

LP WAM - DU II

PHP : fonctions et procédures

TP à rendre n°03

Ce sujet regroupe différents traitements. Chaque procédure ou fonction peut être développée séparément. Une fois le programme principal développé, chaque développement d'une nouvelle fonctionnalité constitue un exercice à part entière.

Votre travail documenté est à me renvoyer par mail avant le 24 février, 22h00 sous la forme d'une archive DU_II-NOM_Prenom-PHP_Tp03.zip

On veut réaliser un programme qui gère une liste de personnes dans un environnement shell (ligne de commande). Chaque personne est caractérisée par son *identifiant*, son *nom*, son *prénom* et son *âge*. Le programme principal présentera le menu suivant :

- 1- Saisie d'une liste de personnes
- 2- Affichage de toutes les personnes
- 3- Sauvegarde dans un fichier
- 4- Chargement d'un fichier
- 0- Quitter

Choix (1,2,3,...) :

Chaque choix appellera une *fonction* ou une *procédure* qui effectuera le traitement demandé. Le tableau des personnes et le nombre de personnes seront définis comme des variables globales. Le tableau des personnes sera toujours trié et les doublons seront supprimés.

Choix 1 : voici un exemple de la saisie des personnes :

Choix (1,2,3,...) : 1

Entrez un ID, un nom, un prénom et un âge (ex : 1;Dupont;Jean;28) :
22;de la fontaine ; jean ; 110

Entrez un ID, un nom, un prénom et un âge (ex : 1;Dupont;Jean;28) :
33 ; martin ; jean christophe ; 21

Entrez un ID, un nom, un prénom et un âge (ex : 1;Dupont;Jean;28) :
10;dupont;jean;28

Entrez un ID, un nom, un prénom et un âge (ex : 1;Dupont;Jean;28) :

Les contraintes à respecter sont :

- Une ligne vide provoque une fin de saisie.
- La saisie des noms et des prénoms composés devra être possible.
- Les noms et les prénoms seront normalisés. Pour cela, faites une fonction de normalisation des noms et des prénoms. Dans un premier temps, cette fonction convertira uniquement en majuscule. Dans un deuxième temps, les noms et prénoms seront normalisés en majuscules, sans accent, ni valeur numérique, ni apostrophe. Les espaces multiples seront remplacés par un seul espace, puis chaque espace d'un nom ou d'un prénom composé sera remplacé par un tiret « - ». Par exemple « Léry » sera normalisé en « LERY », « de la fontaine » sera normalisé en « DE-LA-FONTAINE », ou « d'ortal » sera normalisé en « D-ORTAL ». De même pour les prénoms, « André pieRRe » sera transformé en ANDRE-PIERRE.

- L'identifiant de la personne ainsi que son âge seront convertis en entier.
- L'identifiant de la personne devra être unique, c'est-à-dire non utilisé par une autre personne.
- Si un des éléments est vide, ou si l'identifiant ou l'âge ne sont pas des numériques, un message d'erreur apparaît, et la personne n'est pas rangée dans le tableau des personnes.
- Les informations d'une personne saisie sont rangées dans un tableau associatif `$une_personne[]` avec les étiquettes « ID », « nom », « prenom », « age », puis chaque personne sera ensuite rangée dans un tableau numérique de personnes, `$tab_personnes[]`.
- Le tableau des personnes, `$tab_personnes[]`, sera trié à la fin de la saisie, et les doublons supprimés.

La figure suivante présente le tableau des personnes :

<code>\$tab_personnes</code>				
colonnes				
	<i>ID</i>	<i>nom</i>	<i>prenom</i>	<i>age</i>
Lignes 0	10	DUPONT	JEAN	28
1	22	DE-LA-FONTAINE	JEAN	110
2	33	MARTIN	JEAN-CHRISTOPHE	21

Choix 2 : voici un exemple de l'affichage des personnes :

Choix (1,2,3,...) : 2

Numéro :	ID	Nom	Prénom	Age
0 :	10	DUPONT	JEAN	28
1 :	22	DE-LA-FONTAINE	JEAN	110
2 :	33	MARTIN	JEAN-CHRISTOPHE	21

Les contraintes à respecter sont :

- avant chaque affichage le tableau sera trié (par défaut selon l'identifiant=ID), et les doublons supprimés ;
- la présentation de l'affichage d'un tableau sera réutilisable par d'autres modules comme lors d'une recherche multiple : faire une procédure d'affichage ayant un tableau de personnes à afficher en argument.

Choix 3 : voici un exemple de la sauvegarde dans un fichier de personnes :

Choix (1,2,3,...) : 3

Nom du fichier de sauvegarde : liste.txt

3 personne(s) sauvegardée(s) dans le fichier Sauvegardes/liste.txt

Les contraintes à respecter sont :

- Le nom du fichier sera normalisé via une fonction de normalisation :
 - Le nom du fichier sera en minuscules, sans accent. Les valeurs numériques seront autorisées. Les espaces multiples seront remplacés par un seul espace, puis chaque espace sera remplacé par un « _ ».
 - L'extension sera toujours .txt. Si le nom saisi ne contient pas d'extension, elle sera ajoutée, sinon l'extension saisie sera remplacée.
 - Le répertoire de sauvegarde sera « Sauvegardes/ ».
- Chaque ligne du fichier produit contiendra les informations d'une personne : l'identifiant (ID), le nom, le prénom et l'âge de la personne. Le caractère séparateur sera la tabulation.

Voici un exemple de fichier :

```
$ cat liste.txt
10  DUPONT          JEAN          28
22  DE-LA-FONTAINE  JEAN          110
33  MARTIN          JEAN-CHRISTOPHE  21
```

Choix 4 : voici un exemple du chargement de personnes :

```
Choix (1,2,3,...) : 4
Nom du fichier à charger : liste_personnes
Lecture de : Sauvegardes/liste_personnes.txt
15 personne(s) lue(s)
```

Les contraintes à respecter sont :

- Le nom du fichier sera normalisé de la même manière que pour la sauvegarde (même fonction de normalisation).
- Le chargement du fichier remplace toutes les données actuellement dans le tableau des personnes.
- L'identifiant de la personne devra être unique, c'est-à-dire non utilisé par une autre personne. Dans le cas contraire, la personne en conflit n'est pas chargée.
- Le tableau des personnes sera trié à la fin du chargement et les doublons supprimés.

Choix 0 : voici un exemple de la fin du programme :

```
Choix (1,2,3,...) : 0
Au revoir !
```