

TP_A_rendre1 :

P00 et Manipulation de fichiers

Question 1 : POO

Objectif :

- Créer une classe php
- Définir les propriétés et méthodes d'une classe.
- Définir un constructeur.
- Instancier la classe.

Énoncé :

1. Écrire une classe **Complexe** permettant de représenter des nombres complexes.

Un **nombre complexe** est un nombre qui peut s'écrire sous la forme :

$$Z = \text{PartieR  elle} + \text{PartieImaginaire} * i$$

O   PartieR  elle et PartieImaginaire sont des nombres r  els et i un nombre imaginaire tel que $i^2 = -1$.

Calcul avec des nombres complexes :

Soit les deux nombres complexes suivants : $z1 = a + ib$ et $z2 = x + iy$.

Addition : $z1 + z2 = (a + x) + i(b + y)$

Soustraction : $z1 - z2 = (a - x) + i(b - y)$

Produit : $z1 * z2 = (a * x - b * y) + i(x * b + a * y)$

Division : $z1 / z2 = ((a * x + b * y) / (x^2 + y^2)) + i * ((b * x - a * y) / (x^2 + y^2))$

2. Définir les méthodes d'acc  s aux attributs de la classe.
3. Définir un constructeur d'initialisation pour la classe.
4. Ajouter les méthodes suivantes :

Ajouter(Complexe) : Elle permet de renvoyer le nombre complexe obtenu en ajoutant au nombre en cours un nombre complexe pass   en argument.

Soustraire(Complexe)

Multiplier(Complexe).

Diviser(Complexe).

la m  thode magique __toString() : qui permet de repr  senter le nombre complexe sous la forme : $a + b * i$.

6.   crire un script permettant de tester la classe Complexe :

Exemple d'exécution :

le premier nombre complexe est :

$$3+1*I$$

le deuxieme nombre complexe est :

$$1+2*I$$

L'Addition des deux nombres renvoie :

$$4+3*I$$

La soustraction des deux nombres renvoie :

$$2-1*I$$

La multiplication des deux nombres renvoie :

$$1+7*I$$

La division des deux nombres renvoie :

$$1-0.2*I$$

//Source : www.exelib.net

Question 2 : Manipulation des Fichiers

On souhaite mettre en place un outil pour permettre à ses adhérents de déposer sur le serveur Web leurs devoirs de la semaine.

La page d'accueil de cet outil est présentée à la Figure 1. Elle propose de saisir un nom de fichier ou de parcourir le disque dur (disque du client) pour choisir le fichier. L'utilisateur peut aussi consulter la liste des devoirs (voir lien "*Consulter la liste des devoirs*").

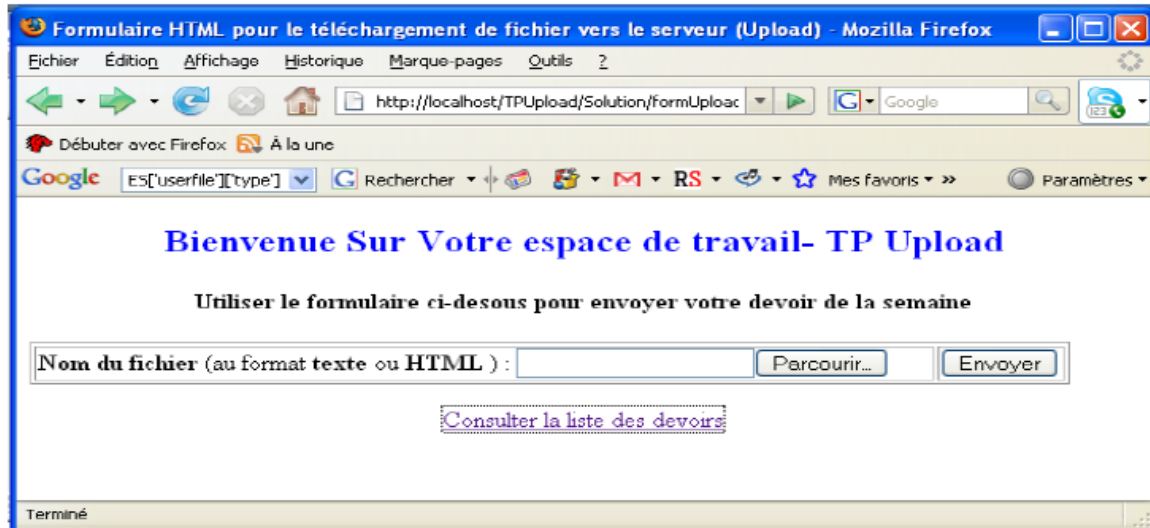


Figure 1 Page d'accueil de l'outil

Lorsque l'utilisateur clique sur le bouton *Envoyer*, il obtient la page présentée à la Figure 2. Cette page affiche un message de confirmation ainsi que l'aperçu du fichier déposé.

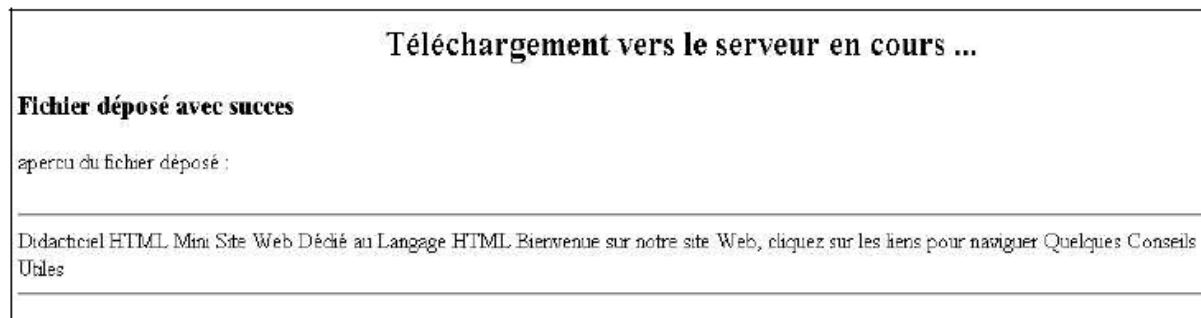


Figure 2. Confirmation & aperçu

Lorsque l'utilisateur clique sur le lien "*Consulter la liste des devoirs*", il obtient une page semblable à celle présentée à la Figure 3.



Figure 3. Liste des devoirs

Pour obtenir des informations sur un fichier de la liste (de la Figure 3), il suffit de cliquer sur son nom, on obtient alors la page présentée ci-dessous.

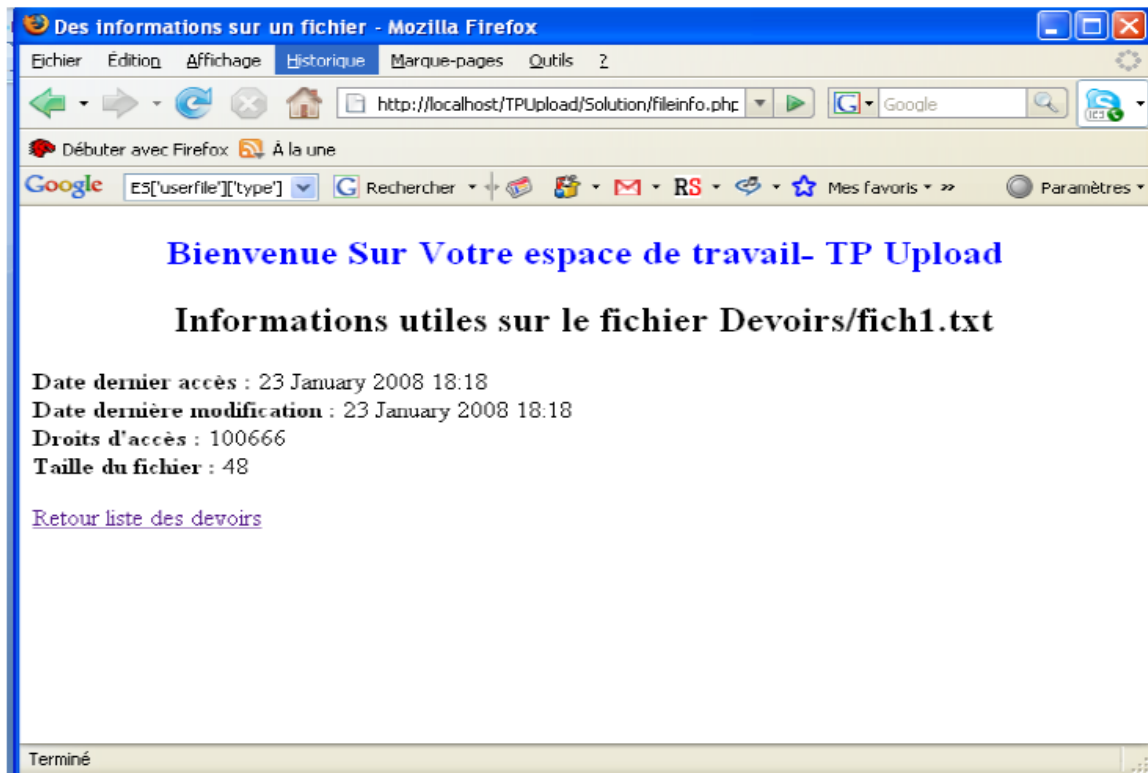


Figure 4. Informations sur un fichier de la liste des devoirs

Démarche de réalisation

L'application peut se découper en quatre fichiers :

- *formUpload.htm* pour afficher le formulaire de la figure 1
- *upload.php* pour générer la page de la figure 2
- *browseDir.php* pour générer la page de la figure 3
- *fileInfo.php* pour obtenir la page de la figure 4

formUpload.htm présente les spécificités suivantes :

- On utilise un élément *Input* de type *File* pour la saisie du nom de fichier.
- L'attribut *Enctype* de la balise *Form* doit avoir la valeur "*multipart/form-data*".

Lorsqu'un fichier est téléchargé vers un serveur Web, il est sauvegardé dans un emplacement temporaire du serveur. *Si vous ne déplacez pas le fichier avant la fin de l'exécution de votre script, le fichier sera supprimé de cet emplacement temporaire.*

En supposant que le formulaire contient un champ appelé *userfile* ("upload.php) la superglobale *\$_FILES[]* :

- *\$_FILES['userfile']['tmp_name']* : emplacement temporaire de stockage du fichier sur le serveur
- *\$_FILES['userfile']['name']* : nom du fichier sur le disque de l'utilisateur
- *\$_FILES['userfile']['size']* : taille du fichier
- *\$_FILES['userfile']['type']* : type MIME du fichier (ex: text/plain, text/html)

upload.php nécessite également l'utilisation des fonctions de manipulation de fichiers et plus particulièrement : *fopen(), fclose(), fread(), fwrite(), filesize(), move_uploaded_file(), is_uploaded_file()*

Le programme *browseDir.php* utilise les fonctions de gestion de répertoires comme par exemple : *opendir(), readdir(), closedir(), ...*

Le programme *fileInfo.php* fait appel aux fonctions d'identification des attributs d'un fichier : *fileatime(), filemtime(), fileperms(), filetype(), ...*

Question 3 : Session

Dans cet exercice on va mettre en pratique les sessions lors de l'authentification.

L'objectif est de créer trois pages PHP:

- **login.php**: qui authentifie l'utilisateur. Elle contient un formulaire renfermant une zone de texte, une zone de mot de passe et un bouton d'envoi.
- **session.php**: qui représente la page à accès limité. Aucun visiteur n'a le droit de voir son contenu s'il n'a pas été authentifié par la page **login.php**.
- **deconnexion.php**: est la page qui permet de déconnecter le client (détruire la session) et rediriger le navigateur vers la page **login.php**.

Si le client tente d'accéder directement à la page **session.php** alors qu'il n'est pas authentifié, il sera aussitôt redirigé vers la page **login.php**. Si'il fournit un bon login et un bon mot de passe alors il sera redirigé vers la page **session.php** qu'il a désormais le droit de consulter.

Pour simplifier, nous allons définir statiquement le bon login qui est "user" et le bon mot de passe qui est "1234".

Pour rediriger le navigateur automatiquement vers une autre page on peut le faire en HTML, Javascript ou en PHP. Cependant, il est préférable de le faire en PHP grâce à la fonction `header()`. Par exemple, si on veut rediriger le navigateur vers la page **session.php** alors on fait **header(location : session.php)**. Notez que la fonction `header()` est une fonction entête comme c'est le cas pour `session_start()`. Il faut donc prendre le soin de la déclarer avant d'envoyer du contenu au navigateur.