

Secteur Tertiaire Informatique
Filière « Etude et développement »

Séquence « Développer des pages Web »

TP PHP

Apprentissage

Mise en pratique

Evaluation



Mise en pratique : Découverte de PHP

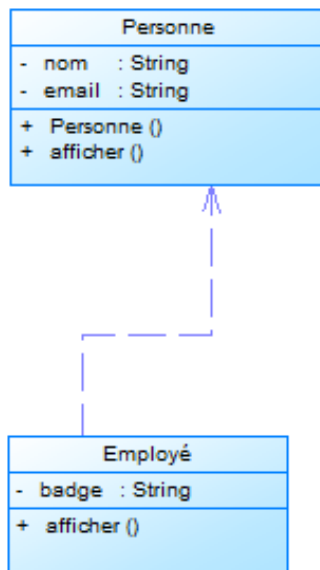
Afpa © 2019 – Section Tertiaire Informatique – Filière « Etude et développement »

TABLE DES MATIERES

1.	CLASSES ET OBJETS	3
2.	MODULES ET MODULES DE CLASSE.....	4
3.	CLASSE, HERITAGE ET CLONAGE	5

1. CLASSES ET OBJETS

Nous allons dans ce TP découvrir la syntaxe pour décrire une classe **Personne** exposant des champs et des méthodes. Cette classe sera ensuite étendue par le mécanisme d'héritage sous la forme d'une autre classe **Employé**.



- 1 – Créez un script **personnes.php** contenant une classe **Personne**. Déclarez dans cette classe deux champs **\$nom** et **\$email**.
- 2 – Hors de la classe, déclarez une variable **\$p1** qui est une instance de **Personne**. Affichez « **jean** » au champ **nom** de **p1** et jean@ailleurs.net au champ **email**.
- 3 – Effectuer un **var_dump** de la variable **p1**.
- 4 – Ajoutez une fonction **constructeur** à l'intérieur de la classe **Personne**. Initialisez les champs **nom** et **email** aux valeurs « **sans nom** », « **sans email** ».
- 5 – Déclarez une autre instance **p2** de la classe **Personne** puis affichez-la au travers d'un **var_dump**. Qu'en concluez-vous ?

6 – Ajoutez à la classe **Personne** une fonction **afficher()** qui imprime la valeur des champs **nom** et **email**. Appliquez cette méthode à l'instance **p1** puis à **p2**.

7 – Créez une classe **Employe** qui hérite de la classe **Personne**. Adjoignez-y un champ **\$badge**.

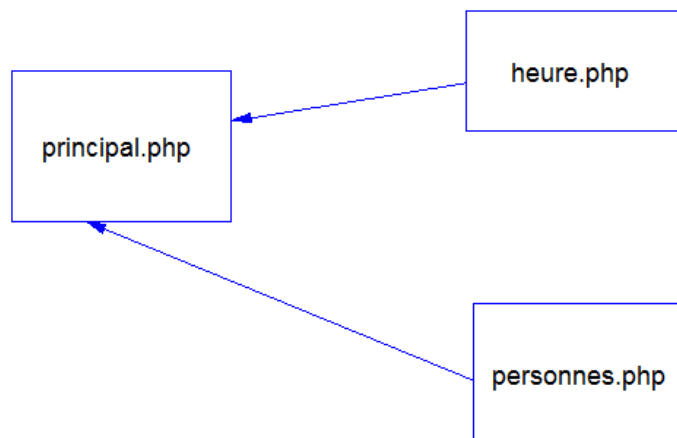
8 – Ajoutez une méthode **constructeur** à la classe **Employe** qui appelle la méthode **constructeur** de la classe parente **Personne** pour l'objet courant. Cette méthode initialise aussi le badge à « **numéro de badge inconnu** ».

9 – Déclarez une instance **\$e1** dans la classe **Employe**. Appliquez la méthode **afficher()** à cette instance. Qu'en concluez-vous ?

10 – Ecrivez une nouvelle version de la méthode **afficher()** dans la classe **Employe** qui tient compte du badge. Rappelez cette méthode pour l'objet **\$e1**. Interprétez le résultat.

2. MODULES ET MODULES DE CLASSE

L'objet de ce TP est de découvrir les conséquences du choix d'une portée de déclaration d'une variable ou d'une fonction dans un module inclus par la directive **include**.



1 – Créez un script **principal.php** contenant une page HTML vierge.

2 – Créez un autre script **heure.php** qui imprime l'heure au format de votre choix.

3 – Intégrez ce module dans le premier.

4 – Incluez à présent le script **personnes.php** réalisé lors du TP précédent. Qu'en concluez-vous ?

5 – Quelles règles proposeriez-vous pour segmenter une application en modules ?

Mise en pratique – Découverte de PHP

Afpa © 2019 – Section Tertiaire Informatique – Filière « Etude et développement »

3. CLASSE, HERITAGE ET CLONAGE

Créez une classe nommée **Form** représentant un formulaire HTML.

Le constructeur doit créer le code d'en-tête du formulaire en utilisant les éléments `<form>` et `<fieldset>`.

Une méthode **setText()** doit permettre d'ajouter une zone de texte.

Une méthode **setSubmit()** doit permettre d'ajouter un bouton d'envoi.

Une méthode **setReset()** doit permettre d'ajouter un bouton reset.

Les paramètres de ces méthodes doivent correspondre aux attributs des éléments HTML correspondants.

La méthode **getForm()** doit retourner tout le code HTML de création du formulaire.

Créer enfin une **méthode statique afficherCommentaire()** dans la classe **Form** qui renverra le message suivant « *Ce que vous pouvez faire avec des Classes !!!* »

Créez des objets **Form**, et ajoutez-y deux zones de texte et un bouton d'envoi et un bouton reset. Testez l'affichage obtenu.

Faite un appel à la méthode statique **afficherCommentaire()** de la classe **Form**.

Créez une sous-classe nommée **Form2** en dérivant la classe **Form**.

Cette nouvelle classe doit permettre de créer des formulaires ayant **en plus** des **boutons radio** et des **cases à cocher**.

Elle doit donc avoir les méthodes supplémentaires qui correspondent à ces créations.

Créez des objets, et testez le résultat.

Faite un appel à la méthode statique **afficherCommentaire()** en sollicitant cette fois la classe **Form2**. Que remarquez-vous ?

Créez un objet à partir de la classe **Form2**, puis créez-en un **clone**.

Modifiez certaines caractéristiques de cet objet, et affichez les deux formulaires obtenus.

Création de formulaires par Classe

Ce que vous pouvez faire avec des Classes !!!

Accès au site

Votre nom :

Votre code :

Envoyer

Annuler

Ce que vous pouvez faire avec des Classes !!!

Coordonnées et sports préférés

Votre nom :

Votre code :

Homme ☒ Femme ☐

Tennis ☐

Equitation ☐

Football ☐

Envoyer

Annuler

Mise en pratique – Découverte de PHP

Afpa © 2019 – Section Tertiaire Informatique – Filière « Etude et développement »

CREDITS

ŒUVRE COLLECTIVE DE l'AFPA

Sous le pilotage de la DIIP et du centre d'ingénierie sectoriel Tertiaire-Services

Equipe de conception (IF, formateur, mediatiseur)

Formateur : Alexandre RESTOUEIX

Date de mise à jour : 08/01/21

Reproduction interdite

Article L 122-4 du code de la propriété intellectuelle.

« Toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droits ou ayants cause est illicite. Il en est de même pour la traduction, l'adaptation ou la reproduction par un art ou un procédé quelconque. »

Mise en pratique – Découverte de PHP

Afpa © 2019 – Section Tertiaire Informatique – Filière « Etude et développement »