

<https://vader-fr.fr/spip.php?article174>



Exemples de fonctions PHP

- Développement web - Les pages dynamiques avec PHP/MySQL -



Date de mise en ligne : mercredi 27 juillet 2016

Copyright © Vader[FR] : ce n'est pas un blog, c'est un Sith - Tous droits

réservés

Retour des erreurs

Lorsque les erreurs et autres messages à caractère informatif sont gérés dans les différents blocs, et empilés dans une variable qui sera au final affichée dans un bloc dédié (lequel peut être dépliant grâce au CSS).

Si l'on souhaite un style différent selon le niveau d'erreur atteint, les différents blocs peuvent faire appel à cette fonction, qui use d'opérateurs ternaires.

Il est aussi possible d'utiliser une variable globale pour le niveau d'erreur actuel, ce qui simplifie les appels à la fonction.

3 niveaux d'erreur sont gérés :

- info : tout va bien, on affiche un (ou plus) message(s) juste pour information
- debug : attention, risque d'erreur, tout n'est pas optimal
- error : erreur critique

```
function raise_debuglvl($lvl,$actuel){
    $debuglvl=$actuel;
    $debuglvl=(( $lvl=="info" )&&($actuel=="info"))?"info":$debuglvl;
    $debuglvl=(( $lvl=="debug" )&&($actuel!="error"))?"debug":$debuglvl;
    $debuglvl=( $lvl=="error")?"error":$debuglvl;
    return $debuglvl;
}
```

La déclaration du bloc d'affichage :

```
// bloc d'information - l'image sert au déploiement mais reste affichée dans le bloc déplié
if ($debug!=""){
    switch ($debuglvl) {
        case "info" : $debugimg="notif-info.gif"; break; // ou action-s-valide
        case "debug" : $debugimg="notif-warning.gif"; break;
        case "error" : $debugimg="notif-error.gif"; break;
        default: $debugimg="notif-warning.gif"; break;
    }
    echo "

[\"$debuglvl\"]/
$debug
";
}
```

La page appelante déclare les variables `$debug` et `$info` :

- `$debug` contiendra les messages à afficher dans le bloc dédié
- `$info` contiendra les messages à afficher dans la page.

Les pages appelées (via `include` ou `require` empileront dans ces variables les messages qu'elles souhaitent soulever.

`$debug.="message";` ou `$info.="message";`, cette dernière variable pouvant être affichée dans la page puis vidée.

exemple d'utilisation

dans un script gérant le login d'un utilisateur, on récupère (en étape 2) login/mot de passe. Selon ce qui est fourni, on retourne en étape 1 en affichant un message d'erreur.

```
if ($etape==2){
    if ((isset($_POST["user"]))&&($_POST["user"]!="")){
        $user=$_POST["user"];
        // vérifier pas d'espace (entre autre)
        if (!preg_match("/^[a-z][a-z0-9]+$/",$user)){
            $debug.="nom utilisateur non conforme";
        }
    }else{
        $debug.="nom utilisateur manquant";
    }
    if ((isset($_POST["motpass"]))&&($_POST["motpass"]!="")){
        $motpass=$_POST["motpass"];
        // vérifier pas d'espace (entre autre)
        if (!preg_match("/^[a-zA-Z0-9]+$/",$user)){
            $debug.="mot de passe non conforme";
        }
    }else{
        $debug.="mot de passe manquant";
    }
    if ($debug!=""){
        $etape=1;
        $info=$debug;
        raise_debuglvl("debug",$debuglvl);
    }
}
if ($etape==2){
    // comparaison BDD
}
if ($etape==1){
    echo "
```

Bloc navigation

```
$info
";
// formulaire
}
```

son style CSS

pour un bloc dépliant CSS situé en haut à gauche de la page :

```
#debug {
  position:absolute;
  top:0;
  left:0;
  width:32px;
  height:32px;
  z-index:42;
  line-height:0;
  font-size:0;
}
#debug img {
  height:auto;
  width:auto;
  max-height:32px;
  max-width:32px;
  visibility:visible;
}
#debug:hover {
  height:auto;
  width:auto;
  min-height:100px;
  min-width:200px;
  line-height:normal;
  font-size:1em;
  overflow-y:auto;
}
#debug:hover img {
  height:auto;
  width:auto;
}
#debug:hover img#deplou_debug {
```

```
height:0;
width:0;
visibility:hidden;
}
/* ===== Messages de débogages
===== */
#debug.error{
background-color:#FF0000;
}
#debug.error:hover {
background-color:#FF8C00;
}

#debug.debug {
background-color:#FF8C00;
}
#debug.info {
background-color:#ADD8E6;
}
#debug.debug:hover, #debug.info:hover {
background-color:#ADD8E6;
}
.debug {
color:#FF8C00; /*#FFB30A;*/ /*#FFA501;*/
}
.info {
color:#40820D;
}
.error {
color:#FF0000;
font-weight:bold;
}
```

redirection en https

Pas vraiment une fonction, mais un petit bout de code à mettre dans la section

Si la page est accédée en http non sécurisé, on redirige vers le mode sécurisé https, via une balise meta.

```
// 'HTTPS' est défini à une valeur non-vide si le script a été appelé via le protocole HTTPS.
if ((!isset($_SERVER['HTTPS']))||($_SERVER['HTTPS']=="")){
$url="https://".$_SERVER['HTTP_HOST'].$_SERVER['REQUEST_URI'];
echo "<meta http-equiv=refresh content=0;url=$url>";
}
```