

Écriture de scripts PHP/ Interface

Informations complémentaires

Test de scripts PHP

Lors de ce TP, il vous sera demandé d'écrire des scripts `php` générant des pages `html`. Pour visualiser le code `html` produit par votre script `php`, il suffit d'appeler l'interpréteur `php` avec comme paramètre le nom du script :

```
$> php mon_script.php
<html><head>
<title> ...
...
```

Il est aussi possible de visualiser la page `html` générée par un script `php` dans un navigateur. Pour cela, vous devrez un créer un répertoire `public_html` à la racine votre répertoire sur `turing`. Tous les fichiers `php` (ou pages `html`) ajoutés dans `~/public_html/<chemin>/<script.php>` seront accessibles dans un navigateur à l'adresse suivante :

<http://codd.u-strasbg.fr/~<login>/<chemin>/<script.php>>

Attention : Le listener oracle n'est pas disponible sous `turing`. Les fichiers `html` devront être ajoutés dans le répertoire `public_html` de **turing** mais les pages seront visualisées à partir de **codd** à l'adresse suivante :

<http://codd.u-strasbg.u-strasbg.fr/~<login>/<page.html>>

Sécurité

Avant de visualiser le résultat de l'interprétation de vos scripts `PHP` dans un navigateur, il faudra s'assurer que le serveur web (`apache`) puissent accéder à votre répertoire `public_html` (droit de parcours des répertoires) et puissent lire vos scripts `php`.

Pour cela, vous pouvez utiliser les `acl`. La commande `getfacl` vous affiche les droits des différents utilisateur sur un fichier/répertoire.

Si vous voulez modifier ces droits, il faudra utiliser la commande `setfacl`. L'option `-m` de `setfacl` permet d'ajouter un droit `acl` à un fichier/répertoire.

Si par exemple, on souhaite utiliser l'utilisateur `apache` à lire un script `php`, on utilisera la commande suivante :

```
$>setfacl -m u:apache:r-- script.php
```

où le droit ajouté est de la forme `u(pour utilisateur) :<nomUtilisateur> :<droit>`

Pour changer les droits de l'utilisateur courant, du groupe ou des autres utilisateurs, on pourra utiliser les commandes suivantes :

```
$>setfacl -m u::rwx script.php (droit de l'utilisateur courant)
$>setfacl -m g::--- script.php (droit du groupe courant)
$>setfacl -m other::--- script.php (droit des utilisateurs)
```

Quelques liens

PHP : <http://fr.php.net/manual/fr/>

PHP/ORACLE : <http://php.net/manual/en/book.oci8.php>

Interface Client et interface Employé

On souhaite développer une interface web permettant un accès spécifique employé ou client :

- un employé de la banque doit pouvoir effectuer toutes les opérations spécifiques à sa fonction (ouverture de compte, dépôt, gestion des pénalités,...)
- un client doit pouvoir visualiser ses comptes et imprimer un relevé de compte (Vous êtes libre du choix de la présentation pour les résultats : paragraphe, liste, tableau, ...), effectuer des virements, imprimer un RIB, ...

1. Ecrire une page web permettant au client de se connecter et une page web permettant à l'employé de se connecter (identification par login + mot de passe).

2. Point de vue employé :

- a) Ecrire une page web permettant d'afficher la liste des clients dont le chargé de clientèle est responsable.

A chaque client sera associé un lien `html` permettant d'afficher les informations détaillées (coordonnées, liste des comptes, historiques, pénalités, rappel de prise de ...). Le lien `html` appelle le script `php` d'affichage des informations détaillées d'un produit avec comme paramètre l'identifiant du produit, passé au script par la méthode `GET`.

Un message sera aussi affiché à côté du client lorsque celui-ci l'objet d'une pénalité ou d'un rappel de prise de contact.

La page permettra aussi de contacter directement le client par mail.

- b) Le chargé de clientèle doit avoir la possibilité de traiter les créations et suppression de comptes clients :
- Une création suppose de rentrer dans un formulaire les renseignements concernant le client et les comptes qui vont être ouverts.
 - On peut supposer qu'une suppression du client peut être faite à partir du moment où tous les comptes du client ont été clôturés.

3. Point de vue Client :

- a) Ecrire la page permettant de visualiser l'ensemble des comptes du client (synthèse).

Le client pourra obtenir la liste des opérations effectuées sur un compte grâce à un lien `html` pour chaque compte.

- b) Sur cette page d'accueil, un menu déroulant permettra d'accéder aux différentes actions possibles du client :
- Effectuer un virement
 - Contacter son chargé de clientèle
 - Imprimer un RIB

Sécurité et intégrité des accès concurrents à la base

1. Assurer vous qu'il ne soit pas possible d'effectuer d'injection `SQL` à partir de votre site (il est conseillé de vérifier les entrées du site et de définir des requêtes utilisant des variables liées).
2. Vérifier que des accès concurrents ne puissent pas provoquer d'incohérences dans votre base. Pour cela, il est impératif d'utiliser les commandes de contrôle de transactions (`COMMIT` et `ROLLBACK`) et de poser des verrous sur les objets de la base si nécessaire (`SELECT ... FOR UPDATE`, `LOCK TABLE ...`).
3. Vous n'êtes pas administrateur de la base `Oracle`.
Indiquer les vues que vous devriez définir et les droits que vous transmettriez si vous étiez administrateur afin que les clients/employées ne puissent voir que les informations qui les concernent (tables, colonne, n-uplets, ...).