

---

## TP3 : Fonctions

Ayoub KARINE (ayoub.karine@isen-ouest.yncrea.fr)

---

### Consignes :

Pour chaque TP,

- créer un dossier TP\_numTP (exemple : TP1)
  - dans le dossier TP\_numTP, créer un fichier spécifique portant le nom du tp en cours suivi de votre nom (ex : <TP1-PHP-VotreNom.php)
  - dans l'en tête du fichier php, placer un titre indiquant le nom du TP entre une balise <h1>
  - Mettre une ligne horizontale (<hr>)
  - Mettre le numéro de l'exercice entre une balise <h2>
  - Mettre une ligne horizontale entre les les exercices
  - Pour chaque exercice, mettre le numéro de question entre une balise <h3>
-

### Exercice 1

1. Créer une fonction qui incrémente une variable interne à chaque fois que cette fonction est appelée

### Exercice 2

1. Créer une fonction qui modifie la valeur d'une variable extérieure à la fonction, mais passée en argument (à cette fonction)

### Exercice 3

1. Créer les trois tableaux ci-dessous :

identite = ['alain', 'basile', 'David', 'Edgar'];

age = [1, 15, 35, 65];

mail = ['penom.nom@gtail.be', 'truc@bruce.zo',

'caro@caramel.org', 'trop@monmel.fr'];

2. Créer une fonction qui prend en argument un mail et qui retourne deux valeurs : le domaine (@...) et l'extension du mail (fr ou org ...).

**Indice** : utiliser les fonctions strstr() et substr()

3. Créer une fonction qui prend en argument les 3 tableaux de la question 1 et qui affiche, en choisissant un indice aléatoire, le texte suivant (si l'âge est inférieur, la fonction doit afficher "AGE an" à la place de "AGE ans" + la première lettre du prénom doit être en majuscule ) :

"je me nomme NOM j'ai AGE ans et mon mail est MAIL du

domaine DOMAINE avec l'extension EXTENSION";

**Exemple d'affichage :**

je me nomme Edgar j'ai 65 ans et mon mail est trop@monmel.fr  
du domaine monmel.fr avec l'extension .fr

**Indice :** utiliser les fonctions sizeof() et ucfirst()

#### Exercice 4

1. Créer 6 fonctions qui affichent chacune des étoiles de la façon suivante :

1- ligne() \* \* \* \* \*

2- carre\_plein()  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*

3- triangle\_iso()  
\*  
\* \*  
\* \* \*  
\* \* \* \*  
\* \* \* \* \*

4- carre\_vide()  
\* \* \* \* \*  
\*       \*  
\*       \*  
\*       \*  
\* \* \* \* \*

5- triangle\_vide()  
\*  
\* \*  
\*   \*  
\*       \*  
\* \* \* \* \*

6- triangle\_vide\_inv()  
\* \* \* \* \*  
\*       \*  
\*       \*  
\* \*  
\*

#### Exercice 5

**Principe :**

Un des plus anciens systèmes de cryptographie est celui de « César » qui consiste à décaler les lettres d'un message avec un certain décalage pour le rendre illisible. Ainsi pour un décalage de 2 les A deviennent des C, les C des E, etc. On suppose que le décalage est strictement positif.

## Exemple (pour un décalage=clé de 3):

Cryptage :

<b>Clair</b>	<b>A</b>	<b>T</b>	<b>T</b>	<b>A</b>	<b>Q</b>	<b>U</b>	<b>E</b>	<b>Z</b>		<b>A</b>	<b>S</b>	<b>T</b>	<b>E</b>	<b>R</b>	<b>I</b>	<b>X</b>
<b>Transcodé</b>	<b>0</b>	<b>19</b>	<b>19</b>	<b>0</b>	<b>16</b>	<b>20</b>	<b>4</b>	<b>25</b>		<b>0</b>	<b>18</b>	<b>19</b>	<b>4</b>	<b>17</b>	<b>8</b>	<b>23</b>
<b>Chiffré</b>	<b>3</b>	<b>22</b>	<b>22</b>	<b>3</b>	<b>19</b>	<b>23</b>	<b>7</b>	<b>28</b>		<b>3</b>	<b>21</b>	<b>22</b>	<b>7</b>	<b>20</b>	<b>11</b>	<b>26</b>
<b>Modulo 26</b>	<b>3</b>	<b>22</b>	<b>22</b>	<b>3</b>	<b>19</b>	<b>23</b>	<b>7</b>	<b>2</b>		<b>3</b>	<b>21</b>	<b>22</b>	<b>7</b>	<b>20</b>	<b>11</b>	<b>0</b>
<b>Chiffré</b>	<b>D</b>	<b>W</b>	<b>W</b>	<b>D</b>	<b>T</b>	<b>X</b>	<b>H</b>	<b>C</b>		<b>D</b>	<b>V</b>	<b>W</b>	<b>H</b>	<b>U</b>	<b>L</b>	<b>A</b>

Décryptage :

<b>Chiffré</b>	<b>D</b>	<b>W</b>	<b>W</b>	<b>D</b>	<b>T</b>	<b>X</b>	<b>H</b>	<b>C</b>		<b>D</b>	<b>V</b>	<b>W</b>	<b>H</b>	<b>U</b>	<b>L</b>	<b>A</b>
<b>Transcodé</b>	<b>3</b>	<b>22</b>	<b>22</b>	<b>3</b>	<b>19</b>	<b>23</b>	<b>7</b>	<b>2</b>		<b>3</b>	<b>21</b>	<b>22</b>	<b>7</b>	<b>20</b>	<b>11</b>	<b>0</b>
<b>Déchiffré</b>	<b>0</b>	<b>19</b>	<b>19</b>	<b>0</b>	<b>16</b>	<b>20</b>	<b>4</b>	<b>25</b>		<b>0</b>	<b>18</b>	<b>19</b>	<b>4</b>	<b>17</b>	<b>8</b>	<b>-3</b>
<b>+26, mod(26)</b>	<b>0</b>	<b>19</b>	<b>19</b>	<b>0</b>	<b>16</b>	<b>20</b>	<b>4</b>	<b>25</b>		<b>0</b>	<b>18</b>	<b>19</b>	<b>4</b>	<b>17</b>	<b>8</b>	<b>23</b>
<b>Déchiffré</b>	<b>A</b>	<b>T</b>	<b>T</b>	<b>A</b>	<b>Q</b>	<b>U</b>	<b>E</b>	<b>Z</b>		<b>A</b>	<b>S</b>	<b>T</b>	<b>E</b>	<b>R</b>	<b>I</b>	<b>X</b>

## Questions :

1. Créer une fonction qui implémente ce mécanisme dans le cas de chiffrement.
2. Créer une fonction qui implémente ce mécanisme dans le cas de déchiffrement.
3. Créer vos deux fonctions avec l'exemple ci-dessus

## Exercice 6

**Principe** : La méthode de cryptographie « Viginère » est similaire à celle de « César » sauf que la clé est sur plusieurs caractères

**Exemple** (pour un décalage=clé de 314):

Cryptage :

<b>Clair</b>	<b>A</b>	<b>T</b>	<b>T</b>	<b>A</b>	<b>Q</b>	<b>U</b>	<b>E</b>	<b>Z</b>		<b>A</b>	<b>S</b>	<b>T</b>	<b>E</b>	<b>R</b>	<b>I</b>	<b>X</b>
Transcodé	0	19	19	0	16	20	4	25		0	18	19	4	17	8	23
<b>Clé</b>	<b>3</b>	<b>1</b>	<b>4</b>	<b>3</b>	<b>1</b>	<b>4</b>	<b>3</b>	<b>1</b>		<b>4</b>	<b>3</b>	<b>1</b>	<b>4</b>	<b>3</b>	<b>1</b>	<b>4</b>
Chiffré	3	20	23	3	17	24	7	26		4	21	20	8	20	9	27
Modulo 26	3	20	23	3	17	24	7	0		4	21	20	8	20	9	1
<b>Chiffré</b>	<b>D</b>	<b>U</b>	<b>X</b>	<b>D</b>	<b>R</b>	<b>Y</b>	<b>H</b>	<b>A</b>		<b>E</b>	<b>V</b>	<b>U</b>	<b>I</b>	<b>U</b>	<b>J</b>	<b>B</b>

Décryptage :

<b>Chiffré</b>	<b>D</b>	<b>U</b>	<b>X</b>	<b>D</b>	<b>R</b>	<b>Y</b>	<b>H</b>	<b>A</b>		<b>E</b>	<b>V</b>	<b>U</b>	<b>I</b>	<b>U</b>	<b>J</b>	<b>B</b>
Transcodé	3	20	23	3	17	24	7	0		4	21	20	8	20	9	1
<b>Clé</b>	<b>3</b>	<b>1</b>	<b>4</b>	<b>3</b>	<b>1</b>	<b>4</b>	<b>3</b>	<b>1</b>		<b>4</b>	<b>3</b>	<b>1</b>	<b>4</b>	<b>3</b>	<b>1</b>	<b>4</b>
Déchiffré	0	19	19	0	16	20	4	-1		0	18	19	4	17	8	-3
+26, mod(26)	0	19	19	0	16	20	4	25		0	18	19	4	17	8	23
<b>Déchiffré</b>	<b>A</b>	<b>T</b>	<b>T</b>	<b>A</b>	<b>Q</b>	<b>U</b>	<b>E</b>	<b>Z</b>		<b>A</b>	<b>S</b>	<b>T</b>	<b>E</b>	<b>R</b>	<b>I</b>	<b>X</b>

## Questions :

1. Créer une fonction qui implémente ce mécanisme dans le cas de chiffrement.
2. Créer une fonction qui implémente ce mécanisme dans le cas de déchiffrement.
3. Créer vos deux fonctions avec l'exemple ci-dessus

## Exercice 7

1. Soit l'annuaire ci-dessous qui contient les numéros de téléphones de plusieurs personnes :

\$annuaire=array(

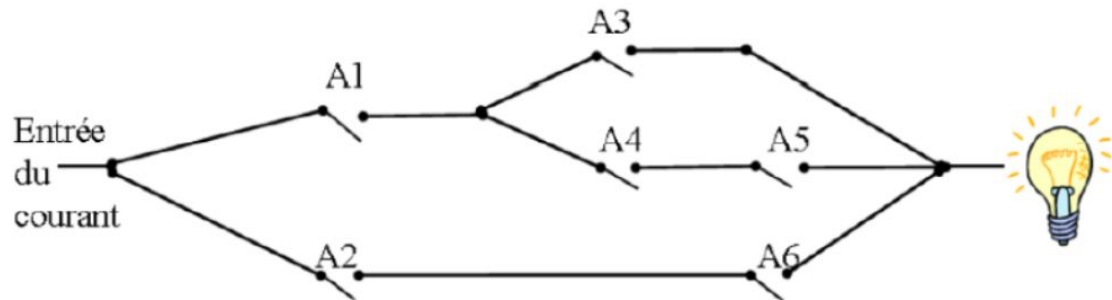
```
"PUJOL Olivier"=>"03 89 72 84 48",
"IMBERT Jo"=>"03 89 36 06 05",
"SPIEGEL Pierre"=>"03 87 67 92 37",
"THOUVENOT Frédéric"=>"01 42 86 02 12",
"MEGEL Pierre"=>"09 59 71 46 96",
"SUCHET Loïc"=>"03 89 33 10 54",
"GIROIS Francis"=>"03 88 01 21 15",
"HOFFMANN Emmanuel"=>"03 89 69 20 05",
"KELLER Fabien"=>"04 18 52 34 25",
```

```
"LEY Jean-Marie"=>"03 89 43 17 85",
"ZOELLE Thomas"=>"04 18 65 67 69",
"WILHELM Olivier"=>"03 89 60 48 78",
"BLIN Nathalie"=>"01 28 59 23 25",
"BICARD Pierre-Eric"=>"03 89 69 25 82",
"ZIEGLER Thierry"=>"03 89 06 33 89",
"BADER Jean"=>"03 89 25 65 72",
"ROSSO Anne-Sophie"=>"01 56 20 02 36",
"ROTTNER Thierry"=>"03 88 29 61 54",
"WEBER Joao"=>"03 89 35 45 20",
"SCHILLINGER Olivier"=>"03 84 21 38 40",
"BICARD Muriel"=>"03 89 33 47 99 ",
"KELLER Christian"=>"03 88 19 16 10 ",
"GROELLY Antonio"=>"03 89 33 60 63",
"ALLARD Aline"=>"03 89 56 49 19",
"WINNINGER Bénédicte"=>"04 16 14 86 66");
```

- a. Créer une fonction qui affiche le contenu dans un tableau HTML après avoir trié la liste par ordre alphabétique (utiliser la fonction ksort <https://www.php.net/manual/fr/function.ksort.php>)

## **Exercice 8**

1. Créer une fonction qui affiche l'état de la lampe (allumée ou éteinte) à base de l'état des interrupteurs. La lampe sera allumée dans les cas suivants :
  - a. A1 et A3 (circuit haut) sont fermés
  - b. A1, A4 et A5 (circuit du centre) sont fermés
  - c. A2 et A6 (circuit bas) sont fermés



### Exercice 9

Soit le tableau suivant :

clients = ["1"=>"Dulong","ville 1"=>"Paris","age 1"=>"35",  
 "2"=>"Leparc","ville 2"=>"Lyon","age 2"=>"47",  
 "3"=>"Dubos","ville 3"=>"Tours","age 3"=>"58"]

1. Ajoutez dans le tableau un client référencé comme client numéro 7, de nom Duval, domicilié à Nantes et âgé de 24 ans
2. Générer l'affichage ci-dessous :

Client	Nom	Ville	Age
client 1	Dulong	Paris	35
client 2	Leparc	Lyon	47
client 3	Dubos	Tours	58
client 7	Duval	Marseille	24

### Exercice 10

1. Ecrire une fonction qui vérifie si une chaîne de caractère est un palindrome. Exemple de palindrome : Madam

### **Exercice 11**

1. Ecrire une fonction qui vérifie si un numéro de carte bancaire est valide ou pas. Pour cela, il est demandé de se baser sur la formule de Luhn ([https://fr.wikipedia.org/wiki/Formule\\_de\\_Luhn](https://fr.wikipedia.org/wiki/Formule_de_Luhn))