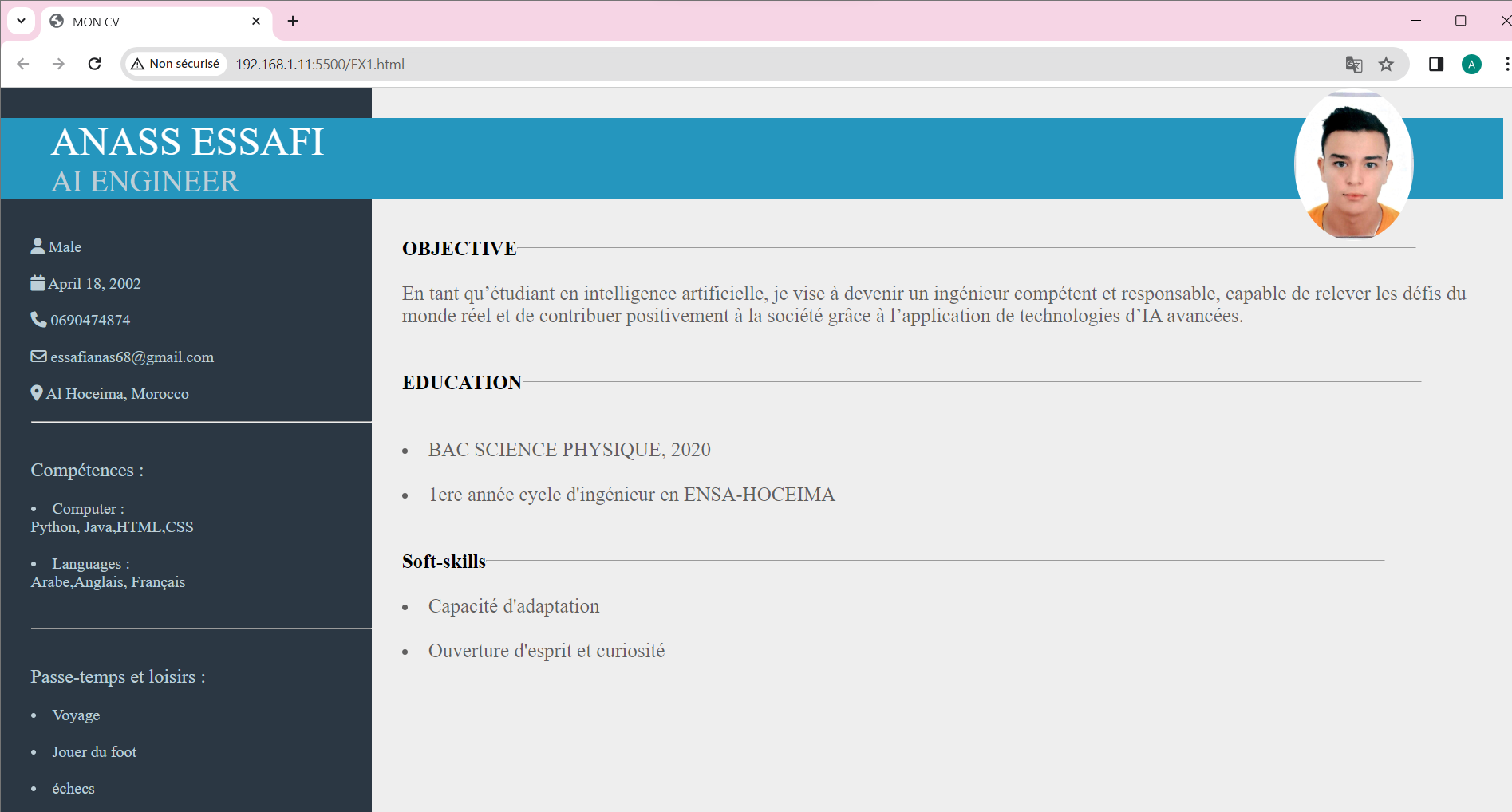
**TP2 :**

**EX1 : Réalisation de mon cv**



**EX2:**

**Écrivez un sélecteur pour chacun des énoncés suivants.**



**1-Tous les éléments <a> emboités dans un élément <p> :**

p a{

/\*selecteur\*/

}

1. **Tous les éléments <img> appartenant à la classe encadre et placés dans un élément <div> appartenant à la classe enonce.**

.enonce .encadre img{

/\*selecteur\*/

}

1. **Tous les éléments <img> dont l'attribut width vaut 300.**

Img[width"300"] {

/\*selecteur\*/

}

1. **Tous les éléments <time> dont l'attribut datetime a été défini et qui sont situés dans un élément <article> appartenant à la classe remarque.**

.article.remarque time[datetime]{

/\*selecteur\*/

}

1. **Tous les éléments <p> qui sont fils d'un élément <div> de la classe exemple.**

div.exemple p{

/\*selecteur\*/

}

1. **Tous les éléments <img> de classe encadre placés dans un élément <p> dont l'id est important.**

p#important img.encadre{

/\*selecteur\*/

}

1. **Le premier élément <img> dont le père est un élément <div>.**

div > img:first-child{

/\*selecteur\*/

}

1. **Tous les éléments <img> fils de rang impair d'un élément quelconque du document html.**

\*:nth-child(odd) > img {

/\* selecteur\*/

}

1. **Tous les éléments <a> dont la cible est un fichier d'extension .jpg situé dans le premier élément <p> qui est fils d'un élément <article> de la classe exercice.**

article.exercice > p:first-child a[href$=".jpg"] {

/\* selecteur\*/

}

**10-Les éléments <div> de la classe remarque qui sont les premiers frères (suivant) d'un élément <div> de classe reponse.**

div.reponse + div.remarque {

/\* selecteur\*/

}

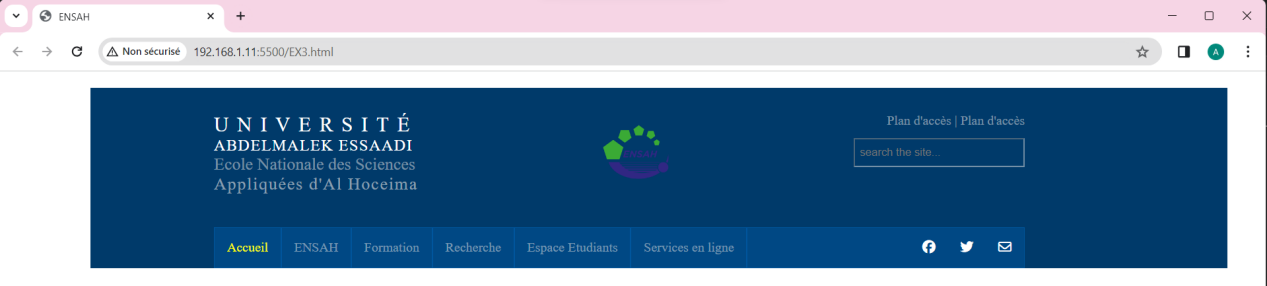
**11-Le second élément de type <p> fils d'un élément <article> placé dans un élément <section>**

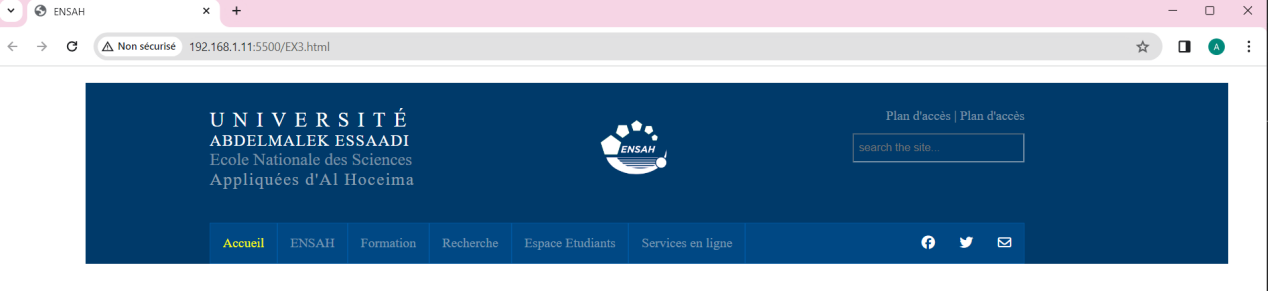
section article > p:nth-child(2){

/\* selecteur\*/

}

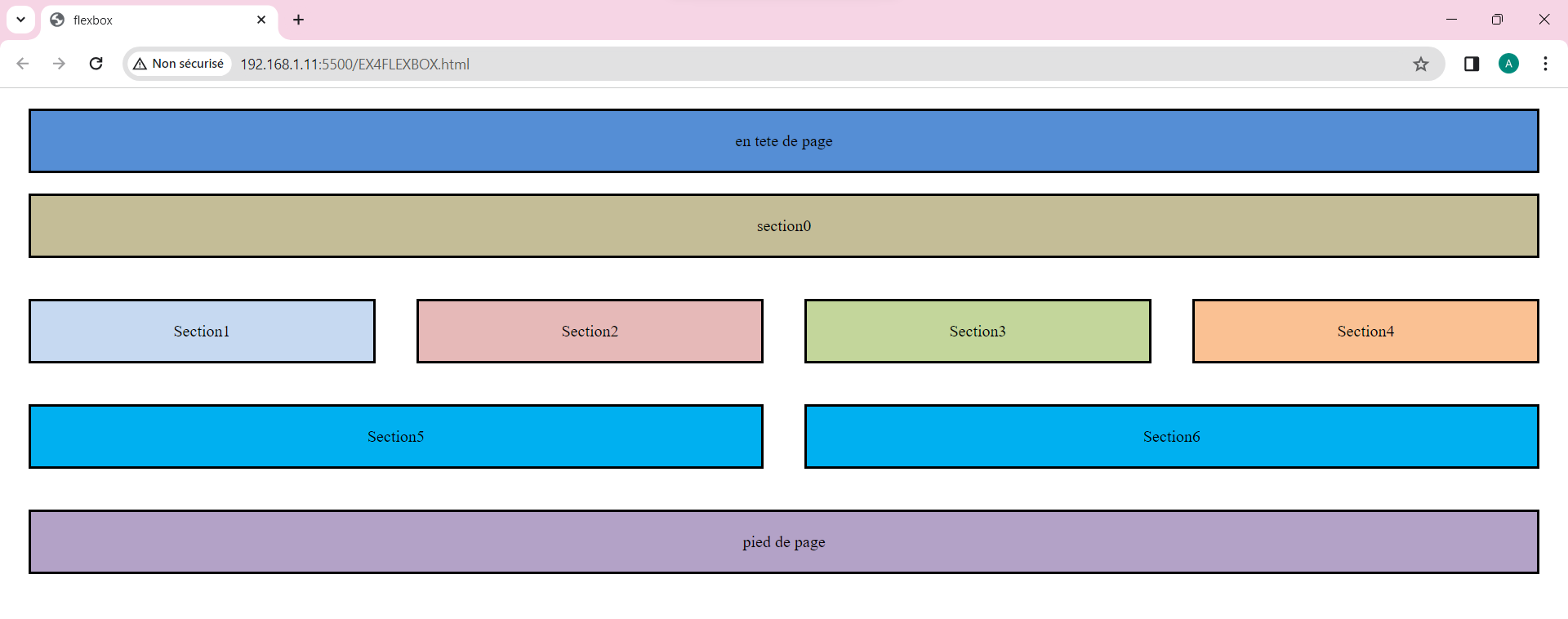
**EX3 :**

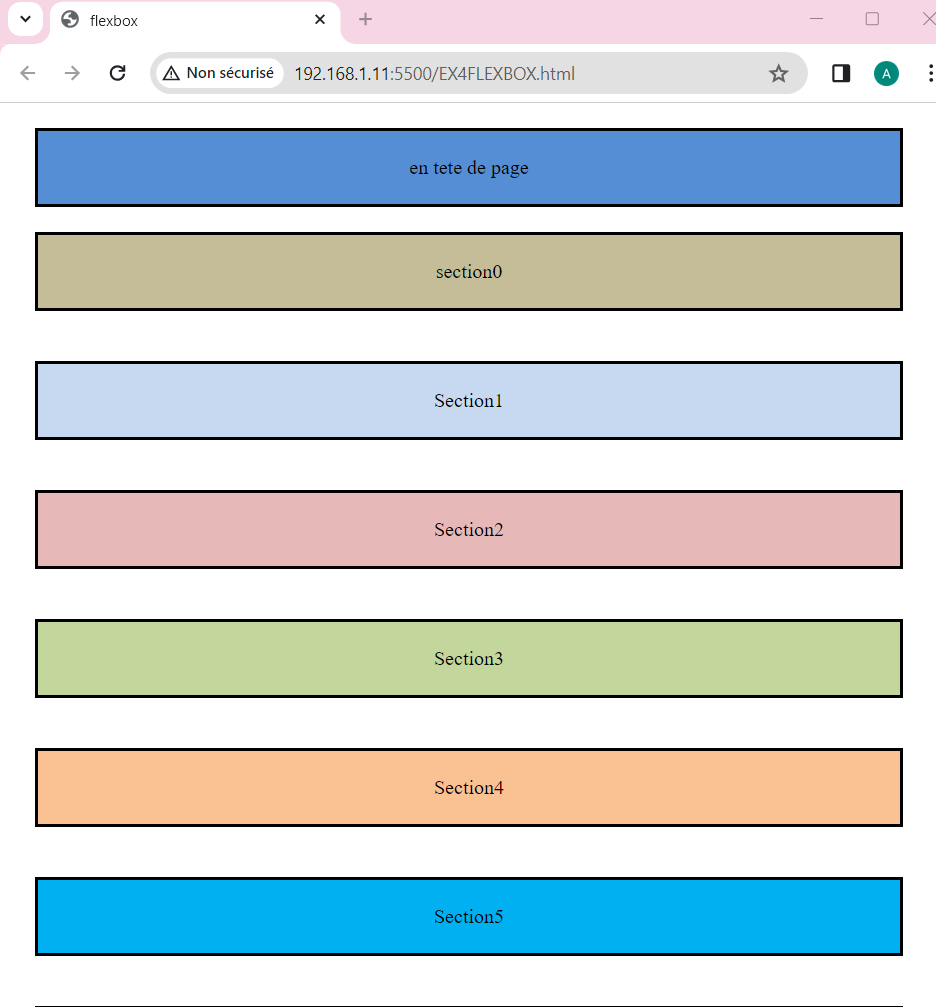




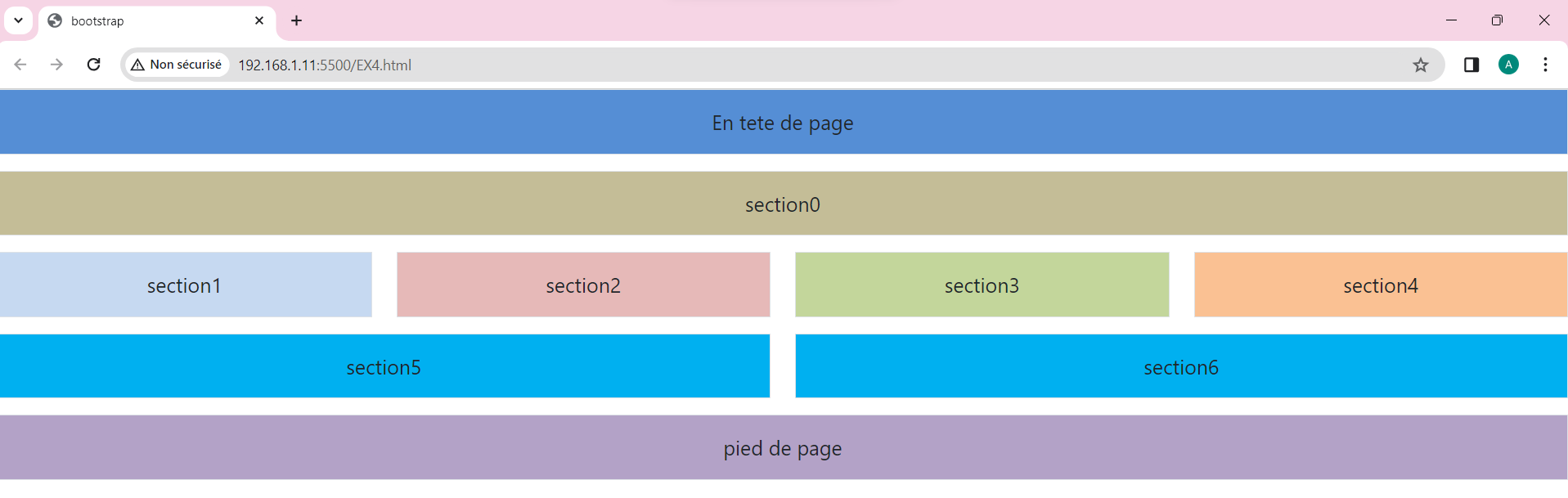
**EX4 :**

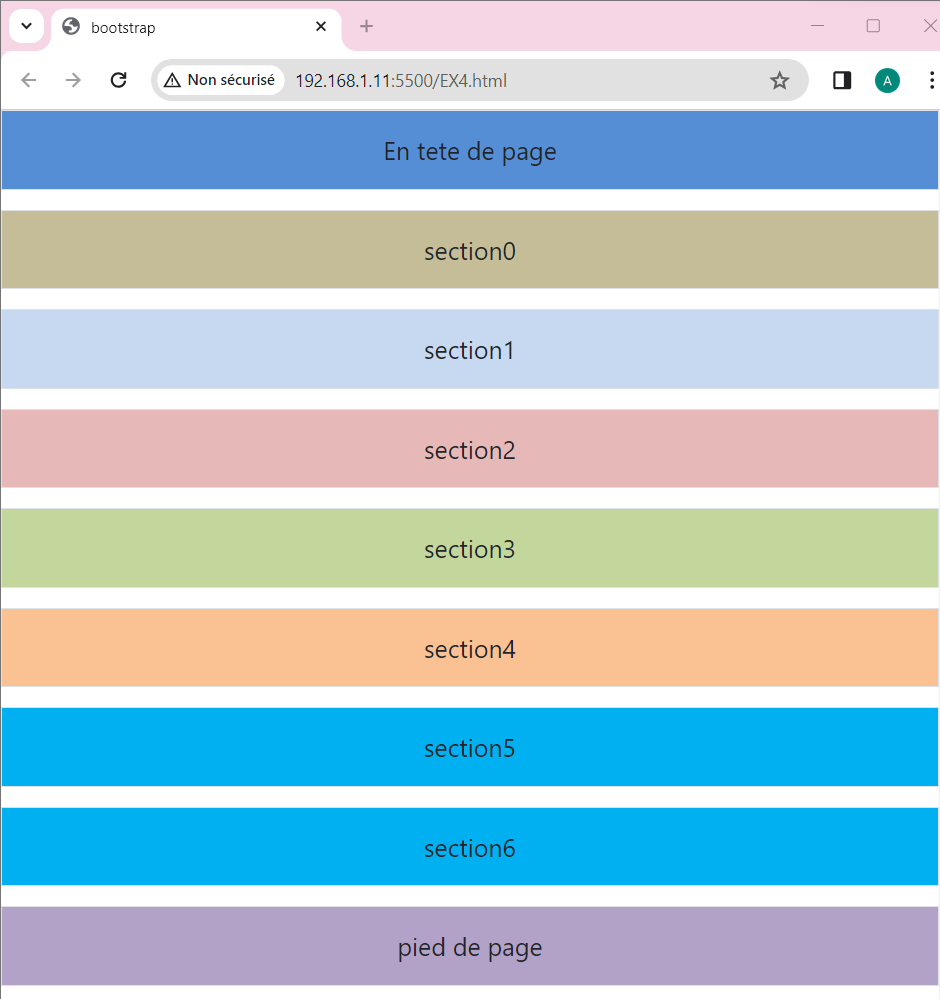
**En utilisant flexbox :**

****

****

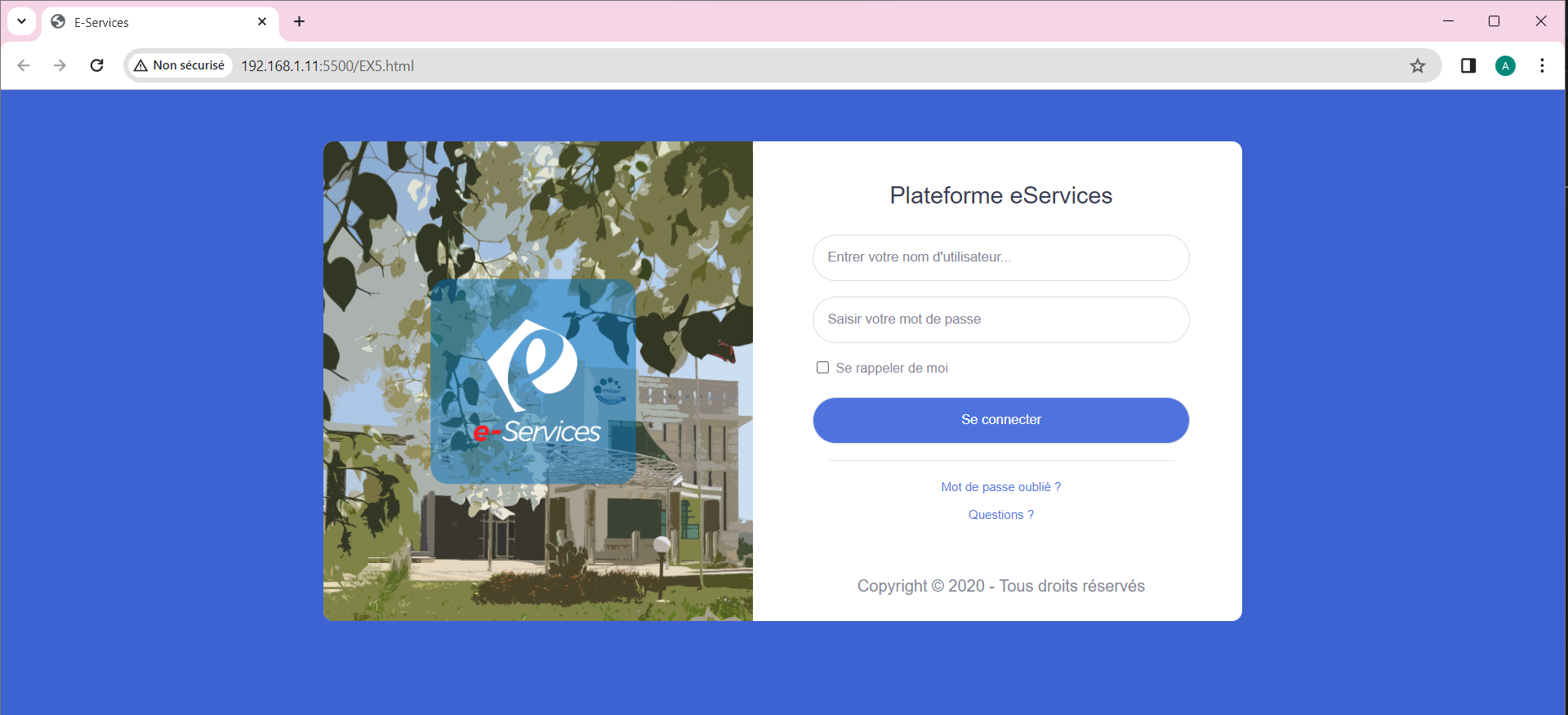
**En utilisant Bootstrap :**

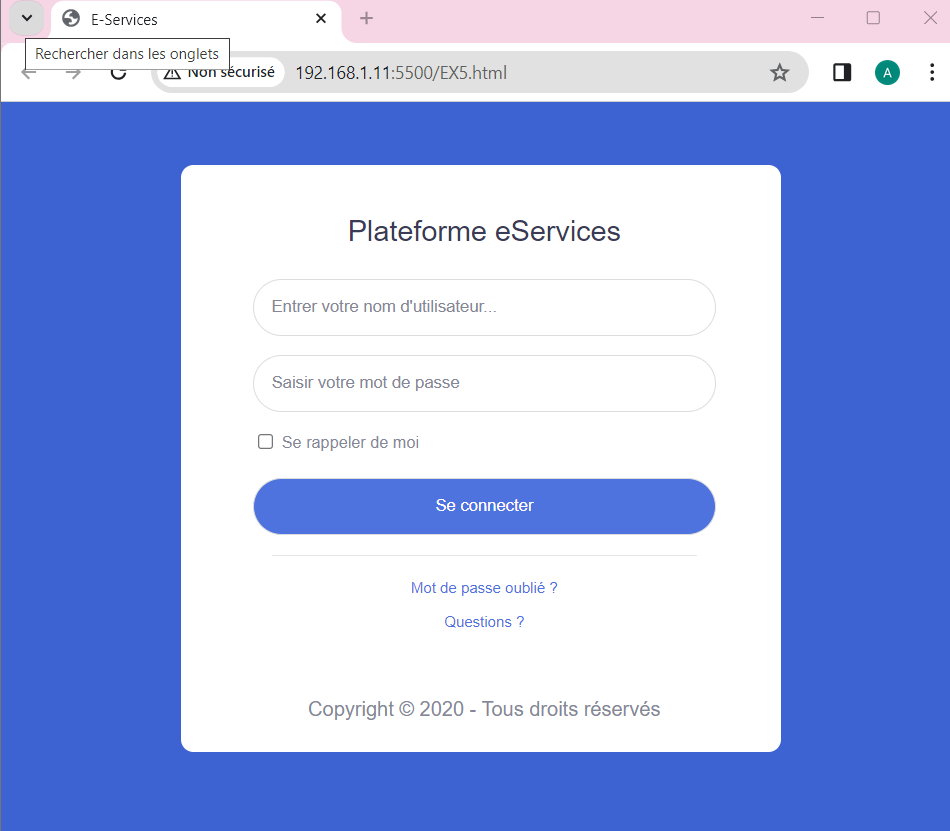
****

****

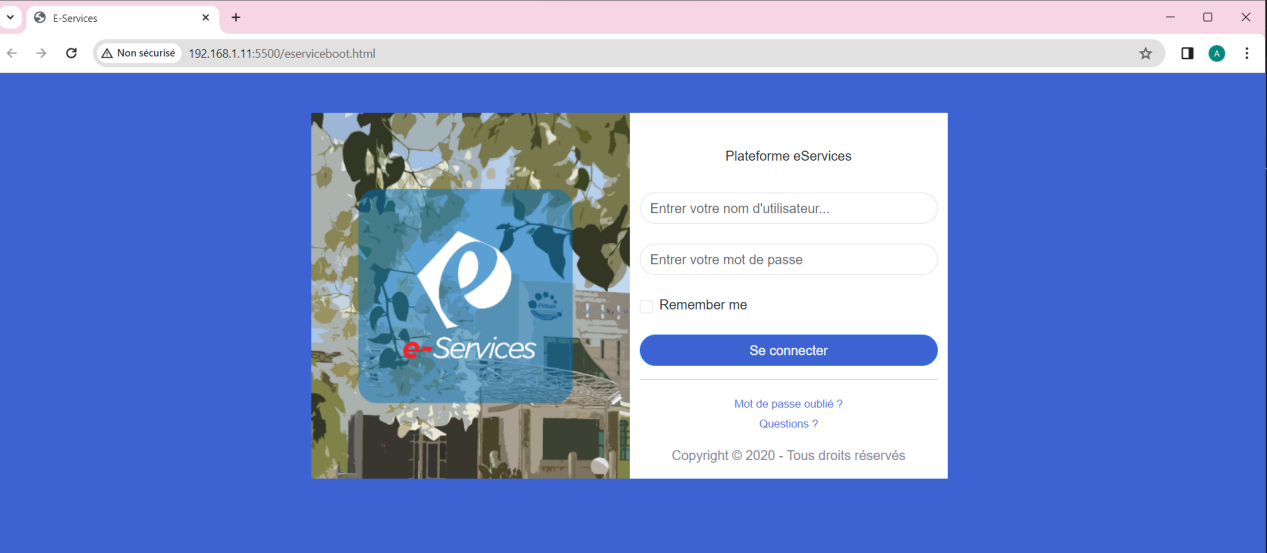
**EX5 :**

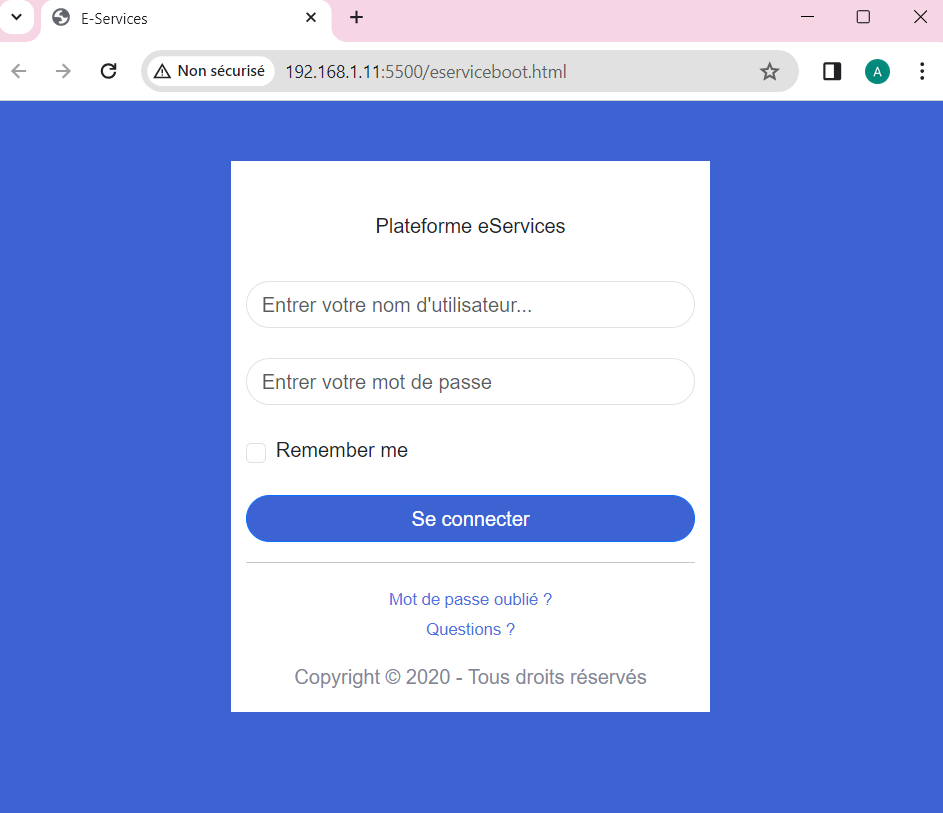
**Sans Bootstrap :**

****

****

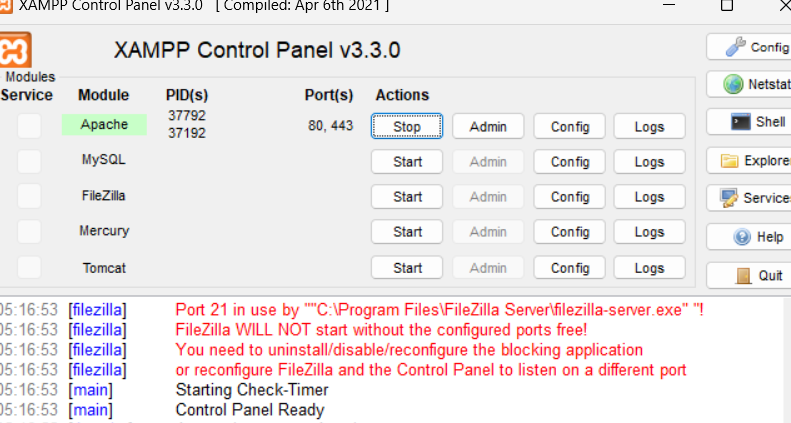
**En utilisant Bootstrap :**

****

****

**TP3 :**

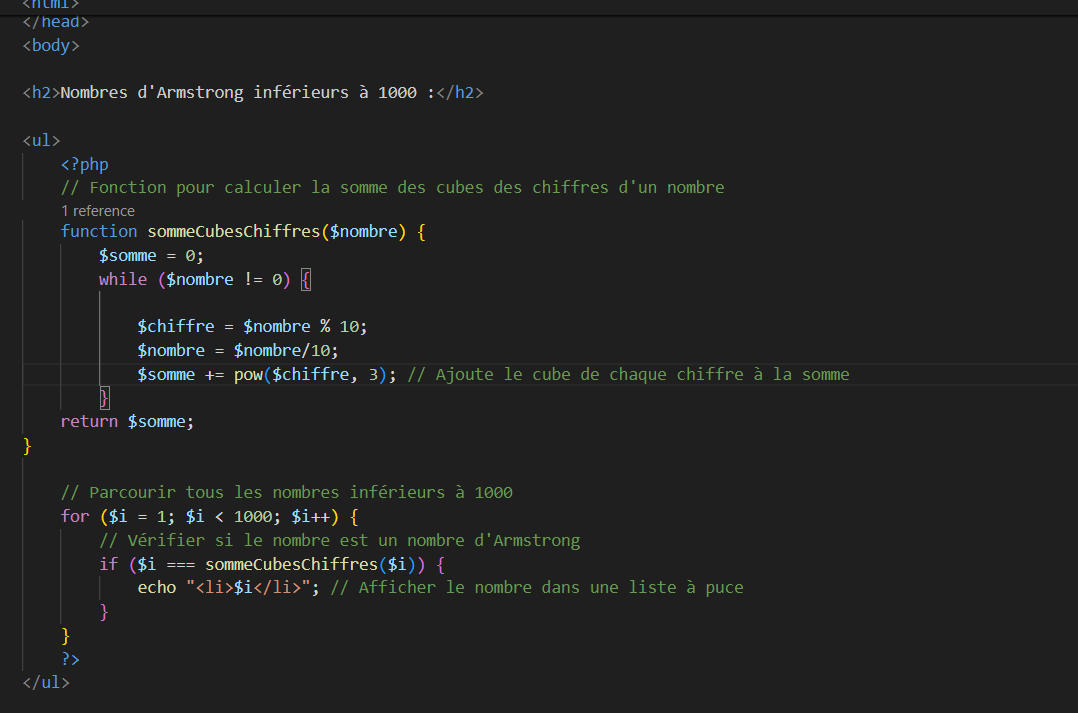
Premierement on va faire un dossier dans **(htdocs)** et après on va activer **Apache** pour qu’on peut exécuter nos programmes.

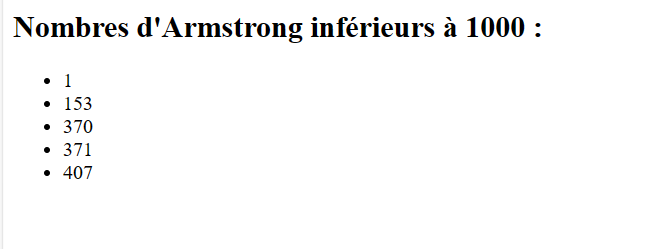
****

**EX1:**

On appelle nombres d’Armstrong les nombres entiers positifs tels que la somme des cubes de leurs chiffres est égale au nombre lui-même. Exemple : 153 est un nombre d’Armstrong. En effet :1\*1\*1+5\*5\*5+3\*3\*3 = 153

Écrire un programme PHP qui affiche tous les nombres d’Armstrong inférieurs à 1000 sous forme d’une liste à puce.

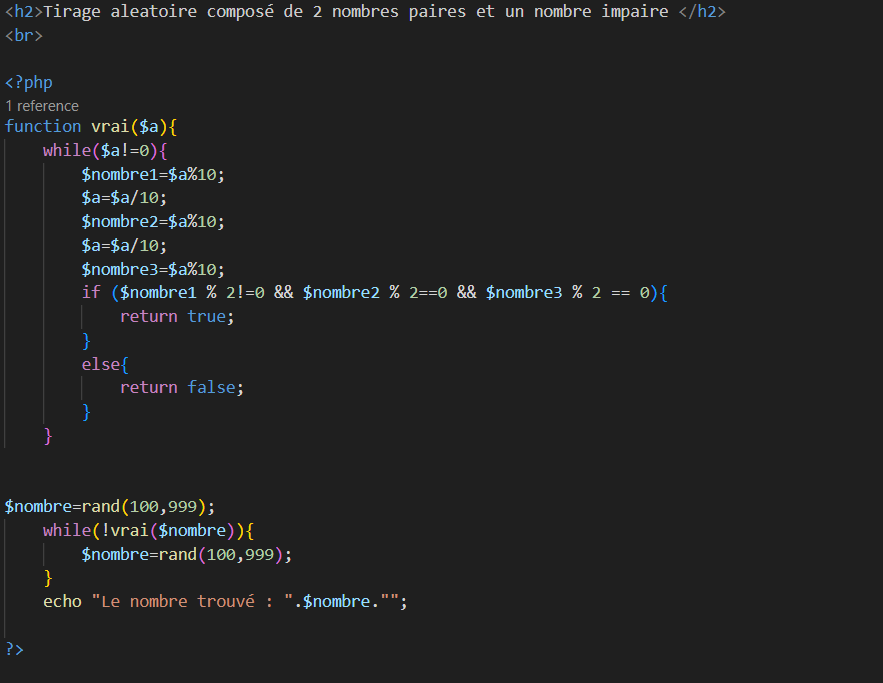




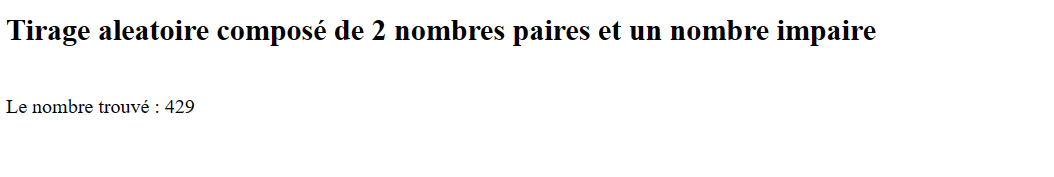
**EX2 :**

Ecrire un programme PHP qui effectue une suite de tirages de nombres aléatoires jusqu’à obtenir une suite

composée de deux nombres pair suivis d’un nombre impair.



Le resultat :

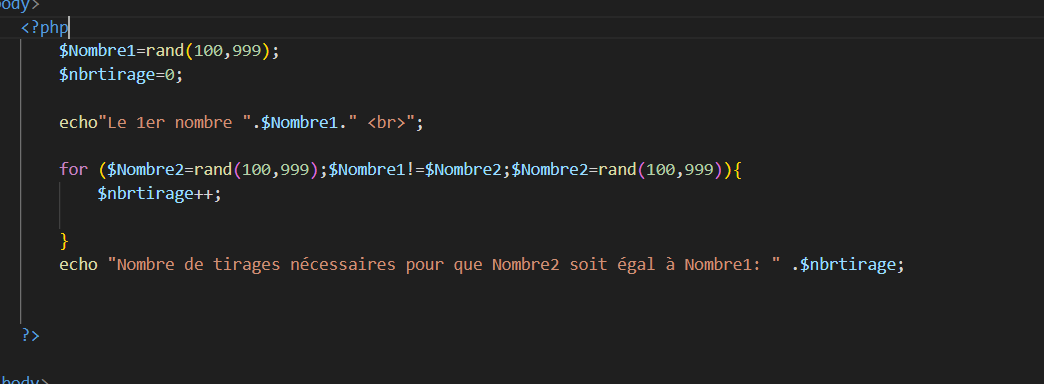


**EX4:**

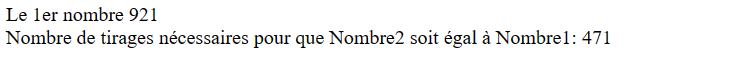
Ecrire un programme PHP qui permet de générer aléatoirement un nombre de trois chiffres N. Ensuite il effectue des tirages aléatoires jusqu’à réobtenir le même nombre N puis il affiche le nombre d’essaies réalisés.

Réalisez ce script d’abord avec l’instruction while puis avec l’instruction for.

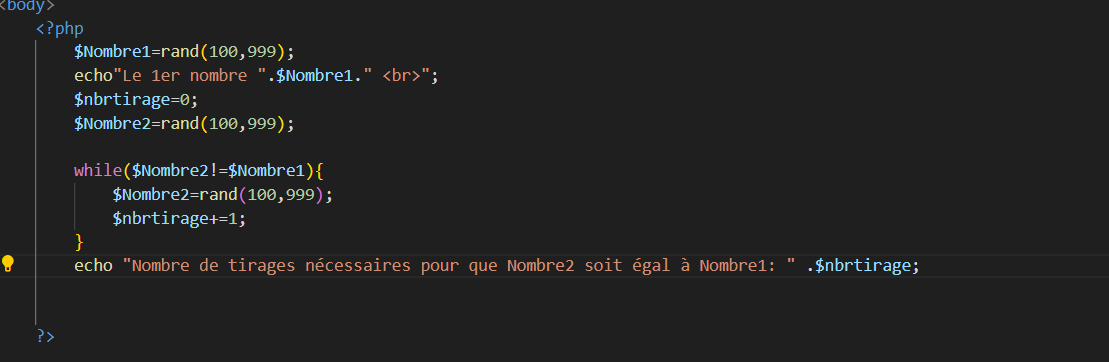
En utilisant la boucle for



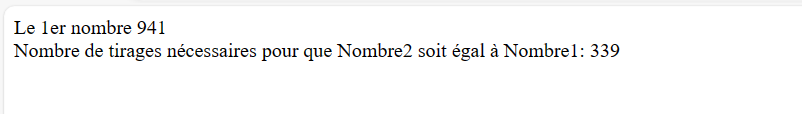
Le resultat :



En utilisant la boucle while :

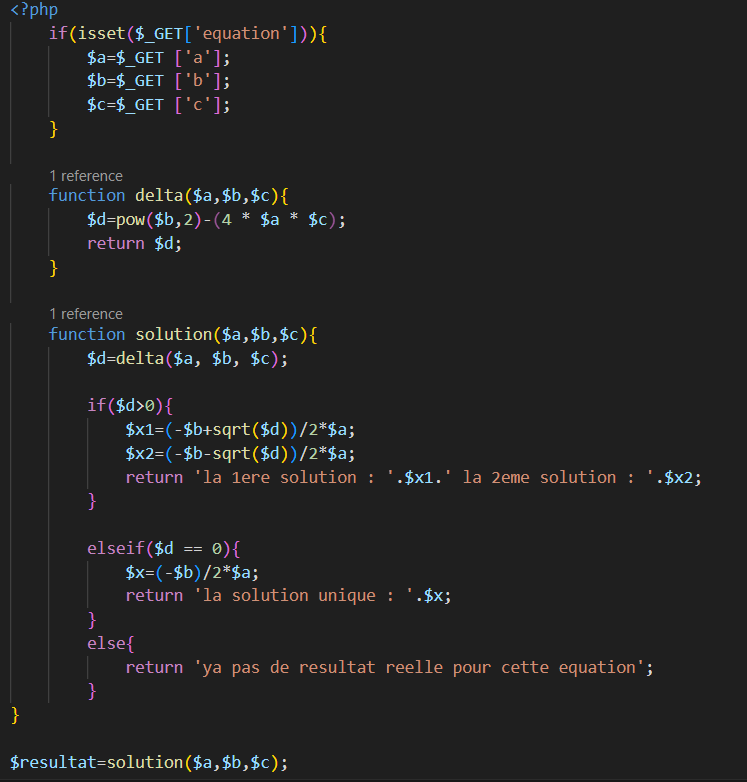


Le resultat :



**EX6 :**

Ecrire une application web qui permet de résoudre les équations du deuxième dégrée en PHP. L’utilisateur aura la possibilité de saisir les données nécessaires via un formulaire et l’application affiche la solution après la soumission de celui-ci.



Le resultat :

