```php
<?php
//Fichier contenant les fonctions
include_once '../php/functions.php';

//Résultat à retourner
$res = array();

//Récupérer les données
$fav['idUser'] = filter_input(INPUT_POST, 'idUser', FILTER_VALIDATE_INT);
$fav['idImage'] = filter_input(INPUT_POST, 'idImage', FILTER_SANITIZE_STRING);

//Vérifier si le favoris existe
$res = checkFavExist($fav);

//Le favoris existe
if($res !== false){
    $res = true;
}

//JSON
header('Content-type: application/json');
echo json_encode($res);
?>

```

```php
<?php
//Constantes
//Connexion à la db
define("DB_HOST", "yd6zf.myd.infomaniak.com");
define("DB_NAME", "wpheaven");
define("DB_USER", "user");
define("DB_PASSWORD", "super2012");
//Lien pour les détails d'une image de Wallhaven
define("URL_API_DETAILS_WP", "https://wallhaven.cc/api/v1/w/");

//Connecteur pour ouvrir la connexion à la DB
function connecteur(){
    static $dbc = null;

    if ($dbc == null) {
        try{
            $dbc = new PDO('mysql:host=' . DB_HOST . ';dbname=' . DB_NAME, DB_USER, DB_PASSWORD);
        } catch (Exception $e) {
            echo 'Erreur : ' . $e->getMessage() . '<br/>';
            echo 'N° : ' . $e->getCode();
            die('Could not connect to MySQL');
        }
    }
    return $dbc;
}

?>
```

```php
<?php
//Fichier contenant le connecteur pour la DB
include_once 'dbconnect.php';

//Début des fonctions SQL

//Fonctions SELECT

//Récupère le salt de l'utilisateur
function getSaltUser($pseudo){
    $req = "SELECT salt FROM `user`
    WHERE pseudo = :pseudo";
    $sth = connecteur()->prepare($req);
    $sth->execute(array(':pseudo' => $pseudo));
    return $sth->fetch(PDO::FETCH_ASSOC)['salt'];
}

//Essaie de connecter l'utilisateur avec les informations de connexion
données
function login($user){
    $req = "SELECT idUser
    FROM user
    WHERE pseudo = :pseudo
    AND pwd = :pwd";
    $sth = connecteur()->prepare($req);
    $sth->execute(array(':pseudo' => $user['pseudo'], ':pwd' =>
$user['pwd']));
    $array = $sth->fetch(PDO::FETCH_ASSOC);
    return ($array == false ? false : $array['idUser']);
}

//Checker que l'email donné n'existe pas déjà
function checkEmailNotExist($email){
    $req = "SELECT email
    FROM user
    WHERE email = :email";
    $sth = connecteur()->prepare($req);
    $sth->execute(array(':email' => $email));
    $array = $sth->fetch(PDO::FETCH_ASSOC);
    return $array === false;
}

//Checker que le pseudo donné n'existe pas déjà
function checkPseudoNotExist($pseudo){
    $req = "SELECT pseudo
    FROM user
    WHERE pseudo = :pseudo";
    $sth = connecteur()->prepare($req);
    $sth->execute(array(':pseudo' => $pseudo));
    $array = $sth->fetch(PDO::FETCH_ASSOC);
    return $array === false;
}

//Checker si l'image donnée est dans les favoris de l'utilisateur
function checkFavExist($fav){
```

```php
function checkFavExist($fav){
    $req = "SELECT idFavorite
    FROM `favorite`
    WHERE idImage = :idImage
    AND idUser = :idUser";
    $sth = connecteur()->prepare($req);
    $sth->execute(array(':idImage' => $fav['idImage'], ':idUser' => $fav['idUser']));
    $array = $sth->fetch(PDO::FETCH_ASSOC);
    return ($array == false ? false : $array['idFavorite']);
}

//Récupérer les favoris de l'utilisateur
function getFavUser($idUser){
    $req = "SELECT idImage
    FROM `favorite`
    WHERE idUser = :idUser
    ORDER BY dateFavorite";
    $sth = connecteur()->prepare($req);
    $sth->execute(array(':idUser' => $idUser));
    return $sth->fetchAll(PDO::FETCH_ASSOC);
}

//Fin fonctions SELECT

//Fonctions CREATE

//Création d'un utilisateur
function createUser($user){
    $req = "INSERT INTO `user` (`idUser`, `pseudo`, `email`, `pwd`, `salt`)
    VALUES (null, :pseudo, :email, :pwd, :salt)";
    $sth = connecteur()->prepare($req);
    $sth->execute(array(':pseudo' => $user['pseudo'], ':email' => $user['email'],
    ':pwd' => $user['pwd'], ':salt' => $user['salt']));
    return connecteur()->lastInsertId();
}

//Ajout d'un favoris
function addFav($fav){
    $req = "INSERT INTO `favorite` (`idFavorite`, `idImage`, `idUser`)
    VALUES (null, :idImage, :idUser)";
    $sth = connecteur()->prepare($req);
    $sth->execute(array(':idImage' => $fav['idImage'], ':idUser' => $fav['idUser']));
    return connecteur()->lastInsertId();
}

//Fin fonctions CREATE

//Fonctions EDIT

//Fin fonctions EDIT

//Fonctions DELETE
```

```php

//Suppression d'un favoris
function delFav($fav){
    $req = "DELETE FROM `favorite` WHERE idImage = :idImage AND idUser = :idUser";
    $sth = connecteur()->prepare($req);
    $sth->execute(array(':idImage' => $fav['idImage'], ':idUser' => $fav['idUser']));
}

//Fin fonctions DELETE

?>

```

```php
<?php
//Fichier contenant les fonctions
include_once '../php/functions.php';

//Résultat à retourner
$res = array();

//Récupérer les données
$idUser = filter_input(INPUT_POST, 'idUser', FILTER_VALIDATE_INT);

$res = getFavUser($idUser);
$currentFav = array();

for ($i=0; $i < count($res); $i++) {
    //Récupérer les infos du wallpaper
    $currentFav = json_decode(file_get_contents(URL_API_DETAILS_WP . $res[$i]['idImage']))->data;
    //Supprimer les champs inutiles
    unset($currentFav->uploader);
    unset($currentFav->tags);
    //L'ajouter au résultat
    $res[$i] = $currentFav;
}

//JSON
header('Content-type: application/json');
echo json_encode($res);
?>
```

```php
<?php
//Fichier contenant les fonctions
include_once '../php/functions.php';

//Résultat à retourner
$res = array();

//Récupérer les données
$user['pseudo'] = filter_input(INPUT_POST, 'pseudo', FILTER_SANITIZE_STRING);
$salt = getSaltUser($user['pseudo']);

if($salt == null){
    $res = false;
}else{
    $user['pwd'] = hash('sha256', $salt . filter_input(INPUT_POST, 'pwd', FILTER_SANITIZE_STRING));

    //Essaie de connecter l'utilisateur
    $res = login($user);
}

//JSON
header('Content-type: application/json');
echo json_encode($res);
?>
```

```php
<?php
//Fichier contenant les fonctions
include_once '../php/functions.php';

//Résultat à retourner
$res = array('done' => false, 'msg' => false);

//Récupérer les données
$user['pseudo'] = filter_input(INPUT_POST, 'pseudo', FILTER_SANITIZE_STRING);
$user['email'] = filter_input(INPUT_POST, 'email', FILTER_SANITIZE_STRING);
$user['pwd'] = filter_input(INPUT_POST, 'pwd', FILTER_SANITIZE_STRING);

//Vérifier que l'email existe pas
if(checkEmailNotExist($user['email'])){
    //Vérifier que le pseudo existe pas
    if(checkPseudoNotExist($user['pseudo'])){
        //Créer le salt et le mdp crypté
        $user['salt'] = hash('sha256', random_bytes(100));
        $user['pwd'] = hash('sha256', $user['salt'] . $user['pwd']);

        //Créer l'user
        $res['msg'] = createUser($user);

        //Tout s'est bien passé
        $res['done'] = true;
    }else{
        $res['msg'] = "Username is already in use";
    }
}else{
    $res['msg'] = "The email is already in use";
}

//JSON
header('Content-type: application/json');
echo json_encode($res);
?>
```

```php
<?php
//Fichier contenant les fonctions
include_once '../php/functions.php';

//Résultat à retourner
$res = array();

//Récupérer les données
$fav['idUser'] = filter_input(INPUT_POST, 'idUser', FILTER_VALIDATE_INT);
$fav['idImage'] = filter_input(INPUT_POST, 'idImage', FILTER_SANITIZE_STRING);

//Vérifier si le favoris existe
$res = checkFavExist($fav);

//Le favoris n'existe pas
if($res === false){
    addFav($fav);
    $res = true;
}else{
    delFav($fav);
    $res = false;
}

//JSON
header('Content-type: application/json');
echo json_encode($res);
?>
```