

Classe Login

Table des matières

1. Contenu	3
2. Définition.....	3
3. Méthodes publiques	3
1. Getters	3
getLogged	3
getLanguePref	3
2. Méthodes publiques.....	4
isLoggedIn.....	4
changeLangue	4
login.....	5
logout	5

1. Contenu

Ce document à pour but d'expliquer et de définir l'usage de la classe **Login** de l'outil **UniversalWeb**.

2. Définition

La classe Login a pour rôle de conserver les informations de l'utilisateur actuellement authentifié sur l'application. Etant donné qu'elle hérite directement de la classe User, toutes les propriétés et méthodes publique de la classe User sont donc disponibles pour la classe Login.

En plus des informations héritées, elle permet de conserver des préférences utilisateurs comme la langue. On trouve le code de cette classe dans le fichier **Login.class.php**.

3. Méthodes publiques

Getters

getLogged

Getter. Obtenir le statut « logué » de l'utilisateur.

Description

```
boolean getLogged();
```

Liste des paramètres

Aucun

Valeurs de retour

Booléen qui informe si l'utilisateur en cours (celui qui utilise l'application) est logué ou non.

Exemple

```
if ($_SESSION[_APP_LOGIN_]->getLogged()) {  
}
```

Note

La classe propose également la méthode **isLogged** qui a exactement la même fonctionnalité.

getLanguePref

Getter. Obtenir la langue préférée de l'utilisateur.

Description

```
string getLanguePref();
```

Liste des paramètres

Aucun

Valeurs de retour

Code ISO 3166 alpha2 correspondant au pays de la langue utilisée. Par défaut, la langue est 'fr'.

Classe Login

Exemple

```
$langue = $_SESSION['_APP_LOGIN_']->getLanguePref();
```

Note

La méthode scanne l'ordinateur à la recherche d'un cookie nommé `user_prefs` qui contient les préférences de l'utilisateur dont la langue de préférence. Si celui-ci est trouvé, elle renvoie l'information stockées ; dans le cas contraire la méthode renvoie 'fr', langue par défaut de l'application, après avoir créé le cookie `user_prefs` en question pour un appel ultérieur. C'est la méthode `changeLangue` qui permettra d'en changer le contenu.

Méthodes publiques

isLoggedIn

Obtenir le statut « logué » de l'utilisateur.

Description

```
boolean isLoggedIn();
```

Liste des paramètres

Aucun

Valeurs de retour

Booléen qui informe si l'utilisateur en cours (celui qui utilise l'application) est logué ou non.

Exemple

```
if ($_SESSION['_APP_LOGIN_']->isLoggedIn()) {  
}
```

Note

Cette méthode est identique au getter `getLogged` et possède exactement la même fonctionnalité.

changeLangue

Modifier la langue par défaut de l'utilisateur.

Description

```
changeLangue(string $langue);
```

Liste des paramètres

`$langue` : Code ISO 3166 alpha2 correspondant au pays de la langue utilisée. Choisir 'fr' (français) ou 'en' (anglais).

Valeurs de retour

Aucune.

Exemple

```
$_SESSION['_APP_LOGIN_']->changeLangue('en');
```

Note

La méthode enregistre dans le cookie de préférence nommé `user_prefs` le code ISO de la nouvelle langue de l'utilisateur.

login

Procède à la connexion (login) de l'utilisateur.

Description

```
integer|boolean login(string $id);
```

Liste des paramètres

\$id : identifiant de l'utilisateur à loguer.

Valeurs de retour

true : l'utilisateur a été logué.

User::ERREUR_USER_UNKNOWN : l'utilisateur est inconnu de la base de données.

Exemple

```
$retour = $_SESSION[_APP_LOGIN_]->login('thx1138');  
if ($retour !== true) {  
    echo 'erreur';  
}
```

Note

La méthode commence par charger les informations de l'utilisateur depuis la base de données, met à jour le `timestamp` de dernière visite, positionnel le flag 'logué' (accessible via getter `getLogged`) et inscrit l'information dans le journal des connexions.

logout

Procède à la déconnexion (logout) de l'utilisateur.

Description

```
logout();
```

Liste des paramètres

Aucun.

Valeurs de retour

Aucune.

Exemple

```
$_SESSION[_APP_LOGIN_]->logout();
```

Note

La méthode réinitialise l'objet `Login` en mémoire puis inscrit l'information de déconnexion dans le journal des connexions.