite dans le backend.