

# Les fonctions

Certaines fonctions ont été récupérées sur le WEB. Par exemple, la fonction "sendmail()" formate les mails pour qu'ils soient lus par la majorité des outils de mail.

Un panel de fonctions PHP pour générer des requêtes MySQL sont décrites ci-dessous. Ces fonctions sont tirées du programme MISSIONS.

## I) Initialisation

=> La "session" PHP est démarrée:

```
session_id($sid);  
session_name('mamiission');  
session_start();
```

=> Les variables `$_SESSION['connection']` et `['???']` prennent des valeurs NULL.

=> Les variables `$_SESSION['mission']` et `['???']` prennent des valeurs par défaut.

=> Le code source suivant est commun à tous les accès à la base de donnée, par la suite nous ne nous intéresserons qu'à la requête.

```
$link = mysql_connect($mysql_location,$mysql_user,$mysql_password)  
or die('Could not connect: ' . mysql_error());  
mysql_select_db($mysql_base) or die('Could not select database');
```

```
$query = 'SELECT * FROM T_UTILISATEUR WHERE UTILISATEUR = \''.$_SESSION[  
'connection'] ['utilisateur'] .\'';
```

```
$result = mysql_query($query) or die('Connection Mysql ; Query failed: ' . mysql_error());  
if ($result) {  
$line = mysql_fetch_array($result, MYSQL_NUM);
```

=> Si le mot de passe est valide pour cet utilisateur alors le script initialise les variables

```
$_SESSION['connection'] ['???'].
```

=> Si aucun identifiant de demande de mission n'est transmis à travers l'URL, alors le script initialise les variables `$_SESSION['mission']` et `['???']` correspondantes à l'utilisateur en ligne.

```
$query2 = 'SELECT * FROM T_CORRESPONDANCE ORDER BY GROUPE';
```

=> La requête ci-dessus initialise les variables `$_SESSION['correspondance']` et `['???'][$i]`.

```
}  
mysql_free_result($result);  
mysql_close($link);
```

--> Retour au [menu](#) <--

## II) Gestion des mots de passe

a) Changement de mot de passe

=> Vérification que les champs soient non vide.

=> La taille du nouveau mot de passe doit contenir entre 6 et 16 caractères.

```
$query = 'SELECT * FROM T_UTILISATEUR WHERE UTILISATEUR = \'$_SESSION[ 'connection' ][ 'utilisateur' ].\'';
```

=> Vérification de l'ancien mot de passe.

```
$query2 = 'UPDATE T_UTILISATEUR SET MOTDEPASSE = \'$_SESSION[ 'connection' ][ 'mot_de_passe_new2' ].\'' WHERE UTILISATEUR=\'$_SESSION[ 'connection' ][ 'utilisateur' ].\'';
```

=> Mise à jour du mot de passe.

=> Envoie d'un mail de rappel à l'utilisateur (appel à la fonction "sendmail()").

b) Rappel de mot de passe

```
$query = 'SELECT * FROM T_UTILISATEUR WHERE UTILISATEUR = \'$_SESSION[ 'connection' ][ 'utilisateur' ].\'';
```

=> Vérification de l'existence de l'utilisateur.

=> Envoie d'un mail de rappel à l'utilisateur.

--> Retour au [menu](#) <--

### III) Recherche simple

=> Récupération de l'identifiant de la mission à rechercher soit par URL soit par l'interface graphique.

```
$query = 'SELECT * FROM T_MISSION WHERE ID_MISSION = '.$ind;
```

=> Vérification des droits d'accès de l'utilisateur à cette mission.

=> Mise à jour des variables *\$\_SESSION[ 'mission' ][ '???' ]*.

```
$query2 = 'SELECT NOM,PRENOM,SS FROM T_UTILISATEUR WHERE UTILISATEUR = \'$_SESSION[ 'mission' ][ 'utilisateur' ].\'';
```

=> Récupération du nom et du prénom à partir du "nom d'utilisateur".

```
$query2 = 'SELECT DATE_FORMAT (\'$line[ "ALLER_DATE" ].\'' , \'%d/%m/%Y\')';
```

=> Mise en forme des dates.

--> Retour au [menu](#) <--

### IV) Recherche évoluée

=> Construction des "clauses where" en fonction des critères choisis par l'utilisateur. Les recherches peuvent être effectuées sur toute ou partie du "Nom", "Prénom" ou "Destination".

```
$where_nom = 'INSTR (T_UTILISATEUR.NOM ,UPPER(\'.$_SESSION[ 'recherche' ][ 'nom' ].\'))>0';  
$where_prenom = 'INSTR (T_UTILISATEUR.PRENOM ,UPPER( \'.$_SESSION[ 'recherche' ][ 'prenom' ].\'))>0';  
$where_groupe = 'T_MISSION.GROUPE = \'.$_SESSION[ 'recherche' ][ 'groupe' ].\'';  
$where_dest = 'INSTR (DESTINATION, UPPER(\'.$_SESSION[ 'recherche' ][ 'destination' ].\'))>0';
```

=> Concaténation des clauses.

=> Comptage du nombre de résultat.

```
$query = 'SELECT COUNT(*) FROM T_MISSION INNER JOIN T_UTILISATEUR ON T_MISSION.UTILISATEUR = T_UTILISATEUR.UTILISATEUR '.$where;
```

=> Construction de la clause "ORDER BY" pour le tri par date.

=> Ajout d'un interval de selection des résultats ("LIMIT").

```
$query = 'SELECT ID_MISSION, NOM ,PRENOM ,T_MISSION.GROUPE ,DESTINATION ,ALLER_DATE ,RETOUR_DATE ,VALIDE FROM T_MISSION INNER JOIN T_UTILISATEUR ON T_MISSION.UTILISATEUR = T_UTILISATEUR.UTILISATEUR '.$where.' '.$orderby.' LIMIT '.$_SESSION[ 'recherche' ][ 'limitebasse' ].','.$_SESSION[ 'recherche' ][ 'nb_par_page' ].'';
```

=> Construction d'une liste des résultats (affectation des variables `$_SESSION[ 'recherche' ][ $i ][ '???' ]`).

--> Retour au [menu](#) <--

## V) Ajout d'une demande

En mode "edition" la fonctionnalité est accessible par l'utilisateur.

=> Vérification que les champs soient non vide.

=> Recherche d'un identifiant de mission.

```
$query = 'SELECT MAX (ID_MISSION) FROM T_MISSION';
```

=> Insertion d'une nouvelle demande

```
$query = 'INSERT INTO T_MISSION( ID_MISSION, UTILISATEUR, GROUPE, DEPART, DESTINATION, OBJET, TYPE, TRANSPORT, ALLER_DATE, ALLER_H_DEPART, ALLER_H_ARRIVEE, RETOUR_DATE, RETOUR_H_DEPART, RETOUR_H_ARRIVEE, COMMENTAIRE, VALIDE) VALUES ( '.$ind.', '.$_SESSION[ mission ][ utilisateur ].', '.$_SESSION[ mission ][ groupe ].', '.$_SESSION[ mission ][ depart ].', '.$_SESSION[ mission ][ destination ].', '.$_SESSION[ mission ][ objet ].', '.$_SESSION[ mission ][ type ].', '.$_SESSION[ mission ][ transport ].',
```

```
STR_TO_DATE('\'. $_SESSION[ mission ][ date_aller ].\'|\'%d/%m/%Y\'),'. $_SESSION [ mission ][ heure_dep_aller ].'. $_SESSION[ mission ][ heure_arr_aller ].';  
STR_TO_DATE('\'. $_SESSION[ mission ][ date_retour ].\'|\'%d/%m/%Y\'),'. $_SESSION[ mission ][ heure_dep_retour ].'. $_SESSION[ mission ][ heure_arr_retour ].'.  
$_SESSION[ mission ][ commentaire ].'. $_SESSION[ mission ][ valide ].');
```

=> Envoie d'un mail au demandeur de mission, au responsable de service ou d'équipe et aux administratifs avec une URL pour accéder directement à la demande après identification.

--> Retour au [menu](#) <--

## VI) Valider une demande

Si la mission n'est pas terminée et que le responsable souhaite valider les demandes de missions, la fonctionnalité est accessible par l'utilisateur.

=> Validation de la demande par le responsable (qui a demandé la fonctionnalité de validation).

```
$query = 'UPDATE T_MISSION SET VALIDE="1" WHERE ID_MISSION = "'. $ind. "';
```

=> Envoie d'un mail au demandeur de mission, au responsable de service ou d'équipe et aux administratifs avec une URL pour accéder directement à la demande après identification.

--> Retour au [menu](#) <--

## VII) Annuler une demande

Si la mission n'est pas déjà annulée, et n'est pas terminée, la fonctionnalité est accessible par le demandeur et par le responsable.

=> Annulation de la demande par le demandeur de mission, le responsable de service ou d'équipe.

```
$query = 'UPDATE T_MISSION SET VALIDE="-1" WHERE ID_MISSION = "'. $ind. "';
```

=> Envoie d'un mail au demandeur de mission, au responsable de service ou d'équipe et aux administratifs avec une URL pour accéder directement à la demande après identification.