

# Framework JQuery - JSON

Introduction

Description

Les Méthodes

Exemple Index.html

Exemple interface JSON pour FW\_GRID

Exemple interface JSON pour FW\_DIALOG

Exemple interface JSON pour FW\_FORMATION

Exemple interface JSON pour FW\_TABS

Exemple interface JSON pour FW\_TREEVIEW

## Introduction :

Le principe est de pouvoir configurer au format **JSON (JavaScript Object Notation)** grâce a **JQuery** les principaux composants d'une **application web**, comme un **tableau**, des **onglets**, des **formulaires**...

**JQuery c'est quoi ?** C'est une bibliothèque JavaScript qui permet de développer des scripts rapidement et de manière concise. Elle intègre la gestion des événements, les animations et les interactions **Ajax**.

**JSON c'est quoi ?** C'est un format léger d'échange de données.

**Pourquoi JSON ?** Car il est facile à lire ou à écrire pour des humains. Il est aisément analysable ou générable par des machines. Il est basé sur un sous-ensemble du langage de programmation JavaScript. Ces propriétés font de JSON un langage d'échange de données idéal.

## Description :

Mise en place d'une sur-couche JQuery qui va permettre d'externaliser les données JSON en JavaScript. Le but est de créer une application web plus facilement et donc cela permet au développeur de gagner du temps, de plus le JSON facilite l'accès aux données.

Possibilité de créer une grid, un formulaires, des onglets, des messages box, des tree views a partir de données JSON.

## Les Méthodes:

|                    |                                  |   |
|--------------------|----------------------------------|---|
| FW_GRID(JSON)      | -> creer un tableau              | JSON = variable JSON qui contient<br>les descriptions du<br>composant |
| FW_TREEVIEW(JSON)  | -> creer un treeview             |   |
| FW_TABS(JSON)      | -> creer des onglets             |   |
| FW_FORMATION(JSON) | -> creer un formulaire           |   |
| FW_DIALOG(JSON)    | -> creer une fenetre de dialogue |   |

## Exemple Index.html :

```
/* Fichier Interface JSON */  
  
<script src="interface.js" type="text/javascript" charset="utf-8"></script>  
  
/*inclusion du FrameWork*/  
  
<script src="FWJQ/include.js" type="text/javascript" charset="utf-8"></script>  
  
/*Indiquer le chemin ou se situe le repertoire FWJQ*/  
  
<script> loadFrameworkJQuery("./"); </script>  
  
<body>  
  
    /*la fonction prend en parametre l'interface JSON*/  
    <script>FW_GRID(grid);</script>  
  
</body>
```

## Exemple interface JSON pour FW\_GRID :

```
var conf = { name:"Grid", width: 100 }  
var source = [  
    { name:"Number", type:"int" },  
    { name:"String", type:"string" },  
    { name:"Mail", type:"string" }  
]  
var theme = [  
    { name:"Mail", width:"50", align:"left", editable:true },  
    { name:"Number", width:"50", align:"left", editable:true },  
    { name:"String", width:"50", align:"left", editable:true }  
]
```

*Plus de details dans la documentation Grid.pdf*

## Exemple interface JSON pour FW\_DIALOG :

```
var conf = { name:"Dialog", title:"Demo Dialog", display:"This is the default dialog" }  
var dialog = { conf:conf }
```

*Plus de details dans la documentation Dialog.pdf*

## Exemple interface JSON pour FW\_FORMATION :

```
var conf =      {name:"Formation", width:"500", widthtaction:"send_form.php", method:"POST"}

var source = [
    {name:"nom", type:"string"},
    {name:"mail", type:"string"},
    {name:"age", type:"int"},
    {name:"fruit", type:"checkbox"},
    {name:"pays", type:"radio"},
    {name:"secure", type:"captcha"},
    {name:"description", type:"textarea"},
    {name:"selection", type:"select"},
    {name:"Submit", type:"button"},
    {name:"Reset", type:"button"}
]

var theme = [
    {name:"nom", display:"Nom :", required:true, defaultValue:"Entrez votre nom"},
    {name:"mail", display:"Email :", required:true, validation:"email"},
    {name:"age", display:"Age :", validation:"number", errorMessage:"Veuillez entrez
un nombre"},
    {name:"fruit", display:"Fruit :", legend:"Choisissez parmi ces fruit",
choices:{"apples":"Apples", "oranges":"Oranges", "grapes":"Grapes"}, required:true},
    {name:"pays", display:"Pays :", legend:"Choisissez un pays",
choices:["Mac", "Linux", "Windows", "Ubuntu"], required:true},
    {name:"secure", captchaQuestions:{'5 + 5 = ?': '10', 'What color is the
sky?': 'blue', '2 + 2 = ?': '4'}, required:true},
    {name:"description", display:"Decrivez vous en quelques mots", required:true,
cols:"60", rows:"10"},
    {name:"selection", display:"Do you like jQuery?", required:true,
choices:{'1':'Yes', '2':'No'}},
    {name:"Submit", display:"Submit"},
    {name:"Reset", display:"Reset", type:"reset"}
]

var formation = { conf:conf, source:source, theme:theme }
```

*Plus de details dans la documentation Formation.pdf*

## Exemple interface JSON pour FW\_TABS :

```
var conf = {name: "tabs", collapsible: true, event: 'mouseover', pinner: 'Chargement en cours...'}

var source = [
  {name:"onglet1", content:"content1"},
  {name:"onglet2", content:"content2"},
  {name:"onglet3", content:"content3"}
]

var theme = [
  {name:"onglet1", display:"Description onglet1", style:"text"},
  {name:"onglet2", display:"Description onglet2", style:"code"},
  {name:"onglet3", display:"Description onglet3", style:"text"}
]

var tabs = {conf:conf, source:source, theme:theme}
```

*Plus de details dans la documentation Tabs.pdf*

## Exemple interface JSON pour FW\_TREEVIEW :

```
var conf = {name:"treeview", width: 100}

var source = [
  {name:"Folder1", type:"folder", subfolder: [{name:"Folder1.1", type:"folder", subfolder:
    [{name:"Folder1.1.1", type:"file"}], {name:"Folder1.2", type:"folder"}]},
  {name:"Folder2", type:"folder"},
  {name:"Folder3", type:"folder"},
  {name:"File1", type:"file"}
]

var theme = [
  {name:"Folder1", state:"closed", items:[{name:"Item1"}, {name:"Item1.2"}]},
  {name:"Folder1.1", state:"closed", items:[{name:"Item1.1.1"}]},
  {name:"Folder1.1.1", state:"opened"},
  {name:"Folder1.2", state:"opened"},
  {name:"Folder2", state:"closed", items:[{name:"Item2.1"}, {name:"Item2.2"}]},
  {name:"Folder3", state:"closed", items:[{name:"Item3.1"}, {name:"Item3.2"}]}
]

var treeview = {conf:conf, source:source, theme:theme}
```

*Plus de details dans la documentation Treeview.pdf*