

## AJAX, JQUERY, Base de données et PHP (corrigé)

La base de données « exempleAjax » comporte une table nommée « user », qui possède les caractéristiques suivantes :

exempleAjax.user	
id	: int(11)
nom	: varchar(25)
prenom	: varchar(25)
adresse	: text
login	: varchar(25)
mdp	: varchar(25)

id	nom	prenom	adresse	login	mdp
1	bidochon	robert	25 rue du gros rouge 75000 Paris	rbidochon	toto
2	galopin	raymonde	25 rue du gros rouge 75000 Paris	rgalopin	titi
3	einstein	albert	place e=mc2 44000 Nantes	aeinstein	tutu
4	asimov	isaac	56 boulevard de la fondation 69000 Lyon	iasimov	toto
5	toto	toto	toto	toto	toto

### Une donnée envoyée – une donnée reçue

1. Placez l'exemple dans un projet PHP que vous aurez créé dans votre répertoire public\_html et testez sa bonne fonctionnalité.
2. Modifiez l'exemple afin d'afficher le prénom de l'utilisateur sélectionné plutôt que son adresse.

liste des utilisateurs

25 rue du gros rouge  
75000 Paris

robert

```
<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>exemple ajax departement</title>
    <script src="js/jquery.js" type="text/javascript"></script>
    <script src="js/ajaxUser.js" type="text/javascript"></script>
  </head>
  <body>
    <div>
      <form>
        <label for="liste-user">liste des utilisateurs</label>
        <select name="liste-user" id="liste-user">

          </select>
        <br />
        <textarea name="adresse" id="adresse"></textarea>
        <textarea name="prenom" id="prenom"></textarea>
      </form>
    </div>
  </body>
</html>
```

```

/*
ajaxUser.js
*/

function selectionNom()
{
    var idUser = $(this).val(); // on récupère la valeur de l'option de la liste

    if (idUser != '-1') { // si l'utilisateur n'est pas le premier (choisissez....)
        // appel à la page majAdresse via ajax
        $.ajax({
            url: 'php/getAdresse.php',
            data: $(this).serialize(),
            type: 'GET',
            dataType: 'json',
            success: // si la requete fonctionne, mise à jour du champs adresse
                function (objetJson) {
                    $("#adresse").text(objetJson);
                },
            error:
                function (xhr, status, error) {
                    console.log("param : " + JSON.stringify(xhr));
                    console.log("status : " + status);
                    console.log("error : " + error);
                }
        });
    }
}

function selectionPrenom()
{
    var idUser = $(this).val(); // on récupère la valeur de l'option de la liste

    if (idUser != '-1') { // si l'utilisateur n'est pas le premier (choisissez....)
        // appel à la page majAdresse via ajax
        $.ajax({
            url: 'php/getPrenom.php',
            data: $(this).serialize(),
            type: 'GET',
            dataType: 'json',
            success: // si la requete fonctionne, mise à jour du champs adresse
                function (objetJson) {
                    $("#prenom").text(objetJson);
                },
            error:
                function (xhr, status, error) {
                    console.log("param : " + JSON.stringify(xhr));
                    console.log("status : " + status);
                    console.log("error : " + error);
                }
        });
    }
}

```

```

function genereListeUser()
{
    $.ajax({
        url: 'php/getUsers.php',
        type: 'GET',
        dataType: 'json',
        success: // si la requete fonctionne, mise à jour de la liste des personnes
            function (objetJson) {
                $("#liste-user").append('<option value="-1">Sélectionnez une personne</option>');
                $.each(objetJson, function (index, ligne) {
                    // ligne contient un objet json de la forme
                    // {"id" : "id de la personne"},
                    // {"nom" : "nom de la personne"}
                    $("#liste-user").append('<option value="' + ligne.id + '">' + ligne.nom + '</option>');
                });
            },
        error:
            function (xhr, status, error) {
                console.log("param : " + JSON.stringify(xhr));
                console.log("status : " + status);
                console.log("error : " + error);
            }
    });
}

$(document).ready(function ()
{
    // génération de la liste déroulante au chargement de la page
    genereListeUser();
    // gestion de la selection d'un nom
    $("#liste-user").change(selectionNom);
    $("#liste-user").change(selectionPrenom);
})

```

<?php

```

require_once 'config.inc';
/**
 * Retourne un objet PDO permettant l'accès aux tables de la base de données
 * @return \PDO
 */
function connexionBD() {
    try {

        $bdd = new PDO('mysql:host=' . SERVEURBD . ';dbname=' . NOMDELABASE, LOGIN, MOTDEPASSE);
    } catch (Exception $ex) {
        die('<br />Pb connexion serveur BD : ' . $ex->getMessage());
    }
    return $bdd;
}

```

```

/**
 * Retourne, au format json, l'adresse de la personne dont l'id est passée en parametre
 * @param type $id
 */
function getAdresse($id) {
    // connexion BD
    $bdd = connexionBD();

    $requete = $bdd->prepare("select adresse from user where id = :idU ;");
    $requete->bindParam(":idU", $id);
    $requete->execute() or die(print_r($requete->errorInfo()));
    $adresse = utf8_encode($requete->fetchColumn());
    $requete->closeCursor();
    //on previent qu'on repond en json
    header('Content-Type: application/json;charset=utf-8');
    // envoyer les données au format json
    echo json_encode($adresse);
}

function getPrenom($id) {
    // connexion BD
    $bdd = connexionBD();

    $requete = $bdd->prepare("select prenom from user where id = :idU ;");
    $requete->bindParam(":idU", $id);
    $requete->execute() or die(print_r($requete->errorInfo()));
    $prenom = utf8_encode($requete->fetchColumn());
    $requete->closeCursor();
    //on previent qu'on repond en json
    header('Content-Type: application/json;charset=utf-8');
    // envoyer les données au format json
    echo json_encode($prenom);
}

/**
 * Retourne, sous forme de tableau json, la liste des personnes (id,nom,prenom) de la table user
 */
function getListeUsers() {
    // connexion BD
    $bdd = connexionBD();
    // utilisation de prepare plutot que query pour avoir le gestion des erreurs
    $requete = $bdd->prepare("select id,nom from user order by nom;");

    $requete->execute() or die(print_r($requete->errorInfo()));

    // creation du tableau
    $tabUsers = array();

    while ($tab = $requete->fetch()) {
        // ajout d'une case dans le tableau
        // la case est elle-même un tableau contenant 2 champs : id, nom
        array_push($tabUsers, array('id' => $tab['id'], 'nom' => utf8_encode($tab['nom'])));
    }
    $requete->closeCursor();
    //on previent qu'on repond en json
    header('Content-Type: application/json;charset=utf-8');
    // envoyer les données au format json
    echo json_encode($tabUsers);}

```

```

<?php
require_once '../inc/fonctions.inc';

$Id = $_GET['liste-user'];
getPrenom($Id);

<?php

require_once '../inc/fonctions.inc';

$Id = $_GET['liste-user'];
getAdresse($Id);

<?php
require_once '../inc/fonctions.inc';
getListeUsers();

```

## N données envoyées – une donnée reçue

1. Placez l'exemple dans un projet PHP que vous aurez créé dans votre répertoire public\_html et testez sa bonne fonctionnalité.
2. Modifiez l'exemple afin que la pastille prenne les couleurs suivantes en fonction des scénarii.

| login   | mdp     | couleur pastille |
|---------|---------|------------------|
| bon     | bon     | vert             |
| bon     | pas bon | orange           |
| pas bon | pas bon | rouge            |

L'astuce consiste à ne pas rechercher le couple login/mdp, mais à récupérer le mot de passe correspondant au login via une requête. Ainsi, si le login n'existe pas, la requête retournera 0 ligne. Si le login existe, la requête retournera une ligne contenant le mot de passe, ce qui confirmera que le login existe. Il ne reste plus qu'à tester si le mot de passe retourné par la requête est le même que celui reçu depuis le formulaire, et tous les cas de figure auront été testés.

Login

mdp



```

<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>exemple ajax identification</title>
    <script src="js/jquery.js" type="text/javascript"></script>
    <script src="js/ajaxIdent.js" type="text/javascript"></script>
    <style>
      .pastilleGrise{
        width: 20px;
        height: 20px;
        background-color: lightgrey;
        border-radius: 10px;
      }
      .pastilleVerte{
        width: 20px;
        height: 20px;
        background-color: greenyellow;
        border-radius: 10px;
      }
      .pastilleRouge{
        width: 20px;
        height: 20px;
        background-color: red;
        border-radius: 10px;
      }
      .pastilleOrange{
        width: 20px;
        height: 20px;
        background-color: orange;
        border-radius: 10px;
      }
    </style>
  </head>
  <body>
    <div>
      <form id="formulaireLogin">
        <label for="login">Login</label>
        <input type="text" id="login" name="login" />
        <br />
        <label for="mdp">mdp</label>
        <input type="password" id="mdp" name="mdp" />
        <br />
        <input type="button" value="identification" id="verifLogin"/>
        <br />
        <p id="pastille" class="pastilleGrise"> </p>
        <p id="bidon"></p>
      </form>
    </div>
  </body>
</html>

```

```

function verifierAuthentification()
{
    // appel du script verifLogin.php via ajax
    $.ajax({
        url: 'php/verifLogin.php',
        data: $("#formulaireLogin").serialize(), // serialisation des données du formulaire (toutes les balises ayant un
champs "name"
                                //
        type: 'POST',
        dataType: 'json',
        success: // si la requete fonctionne, mise à jour de la couleur de pastille
            function(objetJson) {
                $("#pastille").removeClass();
                switch (objetJson){
                    case 'v': $("#pastille").toggleClass("pastilleVerte"); break;
                    case 'r': $("#pastille").toggleClass("pastilleRouge"); break;
                    case 'o': $("#pastille").toggleClass("pastilleOrange"); break;
                }
            },
        error:
            function (xhr, status, error) {
                console.log("param : " + JSON.stringify(xhr));
                console.log("status : " + status);
                console.log("error : " + error);
            }
    });
}

$(document).ready(function ()
{
    // gestion du click sur le bouton d'authentification
    $("#verifLogin").click(verifierAuthentification);
});

<?php
require_once './inc/fonctions.inc';

$log=$_POST['login'];
$mdp=$_POST['mdp'];

verifierLogin($log, $mdp);

```

```

<?php
require_once 'config.inc';

function connexionBD() {
    try {

        $bdd = new PDO('mysql:host=' . SERVEURBD . ';dbname=' . NOMDELABASE, LOGIN, MOTDEPASSE);
    } catch (Exception $ex) {
        die('<br />Pb connexion serveur BD : ' . $ex->getMessage());
    }
    return $bdd;
}

function verifierLogin($log, $mdp) {
    $bdd = connexionBD();
    // recherche du couple login/mdp dans la table user
    //$requete = $bdd->prepare("select id from user where login = :log and mdp = :mdp ;");
    $requete = $bdd->prepare("select mdp from user where login = :log;");
    $requete->bindParam(":log", $log);
    //$requete->bindParam(":mdp", $mdp);
    $requete->execute() or die(print_r($requete->errorInfo()));
    // comptage du nombre de resultats
    $nbLigne = $requete->rowCount();

    /*
    if ($nbLigne == 0) { // le couple login/mdp n'est pas présent dans la table user
        // il faudra retourner 'r'
        $retour = 'r';
    } else { // le couple login/mdp est présent dans la table user
        // il faudra retourner 'v'
        $retour = 'v';
    }
    */

    if ($nbLigne == 0) { // le login n'est pas présent dans la table user
        // il faudra retourner 'r'
        $retour = 'r';
    } else { // le login est présent dans la table user
        if ($nbLigne == 1) { // il n'y a qu'un login correspondant

            if ($mdp == $requete->fetchColumn()) { // le mdp est correct
                $retour = 'v';
            } else { // login ok, mais mdp pas ok
                $retour = 'o';
            }
        } else { // plusieurs login correspondant
            $retour = 'o';
            while ($ligne = $requete->fetch()) {
                if ($ligne['mdp'] == $mdp) {
                    $retour = 'v';
                }
            }
        }
    }
}
$requete->closeCursor();

//on previent qu'on repond en json
header('Content-Type: application/json;charset=utf-8');
// envoyer les données au format json
echo json_encode($retour);
}

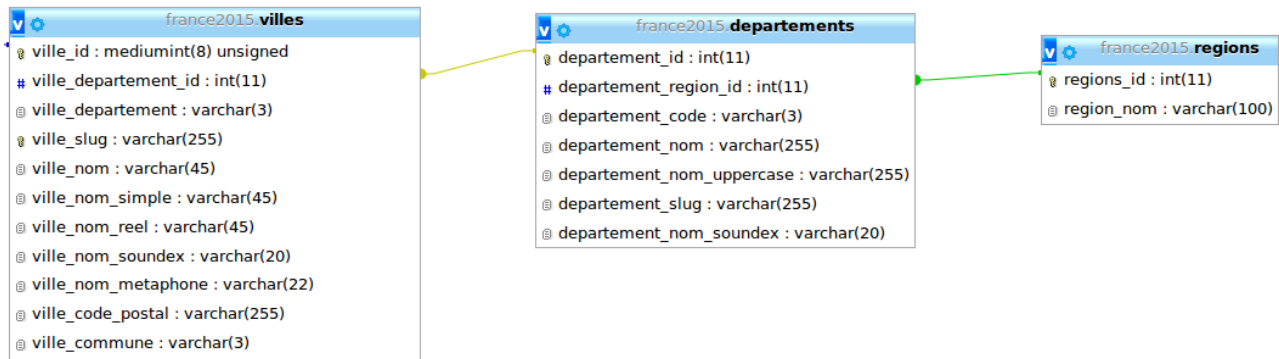
```



On fourni une application qui permet de mettre un point sur une google map en fonction d'une ville saisie dans un formulaire.

## Une donnée envoyée – N données reçus

Nous avons accès à la base de données « france2015 » contenant trois tables:



L'exemple fourni est fonctionnel et permet de choisir une région dans une liste déroulante, ce qui met à jour automatiquement une liste déroulante des départements de la région sélectionnée.

1. Placez l'exemple dans un projet PHP que vous aurez créé dans votre répertoire public\_html et testez sa bonne fonctionnalité.
2. Complétez le code afin que si un département est sélectionné la liste des villes se mette à jour.
3. Complétez votre code afin que toutes les interactions entre les listes soient cohérentes.  
Ex : La liste des villes redevient vide si une nouvelle région est sélectionnée.

Pays de la Loire ▼ Sarthe ▼ Choisissez une ville ▼

- Choisissez une ville
- AIGNE
- AILLIERES-BEAUVOIR
- ALLONNES
- AMNE
- ANCINNES
- ARCONNAY
- ARDENAY-SUR-MERIZE
- ARNAGE
- ARTHEZE**
- ASNIERES-SUR-VEGRE
- ASSE-LE-BOISNE
- ASSE-LE-RIBOUL
- AUBIGNE-RACAN
- AUVERS-LE-HAMON
- AUVERS-SOUS-MONTFAUCON
- AVESNES-EN-SAOSNOIS
- AVESSE
- AVEZE
- AVOISE

```

<!DOCTYPE html>
<html>
  <head>

    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>exemple ajax departement</title>
    <script src="js/jquery.js" type="text/javascript"></script>
    <script src="js/gestionFormulaire.js" type="text/javascript"></script>
  </head>
  <body>
    <form id="formulaire" method="post" action="">
      <select id="regions" name="idRegion">

        </select>
        <select id="departements" name="idDepartement">
          <option value="-1">Choisissez un departement</option>
        </select>

        <select id="villes" name="idVille">
          <option value="-1">Choisissez une ville</option>
        </select>
        <input type="hidden" name="typeDemande" id="typeDemande" value="r" />
      </form>
    </body>
  </html>

<?php

$source=$_GET['typeDemande'];
require_once '../inc/fonctions.inc';

switch ($source){
  case 'r':getListeRegions(); break;
  case 'd':getListeDepartementsFromIdRegion($_GET['idRegion']);break;
  case 'v':getListeVillesFromIdDepartement($_GET['idDepartement']);break;
}

function getListeVillesFromIdDepartement($idDept) {
  // connexion BD

  $bdd = connexionBD();

  $requete = $bdd->prepare("select ville_id,ville_nom from villes where ville_departement_id = :iddept order by ville_nom;");
  $requete->bindParam(":iddept", $idDept);
  $requete->execute() or die(print_r($requete->errorInfo()));

  $tabVille=array();

  while ($tab = $requete->fetch()) {
    array_push($tabVille,array('idVille'=>$tab['ville_id'], 'nomVille'=> utf8_encode($tab['ville_nom']));
  }

  $requete->closeCursor();
  //on previent qu'on repond en json
  header('Content-Type: application/json;charset=utf-8');
  // envoyer les données au format json
  echo json_encode($tabVille);

}

```

```

function genereListeVilles()
{
    var idDepartement = $(this).val(); // on récupère la valeur de la clef primaire correspondant au departement
    // changement de la valeur du type de demande pour faire appel à la fonction majListeDepartement
    $("#typeDemande").val('v');

    $("#villes").empty(); // on vide la liste des villes
    $("#villes").append('<option value="-1">Choisissez une ville </option>');

    if (idDepartement != '-1') { // si la region selectionné existe (pas le "choisissez une region")
        // appel à la page majListeDept via ajax

        $.ajax({
            url: 'php/controleur.php',
            data: $("#formulaire").serialize(), // on envoie toutes les données du formulaire (toutes les balises ayant un champs "name")
            // typeDemande='r' &
            // idDepartement=-1 &
            // idVille=-1 &
            // idRegion=idDeLaRegionSelectionnée
            type: 'GET',
            dataType: 'json',
            success: // si la requete fonctionne, mise à jour de la liste des departements

                function (objetJson) {

                    $.each(objetJson, function (index, ligne) {
                        // ligne contient un objet json de la forme
                        // {"idDepartement" : "id du departement"},
                        // {"nomDepartement" : "nom du departement"}
                        $("#villes").append('<option value="" + ligne.idVille + ">' + ligne.nomVille + '</option>');
                    });
                },
            error:
                function (xhr, status, error) {
                    console.log("param : " + JSON.stringify(xhr));
                    console.log("status : " + status);
                    console.log("error : " + error);
                }
        });
    }
}

$(document).ready(function ()
{
    // generation de la liste deroulante des regions
    genererListeRegions();

    // gestion du changement de region
    $("#regions").change(genereListeDepartements);

    // gestion du changement de departement
    $("#departements").change(genereListeVilles);

});

```

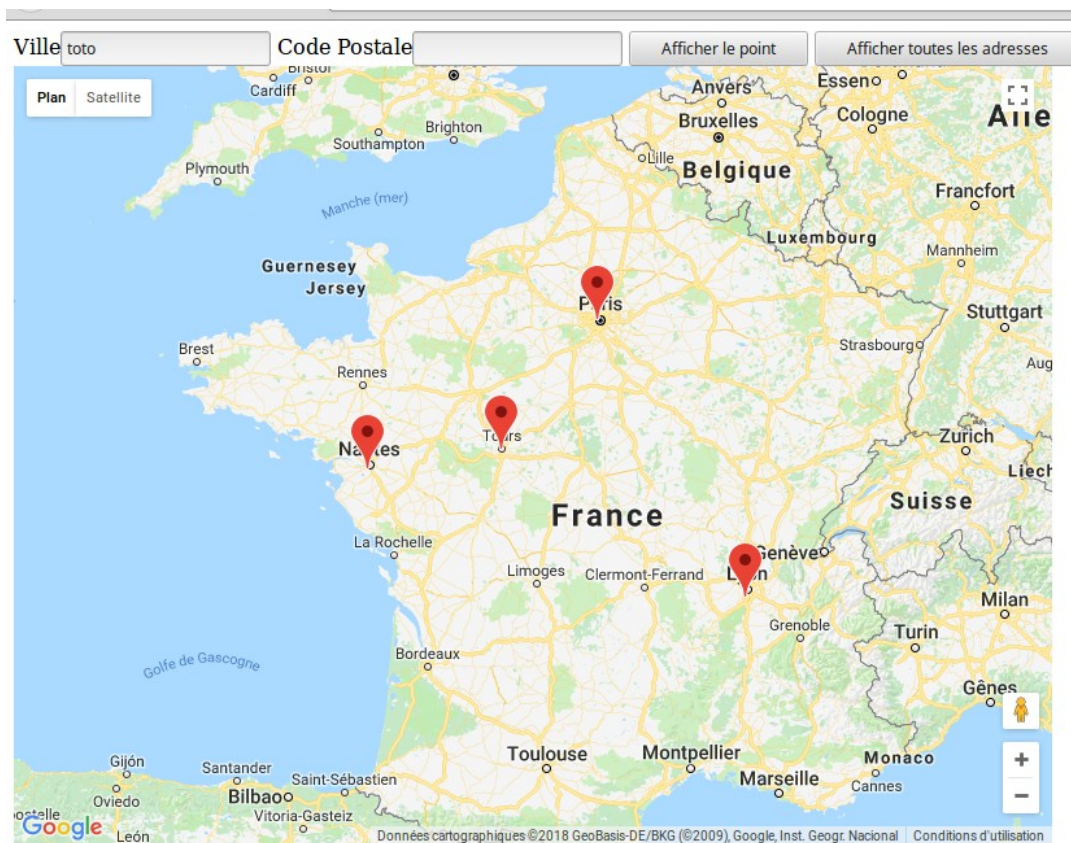
## Utilisation de l'api googleMap

On fourni une application qui permet de mettre un point sur une google map en fonction d'une ville saisie dans un formulaire.

Dans la base de données "exempleAjax" il y a la table "user" qui est structurée ainsi :

| exempleAjax user |             |
|------------------|-------------|
| id :             | int(11)     |
| nom :            | varchar(25) |
| prenom :         | varchar(25) |
| adresse :        | text        |
| login :          | varchar(25) |
| mdp :            | varchar(25) |

1. Modifiez l'exemple afin d'ajouter un bouton "toutes les adresses", qui lorsque l'on cliquera dessus, affichera une pastille pour chacune des adresses de la table user.



```

<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <!--<style type="text/css">
      html { height: 100% }
      body { height: 100%; margin: 0; padding: 0 }
      #map-canvas { height: 100% }
    </style-->

    <script type="text/javascript" src="https://maps.googleapis.com/maps/api/js?key= AIzaSyCG9EmLrvJN67QNqW-
FzyAplMz4dzq_ZGc">
    </script><!-- key=AIzaSyBcH73azCQnTLRojJ9tktk4Y_SnVWqk5Qo&amp; -->
    <script src="js/jquery.js" type="text/javascript"></script>
    <script src="js/apigooglemap.js" type="text/javascript"></script>
    <title>ma map</title>
  </head>
  <body>
    <form>
      Ville<input type="text" name="ville" id="ville"/>
      Code Postale<input type="text" name="cp" id="cp"/>
      <input type="button" value="Afficher le point" id="ajoutPoint">
      <input type="button" value="Afficher toutes les adresses" id="afficherTout" />

    </form>
    <div id="map-canvas" style="width: 800px;height: 600px;"></div>
  </body>
</html>

```

```

<?php
/*
define("SERVEURBD","172.18.58.15");
define("LOGIN","snir");
define("MOTDEPASSE","snir");
define("NOMDELABASE","exempleAjax");
*/

define ("SERVEURBD","127.0.0.1");
define ("LOGIN","root");
define ("MOTDEPASSE","raspberry");
define("NOMDELABASE","exempleAjax");

```

```

$(document).ready(function ()
{
    // définition et affichage de la carte au chargement de la page
    var mapOptions = {
        center: new google.maps.LatLng(46.779231, 2.659431),
        zoom: 8
    };
    map = new google.maps.Map(document.getElementById("map-canvas"),
        mapOptions);
    // gestion du click sur l'element ayant l'id "ajoutPoint"
    // appel à la fonction majCarte pour ajouter un point sur la carte
    $("#ajoutPoint").click(majCarte);
    $("#afficherTout").click(function()
    {
        $.ajax({
            url: 'php/listeVilles.php',
            type: 'GET',
            dataType: 'json',
            success: // si la requete fonctionne, lister les villes
                function(objetJson) {
                    $.each(objetJson, function(index, ligne) {
                        // ligne contient prenom et nomVille
                        $("#ville").val(ligne.nomVille);
                        majCarte();
                    });
                }

        });

    });

});

function connexionBD() {
    try {

        $bdd = new PDO('mysql:host=' . SERVEURBD . ';dbname=' . NOMDELABASE, LOGIN, MOTDEPASSE);
    } catch (Exception $ex) {
        die('<br />Pb connexion serveur BD : ' . $ex->getMessage());
    }
    return $bdd;
}

// connexion BD
$bdd = connexionBD();

$requete = $bdd->query("select prenom, adresse from user;");

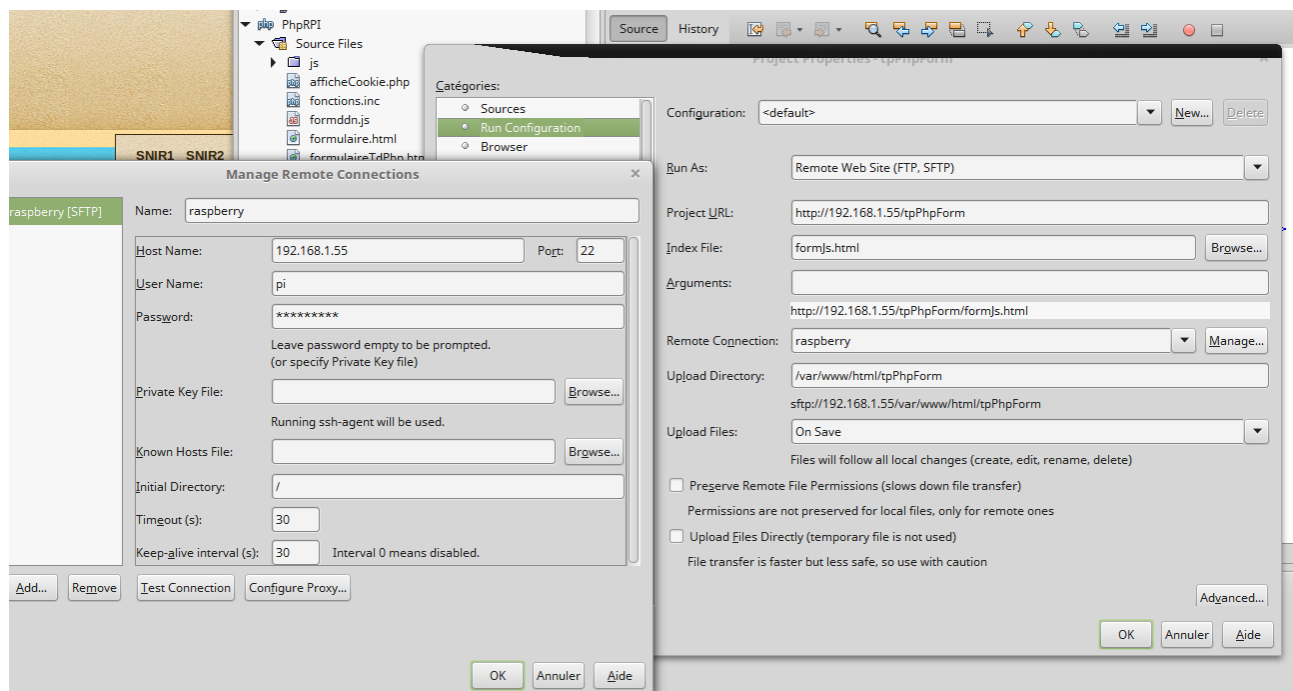
$tabAdr=array();

while ($tab = $requete->fetch()) {
    array_push($tabAdr,array('prenom'=>utf8_encode($tab['prenom']), 'nomVille'=> utf8_encode($tab['adresse'])));
}

$requete->closeCursor();
echo json_encode($tabAdr);

```

## Configuration Remote RaspberryPI



## Configuration local au lycée

