

JQuery 3.5.1



Bibliothèque de fonctions JavaScript
<https://jquery.com/>

JQuery c'est quoi ?



- C'est une bibliothèque JavaScript Open Source qui est sortie en 2006.
- Elle permet de parcourir et de manipuler l'arbre **DOM** d'une page Web très facilement.
- JQuery permet de changer/ajouter/supprimer une classe CSS, créer des animations, modifier des attributs, gérer des évènements, faire des requêtes AJAX simplement...



Comment intégrer JQuery à un projet Web ?

JQuery est une bibliothèque de fonctions **JavaScript** qui est cross-browser et qui permet de condenser et de simplifier la programmation en **JavaScript**.

Vous pouvez utiliser le script **jquery-3.5.1.js** directement dans votre projet

Ou bien

Vous pouvez utiliser un CDN (Content Delivery Network) un réseau de distribution de contenu.

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
```

```
<!DOCTYPE html>
<html lang="fr">

<head>
  <meta charset="UTF-8">
  <title>jQuery</title>
  <script src="js/jquery-3.4.1.js"></script>
  <script>
    $(document).ready(function() {
      // Code JQuery
    });
  </script>
</head>

<body>
  <h1>jQuery</h1>
</body>

</html>
```




Comment intégrer JQuery à un projet Web ?

Afin de respecter les bonnes pratiques de la programmation, vous devez inclure votre propre code **JQuery** dans le **head** de votre document.

Si vous voulez attendre le **chargement** de l'**arbre DOM** et le **contenu texte**, les **images** et les **vidéos** avant l'exécution de votre code **JQuery** vous devez utiliser l'instruction ci-contre ==>

Sinon, si seul le chargement de l'**arbre DOM** vous suffit vous pouvez vous limiter à :

```
<script src="js/jquery.min.js"></script>
<script>
    $(document).ready(function() {

        // Votre code JQuery

    });
</script>
</head>
```

```
<script src="js/jquery.min.js"></script>

<script>
    $(window).on("load",function () {

        // Votre Code JQuery

    });
</script>
</head>
```



Mise en Œuvre de JQuery

JQuery repose sur une seule fonction :

JQuery() ou bien **\$()** qui est plus largement utilisé

C'est une fonction **JavaScript** qui accepte des **paramètres** et qui retourne un **Objet JQuery** auquel on va pouvoir affecter une autre méthode JQuery et qui retournera à son tour un **Objet JQuery** ...

... → **Chaînage des méthodes**



Les Paramètres de la fonction \$()

- La fonction **\$()** accepte comme paramètre un **sélecteur**

Le sélecteur le plus simple est le nom de la **balise**

```
<p>Mon Premier Paragraphe</p>
<p>Mon Deuxième Paragraphe</p>
```

```
$("p").css("color", "red");
```

Mon Premier Paragraphe

Mon Deuxième Paragraphe

Le sélecteur « **#** » permet de designer un élément unique du DOM ayant un **id** donné

```
<p id="para1">Mon Paragraphe</p>
```

```
$("#para1").css("color", "green");
```

Mon Paragraphe

Le sélecteur « **.** » permet de designer les éléments du DOM ayant une **classe** donnée

```
<p class="maclasse">Mon Premier Paragraphe</p>
<p class="maclasse">Mon Deuxième Paragraphe</p>
<p class="maclasse">Mon Troisième Paragraphe</p>
```

```
$(".maclasse").css("color", "purple");
```

Mon Premier Paragraphe

Mon Deuxième Paragraphe

Mon Troisième Paragraphe



Sélectionner des éléments avec JQuery

Outre les sélecteur CSS, jQuery nous fournit quelques méthodes qui vont nous permettre **d'affiner** nos **sélections**.

On peut notamment citer les méthodes **has()**, **filter()**, **not()**, **first()**, **last()** et **eq()**.

```
<script src="js/jquery-3.4.1.js"></script>
<script>
  $(document).ready(function() {
    $("p").has("span").css("color", "orange");
  });
</script>
</head>

<body>
  <h1>Sélection d'éléments en jQuery</h1><br><br>
  <p>Un paragraphe <span>écrit en HTML</span></p>
  <span>Un élément span non contenu dans un élément p</span>
  <p>Un deuxième paragraphe</p>
  <p>Un <span>troisième</span> paragraphe</p>
</body>
```

Un paragraphe écrit en HTML

Un élément span non contenu dans un élément p

Un deuxième paragraphe

Un troisième paragraphe



Sélectionner des éléments avec JQuery

```
<script src="js/jquery-3.4.1.js"></script>
<script>
  $(document).ready(function() {

    $("span").first().css("color", "green");
    $("span").eq(1).css("color", "blue");
    $("span").last().css("color", "orange");

  });
</script>
</head>

<body>
  <h1>Sélection d'éléments en jQuery</h1><br><br>
  <p>Un paragraphe <span>écrit en HTML</span></p>
  <span>Un élément span non contenu dans un élément p</span>
  <p>Un deuxième paragraphe</p>
  <p>Un <span>troisième</span> paragraphe</p>
</body>
```

Sélection d'éléments en jQuery

Un paragraphe écrit en HTML

Un élément span non contenu dans un élément p

Un deuxième paragraphe

Un troisième paragraphe



La méthode CSS()

```
<body>
  <h1>La méthode css() de JQuery</h1>
  <br>
  <p class="maclasse">Mon Premier Paragraphe</p>
  <p class="maclasse">Mon Deuxième Paragraphe</p>
  <p class="maclasse">Mon Troisième Paragraphe</p>
</body>
```

```
<head>
  <meta charset="UTF-8">
  <title>JQuery</title>
  <link href="css/bootstrap.css" rel="stylesheet" />

  <script src="js/jquery-3.4.1.js"></script>
  <script>
    $(document).ready(function() {

      $(".maclasse").css({
        color: "purple",
        fontSize: "30px",
        border: "2px dashed green",
        width: "450px",
        margin: "20px"
      });

    });
  </script>
</head>
```

La méthode css() de JQuery

Mon Premier Paragraphe

Mon Deuxième Paragraphe

Mon Troisième Paragraphe



Les évènements en JQuery

JQuery permet de définir des gestionnaires d'évènement

Il existe une écriture simplifiée :

```
$("#btnSubmit").click(function() {  
    // Corps de la fonction  
});
```

Mais aussi une écriture plus longue et qu'il est préférable d'adopter :

```
$("#btnSubmit").on("click", function() {  
    // Corps de la fonction  
});
```



Les évènements en JQuery - La méthode on()

La méthode `on()` permet d'accrocher une écoute sur plusieurs évènements :

```
$("#btnSubmit").on("click dblclick", function() {  
    alert("Ce code fonctionne !");  
});
```

Elle permet aussi d'accrocher plusieurs fonctions de callback sur plusieurs évènements :

```
$("#btnSubmit").on({  
    click: function() {  
        alert("Vous avez cliqué !");  
    },  
    mouseup: function() {  
        alert("Vous avez relâché le clic !");  
    }  
});
```


Evènement et méthode css()

```
<style>
.petitcadre {
  height: 100px;
  width: 100px;
  float: left;
  margin: 20px;
  position: relative;
  padding: 20px;
  background: ■ #eeeeee;
  border: 1px solid ■ #bbbbbb;
  border-radius: 10px;
  -moz-border-radius: 10px;
  -webkit-border-radius: 10px;
  text-align: center;
}
</style>

<script src="js/jquery-3.4.1.js"></script>
<script>
  $(document).ready(function() {

    $("input[type=radio]").on("change", function() {

      var choixChecked = $(this).val();
      $(".petitcadre").css({
        backgroundColor: choixChecked,
        color: "white"
      });

    });

  });
</script>
</head>
```

Couleur

☐ Gris clair
☒ Pourpre
☐ Gris
☐ Indigo
☐ Orange

```
<div class="petitcadre">Couleur</div>

<input type="radio" name="monChoix" checked value="#eeeeee">Gris clair<br>
<input type="radio" name="monChoix" value="purple">Pourpre<br>
<input type="radio" name="monChoix" value="grey">Gris<br>
<input type="radio" name="monChoix" value="indigo">Indigo<br>
<input type="radio" name="monChoix" value="orange">Orange<br>
```



Naviguer dans le DOM

- Les méthodes **parent()**, **parents()**, **parentsUntil()** vont vous permettre d'accéder aux **parents** et **ancêtres** d'un élément ou d'une collection.
- Les méthodes **children()** et **find()** vont vous permettre d'accéder aux **descendants** et **enfants** d'un élément ou d'une collection.
- Les méthodes **siblings()**, **next()**, **nextAll()**, **nextUntil()**, **prev()**, **prevAll()** et **prevUntil()** vont vous permettre d'accéder aux éléments **frères** d'un élément ou d'une collection.

Naviguer dans le DOM : Exemple de Formulaire

Formulaire en JQuery

Nom :	<input type="text"/>	
Prénom :	<input type="text"/>	
Email :	<input type="text"/>	
<input type="button" value="Envoyer"/>	<input type="button" value="Annuler"/>	

```

<body>
  <h1>Formulaire en JQuery</h1>
  <br>
  <br>
  <form method="post" action="#">
    <div class="container">
      <div class="row">
        <table class="table table-bordered table-dark">
          <tr>
            <td><label for="nom">Nom : </label></td>
            <td><input type="text" id="nom" name="nom" class="champ" autocomplete="off"></td>
            <td class="tderreur"><span class="erreur">Vous avez des erreurs !!!</span></td>
          </tr>
          <tr>
            <td><label for="prenom">Prénom : </label></td>
            <td><input type="text" id="prenom" name="prenom" class="champ" autocomplete="off"></td>
            <td class="tderreur"><span class="erreur">Vous avez des erreurs !!!</span></td>
          </tr>
          <tr>
            <td><label for="email">Email : </label></td>
            <td><input type="text" id="email" name="email" class="champ" autocomplete="off"></td>
            <td class="tderreur"><span class="erreur">Vous avez des erreurs !!!</span></td>
          </tr>
          <tr>
            <td><input type="submit" class="btn btn-sample" id="envoyer" name="envoyer" value="Envoyer" /></td>
            <td><input type="reset" class="btn btn-sample" id="annuler" name="annuler" value="Annuler" /></td>
            <td></td>
          </tr>
        </table>
      </div>
    </div>
  </form>
</body>

```


Naviguer dans le DOM : Exemple de Formulaire

```
$(document).ready(function() {
    var patternMail = /^[a-z0-9\-\_\.]+\@[a-z0-9\-\_\.]+\.[a-z]{2,3}$/i;

    $(".erreur").hide();

    $("#envoyer").on("click", function(e) {

        if ($("#nom").val() == "") {
            $(".erreur").hide();
            e.preventDefault();
            let monErreur = $("#nom").parent().siblings(".tderreur").children(".erreur");
            monErreur.show();
            monErreur.html("<b>Vous devez rentrer un Nom !!!</b>");
        } else if ($("#prenom").val() == "") {
            $(".erreur").hide();
            e.preventDefault();
            let monErreur = $("#prenom").parent().siblings(".tderreur").children(".erreur");
            monErreur.show();
            monErreur.html("<b>Vous devez rentrer un Prénom !!!</b>");
        } else if (!$("#email").val().match(patternMail)) {

            $(".erreur").hide();
            e.preventDefault();
            let monErreur = $("#email").parent().siblings(".tderreur").children(".erreur");
            monErreur.show();
            monErreur.html("<b>Vous devez rentrer une Adresse Email Valide !!!</b>");
        } else {

            $(".erreur").hide();
            return true;
        }
    });
});
```





Naviguer dans le Dom et Modification des éléments

- Maintenant que vous savez naviguer dans le DOM, il va être intéressant de pouvoir **modifier** les valeurs des éléments auxquels vous accéder.
- Les méthodes essentielles pour y arriver sont :
html(), text(), val(), attr(), css() ...

Vous en avez d'ailleurs déjà rencontré certaines jusqu'ici ;-)



Naviguer dans le Dom et Insertion d'éléments

- Au-delà des méthodes **html()** et **text()** qui permettent aussi d'insérer du contenu dans l'arbre DOM.
- Il y a 2 autres méthodes essentielles qui permettent d'insérer du contenu ou des éléments dans l'arbre DOM : **prepend()**, **append()**

prepend() : insère du contenu en tant que **premier enfant** d'un élément cible.

append() : insère du contenu en tant que **dernier enfant** d'un élément cible.



Naviguer dans le Dom et Insertion d'éléments

```
<style>
  .conteneur {
    border: 6px solid ■rgb(112, 218, 117);
    margin: 20px;
    padding: 20px;
    width: 300px;
  }

  span {
    background-color: ■orangered;
    color: ■white;
    font-size: large;
  }
</style>
```

```
<script src="js/jquery-3.4.1.js"></script>
<script>
  $(document).ready(function() {

    $(".conteneur").prepend("<p>Paragraphe ajouté avec <span>prepend()</span></p>");
    $(".conteneur").append("<ul><li>n°1</li><li>n°2</li></ul> Liste ajoutée avec <span>append()</span>");

  });
</script>
```

```
<body>
  <div class="conteneur">
    <p><b>Un paragraphe écrit en HTML</b></p>
    <p><b>Un autre paragraphe écrit en HTML</b></p>
  </div>
</body>
```

Paragraphe ajouté avec **prepend()**

Un paragraphe écrit en HTML

Un autre paragraphe écrit en HTML

- n°1
- n°2

Liste ajoutée avec **append()**



Insertion de contenu avant ou après un élément

- Pour insérer du contenu **avant** ou **après** un **élément**, on va pouvoir utiliser l'une des méthodes jQuery **before()**, **after()**, **insertBefore()** ou **insertAfter()**.

```
<script src="js/jquery-3.4.1.js"></script>
<script>
  $(document).ready(function() {
    $("#p1").before("<p>Paragraphe ajouté avec before()</p>");
    $("<span>Span ajouté avec insertAfter()</span>").insertAfter("#p1").css("color", "green");
  });
</script>
</head>

<body>
  <h1>Sélection d'éléments en jQuery</h1><br><br>
  <p id="p1">Un paragraphe écrit en HTML</p>
  <p id="p2">Un autre paragraphe</p>
</body>
```

Paragraphe ajouté avec before()

Un paragraphe écrit en HTML

Span ajouté avec insertAfter()

Un autre paragraphe



Suppression d'éléments en JQuery

- Pour supprimer un élément ou un contenu du DOM, on va pouvoir utiliser les méthodes **remove()**, **detach()**, **empty()** ou **unwrap()**.

Suppression d'éléments en jQuery

Un paragraphe en dehors du div

```
<body>
  <h1>Suppression d'éléments en jQuery</h1><br><br>
  <div class="conteneur">
    <p id="p1">Un paragraphe écrit en HTML</p>
    <p id="p2">Un autre paragraphe</p>
  </div>
  <section>
    <p>Un paragraphe en dehors du div</p>
  </section>
</body>
```

```
<script src="js/jquery-3.4.1.js"></script>
<script>
  $(document).ready(function() {

    //Supprime le paragraphe avec l'id="p1"
    $("#p1").remove("#p1"); //Idem $("#p1").remove()

    //Supprime h1 et conserve ses données
    let titre = $("h1").detach();
    //Réinsère h1 au début du body
    titre.prependTo("body");

    //Supprime le contenu de #p2
    $("#p2").empty();

    //Supprime le parent des paragraphes si c'est un élément section
    $("p").unwrap("section");

  });
</script>
```




Manipulation des Attributs en JQuery

la méthode **attr()** va permettre de récupérer la valeur d'un **attribut**, **ajouter** un attribut ou **modifier** la valeur d'un attribut déjà existant.

Manipulation d'Attributs avec JQuery

Un paragraphe écrit en HTML

Un autre paragraphe

Un paragraphe en dehors du div

Un [lien](#) AFPA

Valeur du 1er id de p : p1

```
<body>
  <div class="conteneur">
    <h1>Manipulation d'Attributs avec JQuery</h1>
    <p id="p1">Un paragraphe écrit en HTML</p>
    <p id="p2">Un autre paragraphe</p>
  </div>
  <p>Un paragraphe en dehors du div</p>
  <p>Un <a href="https://www.afpa.fr/">lien</a> AFPA</p>
  <span id="valAttr"></span>
</body>
```

```
<script src="js/jquery-3.4.1.js"></script>
```

```
<script>
```

```
  $(document).ready(function() {
```

```
    //Retourne la valeur du premier id d'un paragraphe trouvé
```

```
    let pid = $("p").attr("id");
```

```
    //Affiche la valeur de l'id récupéré dans notre span id="valAttr"
```

```
    document.getElementById("valAttr").textContent = "Valeur du 1er id de p : " + pid;
```

```
    //Ajoute un attribut target à notre lien
```

```
    $("a").attr("target", "_blank");
```

```
  });
```

```
</script>
```



Manipulation de la Propriété d'un Attribut

La méthode **prop()** va vous permettre de récupérer de valeur de la propriété du premier élément de votre sélection.

```
<script src="js/jquery-3.4.1.js"></script>
<script>
  $(document).ready(function() {

    console.log("Valeur de la propriété : " + $("#check1").prop("checked"));
    console.log("Valeur de l'attribut : " + $("#check1").attr("checked"));

    $("#check1").on("click", function() {

      console.log("Valeur de la propriété : " + $(this).prop("checked"));
      console.log("Valeur de l'attribut : " + $("#check1").attr("checked"));

    });

  });
</script>
```

La Méthode prop()

☐ Cochez / décochez moi

Elements	Console	Source
top		
	Valeur de la propriété : true	
	Valeur de l'attribut : checked	
	Valeur de la propriété : false	
	Valeur de l'attribut : checked	
	>	

</head>

<body>

<h1>La Méthode prop()</h1>

<input id="check1" type="checkbox" checked="checked">

<label for="check1">Cochez / décochez moi</label>

</body>



Manipulation de l'Attribut **class**

- JQuery met à votre disposition une série de méthodes qui permet de manipuler l'attribut **class** : **hasClass()**, **addClass()**, **removeClass()**, **toggleClass()**

```
<body>
  <h1>L'Attribut class</h1><br>
  <button id="b1">Changer en bleu</button>
  <button id="b2">Toggle</button>
  <div class="afficher">
    <p>Un premier paragraphe</p>
    <p>Un deuxième paragraphe</p>
  </div>
  <p id="p3">Troisième paragraphe hors div</p>
</body>

</html>
```

```
<script src="js/jquery-3.4.1.js"></script>
<script>
  $(document).ready(function() {

    $("#b1").click(function() {
      //Si #p3 ne possède pas de class="bleu", on en ajoute une
      if (!$("#p3").hasClass("bleu")) {
        $("#p3").addClass("bleu");
      }
      $(".bleu").css("color", "blue");
    });

    $("#b2").click(function() {
      $(".div").toggleClass("cacher afficher");
      //On applique hide() aux éléments qui possèdent une class="cacher"
      $(".cacher").hide();
      //On applique show() aux éléments qui possèdent une class="afficher"
      $(".afficher").show();
    });

  });
</script>
```

Un premier paragraphe
 Un deuxième paragraphe
 Troisième paragraphe hors div

```

<script id="tinyhippos-injected">...</script>
<head>...</head>
<body>
  <h1>L'Attribut class</h1>
  <br>
  <button id="b1">Changer en bleu</button>
  <button id="b2">Toggle</button>
  ...
  <div class="afficher" style="" == $0
    <p>Un premier paragraphe</p>
    <p>Un deuxième paragraphe</p>
  </div>
  <p id="p3" class="bleu" style="color: blue;">Troisième paragraphe hors div</p>
</body>
</html>

```




AJAX avec JQuery

La méthode `$.ajax()` est au cœur des requêtes AJAX en JQuery

```
<script src="js/jquery-3.4.1.js"></script>
<script>
    $(document).ready(function() {
        $.ajax({
            url: 'pays.json',
            type: 'GET',
            dataType: 'json',
            // Paramètre success de retour qui est exécuté lorsque la requête Ajax est une réussite
            success: function(monjson, statut) {
                var html = "";
                monjson.forEach(element => {

                    html += "<li id='" + element.id + "'" + element.nom_fr_fr + "<br></li>";

                });
                $("#pays").show();
                $("#pays").empty();
                $("#pays").append(html);
            },
            // Paramètre error de retour qui est exécuté lorsqu'il y a une erreur
            error: function(resultat, statut, erreur) {

                erreur = alert("Il y a eu un problème avec votre requête Ajax !!!");
            },

            complete: function(resultat, statut) {
                console.log(resultat.responseText);
            }
        });
    });
</script>
```

```
<body>
    <h1>Requête AJAX et JQuery</h1><br>
    <ul name="pays" id="pays"></ul>
</body>
```

AJAX avec JQuery : Fichier JSON

JQuery > jqueryinit > {} pays.json > ...

```

1  [
2  {"id":"1","code":"4","alpha2":"AF","alpha3":"AFG","nom_en_gb":"Afghanistan","nom_fr_fr":"Afghanistan"},
3  {"id":"2","code":"8","alpha2":"AL","alpha3":"ALB","nom_en_gb":"Albania","nom_fr_fr":"Albanie"},
4  {"id":"3","code":"10","alpha2":"AQ","alpha3":"ATA","nom_en_gb":"Antarctica","nom_fr_fr":"Antarctique"},
5  {"id":"4","code":"12","alpha2":"DZ","alpha3":"DZA","nom_en_gb":"Algeria","nom_fr_fr":"Alg\u00e9rie"},
6  {"id":"5","code":"16","alpha2":"AS","alpha3":"ASM","nom_en_gb":"American Samoa","nom_fr_fr":"Samoa Am\u00e9ricaines"},
7  {"id":"6","code":"20","alpha2":"AD","alpha3":"AND","nom_en_gb":"Andorra","nom_fr_fr":"Andorre"},
8  {"id":"7","code":"24","alpha2":"AO","alpha3":"AGO","nom_en_gb":"Angola","nom_fr_fr":"Angola"},
9  {"id":"8","code":"28","alpha2":"AG","alpha3":"ATG","nom_en_gb":"Antigua and Barbuda","nom_fr_fr":"Antigua-et-Barbuda"},
10 {"id":"9","code":"31","alpha2":"AZ","alpha3":"AZE","nom_en_gb":"Azerbaijan","nom_fr_fr":"Azerba\u00efdjan"},
11 {"id":"10","code":"32","alpha2":"AR","alpha3":"ARG","nom_en_gb":"Argentina","nom_fr_fr":"Argentine"},
12 {"id":"11","code":"36","alpha2":"AU","alpha3":"AUS","nom_en_gb":"Australia","nom_fr_fr":"Australie"},
13 {"id":"12","code":"40","alpha2":"AT","alpha3":"AUT","nom_en_gb":"Austria","nom_fr_fr":"Autriche"},
14 {"id":"13","code":"44","alpha2":"BS","alpha3":"BHS","nom_en_gb":"Bahamas","nom_fr_fr":"Bahamas"},
15 {"id":"14","code":"48","alpha2":"BH","alpha3":"BHR","nom_en_gb":"Bahrain","nom_fr_fr":"Bahre\u00efn"},
16 {"id":"15","code":"50","alpha2":"BD","alpha3":"BGD","nom_en_gb":"Bangladesh","nom_fr_fr":"Bangladesh"},
17 {"id":"16","code":"51","alpha2":"AM","alpha3":"ARM","nom_en_gb":"Armenia","nom_fr_fr":"Arm\u00e9nie"},
18 {"id":"17","code":"52","alpha2":"BB","alpha3":"BRB","nom_en_gb":"Barbados","nom_fr_fr":"Barbade"},
19 {"id":"18","code":"56","alpha2":"BE","alpha3":"BEL","nom_en_gb":"Belgium","nom_fr_fr":"Belgique"},
20 {"id":"19","code":"60","alpha2":"BM","alpha3":"BMU","nom_en_gb":"Bermuda","nom_fr_fr":"Bermudes"},
21 {"id":"20","code":"64","alpha2":"BT","alpha3":"BTN","nom_en_gb":"Bhutan","nom_fr_fr":"Bhoutan"},
22 {"id":"21","code":"68","alpha2":"BO","alpha3":"BOL","nom_en_gb":"Bolivia","nom_fr_fr":"Bolivie"},

```



Les autres méthodes JQuery pour AJAX

- Les méthodes **\$.get()**, **\$.post()**, **\$.getJSON()** sont construites à partir de la méthode **\$.ajax()** mais elles sont moins flexibles et possèdent moins d'options de configuration.

\$.get() : permet d'effectuer une requête GET à l'URL fournie ;

\$.post() : permet d'effectuer une requête POST à l'URL fournie ;

\$.getJSON() : permet d'effectuer une requête GET à l'URL fournie et demande à ce que du JSON soit renvoyé en réponse.

Elles sont plus rapides à utiliser quand on ne s'occupe pas de la gestion des erreurs mais il est préférable d'utiliser la méthode **\$.ajax()**



Sérialisation de Formulaire

```
<script src="js/jquery-3.4.1.js"></script>
<script>
  $(document).ready(function() {

    $("button").on("click", function() {
      /*Transforme les données du formulaire en chaine de requête de la forme
      prenom=sacha&mail=sacha.%40gmail.com*/
      let chaine = $("form").serialize();

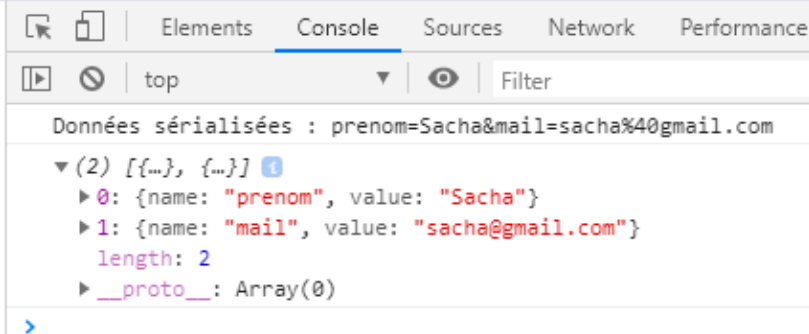
      //Transforme les données du formulaire en un tableau d'objets
      let tb = $("form").serializeArray();

      //Affiche les résultats dans la console
      console.log("Données sérialisées : " + chaine);
      console.log(tb);
    });
  });
</script>
```

Sérialisation de Formulaire

Prénom :

Mail :



Sérialisation de Formulaire

Sérialisation de Formulaire

Retour du serveur : sacha

Prénom :

Mail :

```
<?php
if(isset($_POST["prenom"]))
{
    $reponse = $_POST["prenom"];
    echo $reponse;
}
?>
```

```
<script>
$(document).ready(function() {

    $("#envoyer").on("click", function(e) {

        if ($("#prenom").val().length === 0) {
            $("#prenom").after("<span>Merci de remplir ce champ</span>");
            e.preventDefault();
        } else {
            $.ajax({
                url: 'traitement.php',
                type: 'POST',
                data: $("#form").serialize(),
                dataType: "html",
                success: function(reponse) {
                    $("#resultat").html("Retour du serveur : " + reponse);
                }
            });
        }
    });
});
</script>
```



Bibliographie

- <https://api.jquery.com/>



- <https://jqueryui.com/>



- <https://jquerymobile.com/>

