

Javascript

jQuery - Ajax - APIs

Javascript

Le Javascript accompagne les sites web depuis toujours. C'est un langage de programmation intégré dans le navigateur et exécuté depuis l'ordinateur de l'internaute, donc dépendant de sa puissance.

Longtemps considéré comme lourd et obscur, aujourd'hui le javascript connaît une explosion et des dizaines de plateformes voient le jour régulièrement, comme Angular JS de Google, pour mettre à disposition des outils afin de réaliser des applications web capable de répondre en temps réel.

Cours Javascript

<http://fr.openclassrooms.com/informatique/javascript/cours>

Les APIs

Une API est une interface de programmation qu'un site fourni afin de profiter de ses services depuis votre site.

Définition

http://en.wikipedia.org/wiki/Application_programming_interface

Par exemple, on peut utiliser l'API Youtube pour insérer un lecteur vidéo personnalisé sur notre site, ou encore l'API Flickr pour récupérer des photos, l'API Google Maps pour afficher une carte et dessiner dessus. l'API de facebook permet de lier vos infos à une application, pour une connexion plus rapide par exemple ou pour récupérer vos contacts.

Beaucoup de web service proposent une API, ce qui permet de créer des ponts entre les applications.

API Youtube

La documentation se trouve à cette adresse :

https://developers.google.com/youtube/iframe_api_reference

Code pour intégrer un lecteur vidéo

```

<!DOCTYPE html>
<html>
  <body>
    <!-- 1. The <iframe> (and video player) will replace this <div> tag. -->
    <div id="player"></div>

    <script>
      // 2. This code loads the IFrame Player API code asynchronously.
      var tag = document.createElement('script');

      tag.src = "https://www.youtube.com/iframe_api";
      var firstScriptTag = document.getElementsByTagName('script')[0];
      firstScriptTag.parentNode.insertBefore(tag, firstScriptTag);

      // 3. This function creates an <iframe> (and YouTube player)
      //    after the API code downloads.
      var player;
      function onYouTubeIframeAPIReady() {
        player = new YT.Player('player', {
          height: '390',
          width: '640',
          videoId: 'M7lc1UVf-VE',
          events: {
            'onReady': onPlayerReady,
            'onStateChange': onPlayerStateChange
          }
        });
      }

      // 4. The API will call this function when the video player is ready.
      function onPlayerReady(event) {
        event.target.playVideo();
      }

      // 5. The API calls this function when the player's state changes.
      //    The function indicates that when playing a video (state=1),

```

```

// the player should play for six seconds and then stop.
var done = false;
function onPlayerStateChange(event) {
    if (event.data == YT.PlayerState.PLAYING && !done) {
        setTimeout(stopVideo, 6000);
        done = true;
    }
}
function stopVideo() {
    player.stopVideo();
}
</script>
</body>
</html>

```

Exemple :

<http://codepen.io/smoothie-creative/pen/5f5de7d55197b0ab73648dd3bab91e0a>

Piloter le lecteur : Chapitres

Il est ensuite possible de piloter cette vidéo, ajouter des chapitres...

Voir exemple sur www.objectif-habitat.com

Exemple

<http://codepen.io/smoothie-creative/pen/922c78e2c2fe90feb18084ee87ca2d6d?editors=101>

On utilise donc les fonctions fournies par l'API pour faire ce que l'on souhaite.

API Google Maps

L'API Gmaps est très puissante, on va pouvoir créer une carte, choisir le fond, dessiner des formes, géolocaliser, convertir une adresse en coordonnées, afficher des punaises, les regrouper, zoomer, afficher des infos, streetview, etc ...

Exemple d'utilisation conjointe du champ Suggestion Google Maps et d'une Gmap

www.leboncoequipier.fr

Api Gmaps

<https://developers.google.com/maps/>

Exemple de map Google plein écran

<http://codepen.io/smoothie-creative/pen/c2c7f86ed2279556f112c75a1bb8dceb>

Exemple dans un projet Web réel, avec géolocalisation, punaises personnalisées et secteurs :

<http://citelib.com/cite-plan/>

Ajouter un marqueur sur la carte

<https://developers.google.com/maps/documentation/javascript/markers>

Le Géocoding

Le géocoding permet de transformer une adresse en coordonnées GPS

<https://developers.google.com/maps/documentation/javascript/geocoding>

Places Autocomplete

Permet d'ajouter un moteur de suggestion à un champ HTML, pour proposer des lieux, comme sur Google Maps

<https://developers.google.com/maps/documentation/javascript/places-autocomplete>

Géolocalisation

On peut coupler avec la technologie HTML5 de géolocalisation pour afficher votre emplacement sur la carte.

<https://developers.google.com/maps/documentation/javascript/examples/map-geolocation?hl=fr>

Utiliser une bibliothèque javascript dans son projet Web

Au lieu de réinventer la roue à chaque fois, on peut utiliser des milliers de plugins jQuery et javascript conçus en open source par la communauté. Un gain de temps assuré et une meilleure ergonomie.

Select2

Le Champ select2 est très pratique, il permet, comme avec le champ select de HTML, de proposer des choix, mais de manière plus intelligente. C'est aussi un fallback du champ search HTML5.

Le Plugin, exemples et codes :

<http://ivaynberg.github.io/select2/>

Multi selection, suggestion, choix sous forme de tags, recherche, Chargement dynamique ...

jQuery

Parce que Javascript est assez complexe à comprendre, et à écrire. Une bibliothèque permettant de simplifier a vu le jour, c'est jQuery.

Largement utilisée dans la plupart des projets, vous devrez l'intégrer presque à coup sur dans vos projets pour faire fonctionner vos plugins javascript.

jQuery offre une meilleure ergonomie d'écriture. Son avantage majeur est de proposer une fonction pour facilement sélectionner des éléments du DOM, comme on le fait avec CSS.

jQuery

<http://jquery.com/>

Documentation

<http://api.jquery.com/>

Exemples

<http://codepen.io/smoothie-creative/pen/7401359a9c090f1d4de65a3b0cdc9359>

Méthodes

jQuery possède également plein de méthodes (=fonctions) pour créer des animations, changer l'état d'un élément, modifier le contenu ...

En voici quelques unes

<http://codepen.io/smoothie-creative/pen/df8dd26fb4d1d3dc0910d11ec8afd408>

Événements

jQuery permet de prendre en charge tous les événements possibles :

- Clic de la souris
- Double clic
- Survol d'une zone
- Appui sur une touche du clavier
- Entrée dans un champ HTML
- Modification d'un champ
- Sortie d'un champ

Quelques événements en action

<http://codepen.io/smoothie-creative/pen/3c7c1c81a3a67999bb5ca81f5eb7ec35>

Ressources

Un bon récapitulatif jQuery :

<http://oscarotero.com/jquery/>

Ajax

Ajax permet d'envoyer une requête au serveur, pour mettre à jour une information par exemple sans demander à recharger complètement la page.

web standard avec http : la page demande une URL, le serveur renvoie une nouvelle page;

Avec Ajax on peut demander une mise à jour d'une donnée et l'appliquer dans le DOM en js sans avoir à tout recharger.

Exemple : Facebook

Types de requêtes Post et Get

<http://api.jquery.com/jquery.post/>

<http://api.jquery.com/jquery.get/>

L'ajax permet donc un web beaucoup plus dynamique et temps réel. Et s'avère très puissant.

Aller plus loin

Il existe aujourd'hui énormément de plateformes front-end (javascript) permettant de gérer des pages en direct.

Les plateformes les plus imposantes ne sont pas codées main, elles utilisent des frameworks puissants capable de faire gagner énormément de temps.

Selon vos besoins, vous pouvez utiliser les plateformes suivantes :

- Angular JS de Google <https://angularjs.org/>
- Meteor JS <https://www.meteor.com/>
- Backbone JS <http://documentcloud.github.io/backbone/>
- Node JS <http://nodejs.org/>
- React de Facebook <http://facebook.github.io/react/>
- Ember JS <http://emberjs.com/>
- ...