```php
<?php
/*******************************************************************************
AUTEUR      : Constantin Herrmann
LIEU        : CFPT Informatique Genève
DATE        : 14.06.2018

TITRE PROJET: KidsGames Geneva Score

TITRE PAGE  : staff.c
DESCRIPTION : Ce script contient toutes les fonctions concernant le staff
VERSION     : 1.0
*******************************************************************************/

/**
* Récupère un coach en fonction de son id
* Si le coach ou l'id n'éxiste pas alors le return sera égal à FALSE
*
* @param string id du membre du staff
* @return array un tableau avec le membre sélectionner par son id
*        ['id'] -> l'id du membre
*        ['nom'] -> le nom
*        ['prenom'] -> le prénom
*        ['age'] -> l'âge
*        ['idRole'] -> l'id de son Rôle
*        ['phone'] -> le numéro de téléphone
*/
function GetStaffById($id){
  static $query = null;
  if ($query == null) {
    $req = "SELECT `id`, `nom`, `prenom`, `age`, `idRole`, `phone` FROM `Staff`
    WHERE `id` = :id";
    $query = connecteur()->prepare($req);
  }
  try {
    $query->bindParam(':id', $id, PDO::PARAM_STR);
    $query->execute();
    $res = $query->fetch(PDO::FETCH_ASSOC);
  }
  catch (Exception $e) {
    error_log($e->getMessage());
    $res = false;
  }
  return $res;
}

/**
* Récupère tout les membres du staff
*
* @return array un tableau avec tout les membres
*      [index]
*        ['id'] -> l'id du membre
*        ['nom'] -> le nom
*        ['prenom'] -> le prénom
*        ['age'] -> l'âge
*        ['idRole'] -> l'id de son Rôle
*        ['phone'] -> le numéro de téléphone
*/
function GetAllStaff(){
```

```php
  static $query = null;
  if ($query == null) {
    $req = "SELECT `id`, `nom`, `prenom`, `age`, `idRole`, `phone` FROM `Staff`";
    $query = connecteur()->prepare($req);
  }
  try {
    $query->execute();
    $res = $query->fetchAll(PDO::FETCH_ASSOC);
  }
  catch (Exception $e) {
    error_log($e->getMessage());
    $res = false;
  }
  return $res;
}

/**
* Ajoute un membre à la table staff
*
* @param string le nom
* @param string le prénom
* @param string id du Rôle
*/
function AddStaff($nom, $prenom, $idRole){
  static $query = null;

  if ($query == null) {
    $req = "INSERT INTO `Staff`(`nom`, `prenom`, `idRole`) VALUES (:nom, :prenom,
    :idRole)";
    $query = connecteur()->prepare($req);
  }
  try {
    $query->bindParam(':nom', $nom, PDO::PARAM_STR);
    $query->bindParam(':prenom', $prenom, PDO::PARAM_STR);
    $query->bindParam(':idRole', $idRole, PDO::PARAM_STR);
    $query->execute();
    $query->fetch();
  }
  catch (Exception $e) {
    error_log($e->getMessage());
  }
}

/**
* Met à jour un membre du staff
*
* @param string id du membre à modifier
* @param string le nom
* @param string le prénom
* @param string id du Rôle
*/
function UpdateStaff($id, $nom, $prenom, $idRole){
  static $query = null;

  if ($query == null) {
    $req = "UPDATE `Staff` SET `nom`= :nom,`prenom`= :prenom,`idRole`= :role
    WHERE `id` = :id";
    $query = connecteur()->prepare($req);
```

```php
  }
  try {
    $query->bindParam(':id', $id, PDO::PARAM_STR);
    $query->bindParam(':nom', $nom, PDO::PARAM_STR);
    $query->bindParam(':prenom', $prenom, PDO::PARAM_STR);
    $query->bindParam(':role', $idRole, PDO::PARAM_STR);
    $query->execute();
    $query->fetch();
  }
  catch (Exception $e) {
    error_log($e->getMessage());
  }
}

/**
* Supprime un membre du staff
* Supprime également la liaison entre luo et son équipe si il à un équipe
*
* @param string id du membre à supprimer
*/
function DeleteStaff($id){

  // d'abord on supprime la liason
  RemoveCoachFromHisTeam($id);
  static $query = null;

  if ($query == null) {
    $req = "DELETE FROM `Staff` WHERE `id` = :id";
    $query = connecteur()->prepare($req);
  }
  try {
    $query->bindParam(':id', $id, PDO::PARAM_STR);
    $query->execute();
    $query->fetch();
  }
  catch (Exception $e) {
    error_log($e->getMessage());
  }
}

/**
* Retourne tout les coachs qui n'ont pas d'équipe
*
* @return array un tableau avec tout les membres
*      [index]
*        ['id'] -> l'id du membre
*        ['nom'] -> le nom
*        ['prenom'] -> le prénom
*/
function GetAllCoachsWithoutTeam(){
  static $query = null;
  if ($query == null) {
    $req = "SELECT `id`, `nom`, `prenom` FROM `Staff` WHERE `idRole` = 2 AND `id`
    NOT IN (SELECT idCoach FROM TEAMS WHERE id != 0) ORDER BY `prenom`";
    $query = connecteur()->prepare($req);
  }
  try {
    $query->execute();
    $res = $query->fetchAll(PDO::FETCH_ASSOC);
```

```
    $res = $query->fetchAll(PDO::FETCH_ASSOC);
  }
  catch (Exception $e) {
    error_log($e->getMessage());
    $res = false;
  }
  return $res;
}

/**
* Met à -1 l'équipe avec l'id du coach indiqué
*
* @param string id du coach
*/
function RemoveCoachFromHisTeam($idCoach){
  if ($query == null) {
    $req = "UPDATE `Teams` SET `idCoach`= -1 WHERE `idCoach` = :idCoach";
    $query = connecteur()->prepare($req);
  }
  try {
    $query->bindParam(':idCoach', $idCoach, PDO::PARAM_STR);
    $query->execute();
    $query->fetch();
  }
  catch (Exception $e) {
    error_log($e->getMessage());
  }
}

/**
* Retourne tout les arbitres de la table Staff
*
* @return array un tableau avec tout les arbitres
*      [index]
*        ['id'] -> l'id du membre
*        ['nom'] -> le nom
*        ['prenom'] -> le prénom
*/
function GetArbitres(){
  static $query = null;
  if ($query == null) {
    $req = "SELECT `id`, `nom`, `prenom` FROM `Staff` WHERE `idRole` = 3 ORDER BY
    `prenom`";
    $query = connecteur()->prepare($req);
  }
  try {
    $query->execute();
    $res = $query->fetchAll(PDO::FETCH_ASSOC);
  }
  catch (Exception $e) {
    error_log($e->getMessage());
    $res = false;
  }
  return $res;
}

/**
* Verifie si un membre es admin
*
```

```php
 * @param string id du membre
 * @return boolean true si il est admin, false si non
 */
function IsAdmin($id){
  static $query = null;

  if ($query == null) {
    $req = "SELECT `Staff`.`prenom` FROM `Staff` WHERE `Staff`.`idRole` = (SELECT
    `Role`.`id` FROM `Role` WHERE `Role`.`intitule` = 'Admin') AND `Staff`.`id` =
    :id";
    $query = connecteur()->prepare($req);
  }
  try {
    $query->bindParam(':id', $id, PDO::PARAM_STR);
    $query->execute();
    $res = $query->fetch(PDO::FETCH_ASSOC);
  }
  catch (Exception $e) {
    error_log($e->getMessage());
    $res = false;
  }

  if ($res != false)
  return true;
  else
  return false;
}

```