

Université Abdelmalek Essaadi
Ecole Nationale des Sciences Appliquées Al-Hoceima



RAPPORT DE TP3

Première Année TDIA

Réalisé par :

- **Abdelilah EL FEDG**

Encadré par : Pr. Mohamed CHERRADI

Exercice 1 :

Enonce de l'exercice :

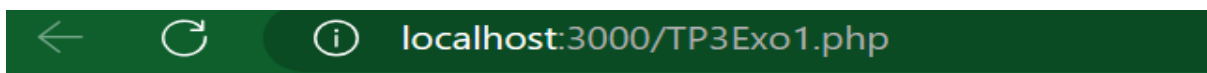
On appelle nombres d'Armstrong les nombres entiers positifs tels que la somme des cubes de leurs chiffres est égale au nombre lui-même. Exemple : 153 est un nombre d'Armstrong. En effet : $1*1*1+5*5*5+3*3*3 = 153$ Écrire un programme PHP qui affiche tous les nombres d'Armstrong inférieurs à 1000 sous forme d'une liste à puce.

Réponse :

- Le programme :

```
<?php
echo '<h3>Les nombres d\'armstrong inferieurs a 1000 sont:</h3>';
echo "<ul>";
for($i = 0; $i < 1000; $i++){
    $chiffres = str_split($i);
    $somme = 0;
    foreach($chiffres as $chiffre){
        $somme += pow($chiffre, 3);
    }
    if($somme === $i)
        echo "<li>$i</li>";
}
echo "</ul>"
?>
```

- Résultat de l'exécution :



Les nombres d'armstrong inferieurs a 1000 sont:

- 0
- 1
- 153
- 370
- 371
- 407

Exercice 2 :

Enonce de l'exercice :

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.

Réponse :

- Le programme :

```
<?php
$nombre = [];
while (true) {
    $nombre[0] = rand(1,10);
    $nombre[1] = rand(1,10);
    $nombre[2] = rand(1,10);

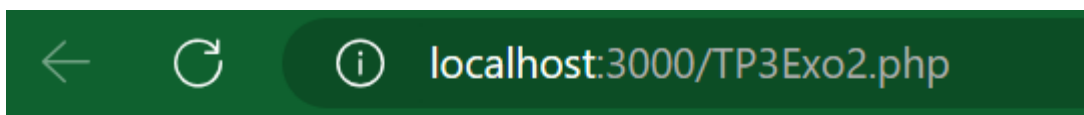
    if ($nombre[0] % 2 === 0 && $nombre[1] % 2 === 0 && $nombre[2] % 2 === 1) {
        break;
    }
}

// Affichage des nombres
echo "Premier nombre : $nombre[0]<br>";

echo "Deuxième nombre : $nombre[1]<br>";
echo "Troisième nombre : $nombre[2]";

?>
```

- Résultat de l'exécution :



Premier nombre : 4
Deuxième nombre : 2
Troisième nombre : 3

Exercice 3 :

Enonce de l'exercice :

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'essais réalisés.

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

Réponse (En utilisant la boucle while):

- Le programme :

```
<?php

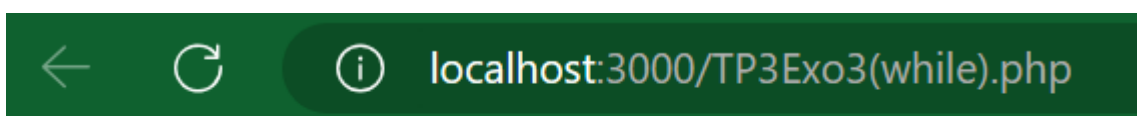
// Génération du nombre aléatoire à trois chiffres
$N = rand(100, 999);
$tirageAleatoire = rand(100, 999);
$compteur = 1;

while ($N !== $tirageAleatoire) {
    $N = rand(100, 999);
    $compteur++;
}

// Affichage du nombre d'essais
echo "Nombre d'essais : $compteur\n";

?>
```

- Résultat de l'exécution :



Nombre d'essais : 111

← ↻ ⓘ localhost:3000/TP3Exo3(while).php

Nombre d'essais : 180

← ↻ ⓘ localhost:3000/TP3Exo3(while).php

Nombre d'essais : 1363

Réponse (En utilisant la boucle for):

- Le programme :

```
<?php

// Génération du nombre aléatoire à trois chiffres
$N = rand(100, 999);
$nombre = rand(100, 999);

for ($compteur=1; $N !== $nombre; $compteur++) {
    $N = rand(100, 999);
}

// Affichage du nombre d'essais
echo "Nombre d'essais : $compteur\n";

?>
```

- Résultat de l'exécution :



localhost:3000/TP3Exo3(for).php

Nombre d'essais : 509

Exercice 3 :

Enonce de l'exercice :

Ecrire une application web qui permet de résoudre les équations du deuxième degré 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.

Réponse :

- Le code HTML :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Resoudre Equation de deuxieme degree</title>
</head>
<body>
  <form action="ResoudreEquation.php" method="post">
    <label for="a">Coefficient a:</label>
    <input type="number" id="a" name="a" required>
    <br>
    <label for="b">Coefficient b:</label>
    <input type="number" id="b" name="b" required>
    <br>
    <label for="c">Coefficient c:</label>
    <input type="number" id="c" name="c" required>
    <br>
    <button>Resoudre</button>
  </form>
</body>
</html>
```

- Le code PHP :

```

<?php
$a = $_POST['a'];
$b = $_POST['b'];
$c = $_POST['c'];

$delta = pow($b, 2) - 4 * $a * $c;

if ($delta < 0){
    $nbrSolutions = 0; // le variable "$nbrSolutions" stocke le nombre de solutions de l'equation.
} elseif ($delta === 0){
    $nbrSolutions = 1;
} else {
    $nbrSolutions = 2;
}

echo "<h2>Solutions Equation du second degre</h2>";
echo "<p>Equation : $a * x^2 + $b * x + $c = 0</p>";

if ($nbrSolutions === 0){
    echo "<p>l'equation n'a pas de solution reelle.</p>";
} elseif ($nbrSolutions === 1){
    $x = -$b / (2 * $a);
    echo "<p>l'equation a une solution unique : x = $x</p>";
} else {
    $x1 = (-$b + sqrt($delta)) / (2 * $a);
    $x2 = (-$b - sqrt($delta)) / (2 * $a);
    echo "<p>l'equation a deux solutions : x1 = $x1 et x2 = $x2</p>";
}
?>

```

- Résultat de l'exécution :

← ↻ ⓘ localhost:3000/index.html

Coefficient a:

Coefficient b:

Coefficient c:

← ↻ ⓘ localhost:3000/ResoudreEquation.php

Solutions Equation du second degre

Equation : $1 * x^2 + 1 * x + 0 = 0$

l'equation a deux solutions : $x1 = 0$ et $x2 = -1$

