



**Première Année Transformation Digitale et  
Intelligence Artificielle**

**Rapport TP2**

**Réalisé par :**

---

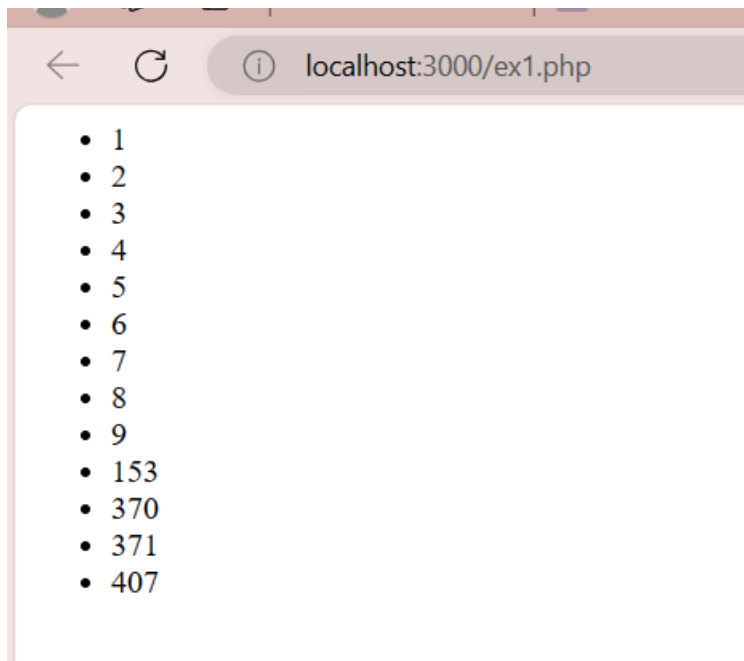
-Rim Agrouaz

---

**Encadré par :** Pr. Cherradi Mohamed

## Exercice 1 :

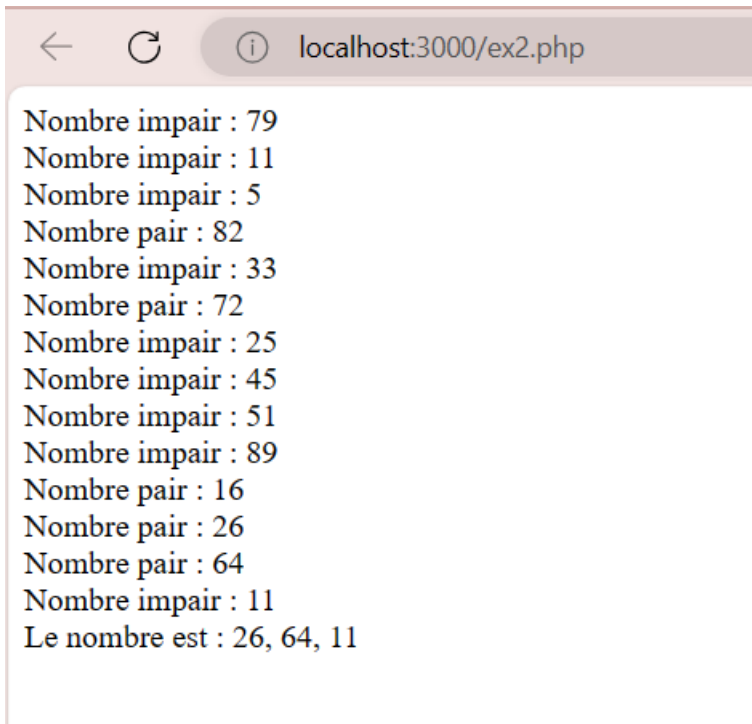
Dans cet exercice on doit afficher tous les nombres d'Armstrong inférieur à 1000. Afin de se faire, on divisera le nombre en plusieurs chiffres qu'on va calculer leur longueur, ensuite on créera une variable qui stockera la somme de ses chiffres à la puissance cubique. Puis on utilisera une boucle for pour lister tous les nombres d'Armstrong inférieur à 1000. Voici le résultat affiché :



## Exercice 2 :

Pour vérifier la parité du nombre généré, on a créé une fonction EstPaire, ensuite on va générer les chiffres en vérifiant leurs parités en même temps. En cas d'échec les variables seront affectées différemment et la boucle continue jusqu'à l'obtention du résultat désiré.

Voici le résultat de l'exécution du code :

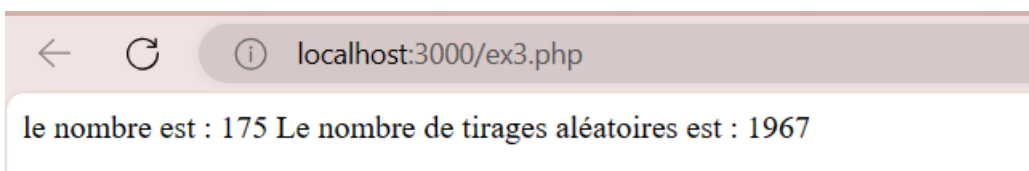


```
← ↻ ⓘ localhost:3000/ex2.php
Nombre impair : 79
Nombre impair : 11
Nombre impair : 5
Nombre pair : 82
Nombre impair : 33
Nombre pair : 72
Nombre impair : 25
Nombre impair : 45
Nombre impair : 51
Nombre impair : 89
Nombre pair : 16
Nombre pair : 26
Nombre pair : 64
Nombre impair : 11
Le nombre est : 26, 64, 11
```

### Exercice 3 :

Le but de cet exercice est de compter le nombre de tirage aléatoire pour ravoir le même nombre tiré au début. Pour se faire o définira une variable \$nombre qui stockera le nombre tiré en premier lieu, ensuite on créera une boucle qui vérifiera l'égalité du nombre tiré en premier avec le nombre tiré après tout en incrémentant le compteur de nombre de tirage.

Voici le résultat affiché :



```
← ↻ ⓘ localhost:3000/ex3.php
le nombre est : 175 Le nombre de tirages aléatoires est : 1967
```

### Exercice 4 :

## Année Universitaire : 2022/2023

Le but de cet exercice est de créer une application qui permet de résoudre les équations de second degré. On aura donc besoin d'un code HTML pour créer un formulaire de saisie pour les coefficients de l'application. Ensuite d'un code PHP pour résoudre l'équation, où on va récupérer les données entrées dans le formulaire puis on calcule "delta" et on écrit les conditions de vérifications pour trouver le résultat.

Voici le rendu final :



The screenshot shows a web browser window with the address bar displaying "localhost:3000/ex4.php". The page content is titled "Résolution de l'équation de 2nd degré". It features a form with four input fields: "Entrer la valeur de a", "Entrer la valeur de b", "Entrer la valeur de c", and a button labeled "Entrer". Below the form, the text "L'équation admet deux résultats : x1 = -0.1771243444677 et x2 = -2.8228756555323" is displayed.

FIN.