



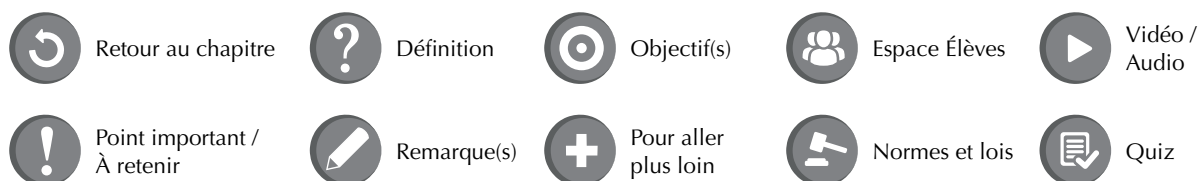
# Savoir utiliser le debugger JavaScript

# Sommaire

Introduction.....	3
I. Lancement du debugger.....	4
II. L'objet window.....	6
A. Propriétés et méthodes de l'objet window .....	6

Crédits des illustrations :  
© DR, Fotolia.

## Les repères de lecture



# Introduction

Le debugger JavaScript sera votre outil principal pour la mise au point de vos scripts. Nous l'utiliserons également pour votre apprentissage, aussi nous avons décidé qu'il ferait l'objet de la première leçon.

Chaque navigateur fournit son propre debugger JavaScript. Parmi les plus conviviaux, nous avons décidé d'utiliser, pour ce manuel, celui du navigateur Chrome. Lorsque vous serez formé à son utilisation, les autres vous paraîtront simples à apprendre !



# I. Lancement du debugger

Pour lancer le debugger :

1. Ouvrez le navigateur Chrome et entrez l'URL <https://www.facebook.com/> (nous utilisons ici le site de Facebook pour exemple) ;
2. Faites un clic droit sur le logo et choisissez le menu « Procédez à l'inspection de l'élément ».

Ceci fait, voici comment doit apparaître la fenêtre de votre navigateur :

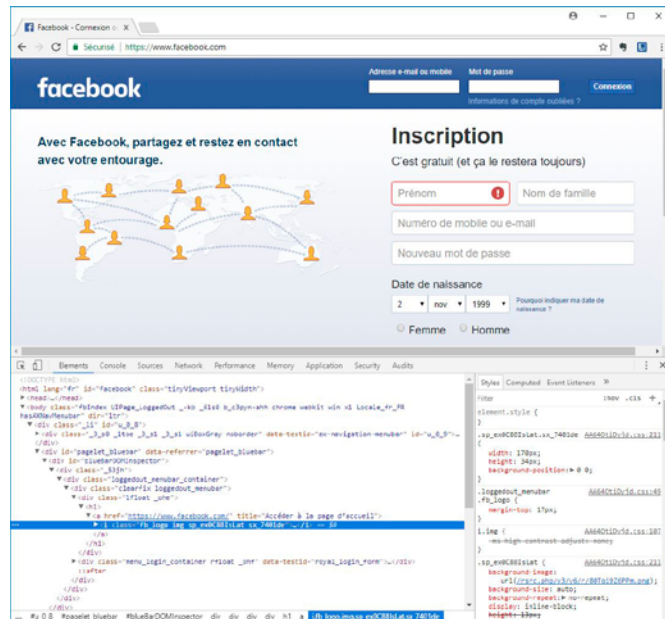


Fig. 1

Dans la fenêtre du debugger, en bas, à droite, est affiché le code de la page HTML avec une ligne sur fond gris, c'est l'élément HTML que vous avez sélectionné pour inspection :

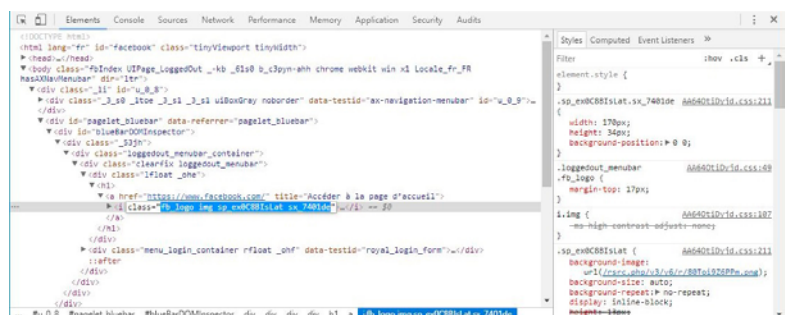


Fig. 2

La valeur des attributs peut être modifiable, il suffit de faire un clic droit sur un attribut (« class » par exemple) et de sélectionner le menu « Edit Attribute ». Vous pouvez essayer.

**Dans le panneau de droite**, vous pouvez visualiser les styles CSS appliqués à l'élément sélectionné. Les valeurs des différentes propriétés sont modifiables par simple clic (très pratique pour mettre au point vos feuilles de styles) :

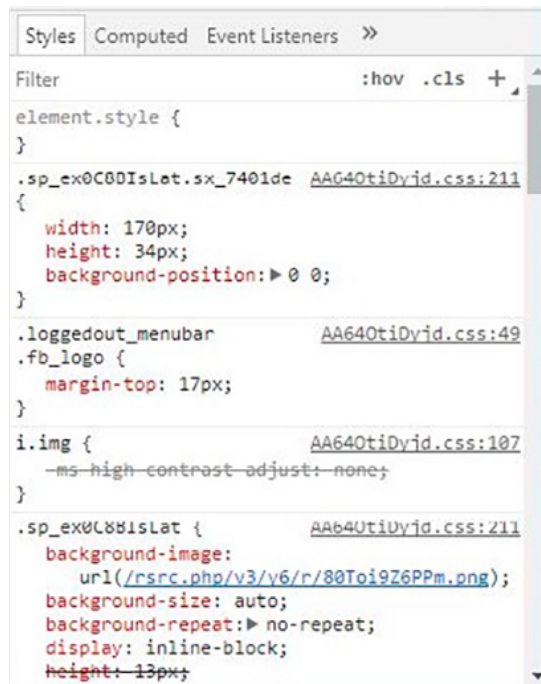


Fig.3

**En haut de la fenêtre du debugger**, un menu vous permet de fermer le debugger (la petite croix à droite) et d'accéder aux autres outils du debugger ; l'outil sélectionné actuellement est « Elements » :



Fig.4

Maintenant, vous pouvez jouer un moment avec le debugger, modifier les attributs, les textes, les couleurs.

Nous verrons, dans les leçons suivantes, à l'aide de cas concrets, comment utiliser le debugger pour mettre au point vos scripts.

## II. L'objet window

Lorsque le navigateur ouvre une nouvelle fenêtre, un *iframe* ou un onglet, il crée un objet *window*, cet objet représente la fenêtre d'affichage du navigateur.

### A. Propriétés et méthodes de l'objet window

L'objet window fournit des méthodes et des propriétés permettant d'obtenir des informations concernant la fenêtre et de les manipuler.

Comme pour tout objet JavaScript, la syntaxe générale pour accéder aux propriétés ou aux méthodes de l'objet window suit la syntaxe classique pour les objets :

objet. methode ou objet. propriété

Si le nom de la méthode contient un caractère pouvant être compris par JavaScript comme une opération (le tiret par exemple), vous devez utiliser la syntaxe alternative entre crochet.

#### Exemple :

objet ["propriete-avec-tiret"]

*Vous trouverez, ci-après, la liste des propriétés et méthodes de l'objet window. Vous devez savoir qu'elles existent. Vous trouverez par ailleurs, dans la leçon "Documentation", une liste de site fournissant une documentation détaillée de chaque propriété et de chaque méthode.*

Tableau n°1

PROPRIÉTÉ	DESCRIPTION
closed	Retourne une valeur booléenne indiquant si la fenêtre a été fermée ou non.
defaultStatus	Retourne ou modifie le texte par défaut affiché dans la barre d'états de la fenêtre.
document	Retourne l'objet représentant le document HTML chargé dans la fenêtre (abordé dans leçon suivante).
frames	Retourne un tableau de tous les frames et iframe de la fenêtre.
history	Retourne l'objet history (historique de navigation) de la fenêtre.
innerHeight	Retourne ou modifie la hauteur de la zone d'affichage de la fenêtre.
innerWidth	Retourne ou modifie la largeur de la zone d'affichage de la fenêtre.
length	Retourne le nombre de frame et d'iframe de la fenêtre.
location	Retourne l'objet location de la fenêtre (voir plus loin).
name	Retourne ou modifie le nom de la fenêtre.
navigator	Retourne l'objet navigator de la fenêtre qui permet d'accéder aux propriétés et méthodes du navigateur.
opener	Retourne une référence à l'objet window qui a créé la fenêtre.
outerHeight	Retourne ou modifie la hauteur de la fenêtre du navigateur.
outerWidth	Retourne ou modifie la largeur de la fenêtre du navigateur.
pageXOffset	Retourne de combien de pixel le document HTML a été paginé horizontalement dans la fenêtre.

PROPRIÉTÉ	DESCRIPTION
pageYOffset	Retourne de combien de pixel le document HTML a été paginé verticalement dans la fenêtre.
parent	Retourne l'objet window correspondant à la fenêtre parente de la fenêtre actuelle.
screen	Retourne l'objet screen (écran) de la fenêtre.
screenLeft	Retourne la coordonnée horizontale de la fenêtre relative à l'écran.
screenTop	Retourne la coordonnée verticale de la fenêtre relative à l'écran.
screenX	Retourne la coordonnée horizontale de la fenêtre relative à l'écran.
screenY	Retourne la coordonnée verticale de la fenêtre relative à l'écran.
self	Retourne l'objet window courant.
status	Modifie le texte de la barre d'états de la fenêtre.
top	Retourne l'objet window correspondant à la plus haute fenêtre du navigateur.

Tableau n°2

MÉTHODE	DESCRIPTION
alert()	Affiche une boîte d'alerte avec un message et un bouton « OK ».
blur()	Supprime le focus sur la fenêtre actuelle.
clearInterval()	Supprime une action différée à l'aide de la méthode setInterval().
clearTimeout()	Supprime une action différée à l'aide de la méthode setTimeout().
close()	Ferme la fenêtre actuelle.
confirm()	Affiche une boîte de dialogue avec un message, un bouton « OK » et un bouton « Annuler ».
createPopup()	Crée une fenêtre popup.
focus()	Positionne le focus sur la fenêtre courante.
moveBy()	Déplace une fenêtre par rapport à sa position actuelle.
moveTo()	Déplace une fenêtre à la position spécifiée.
open()	Ouvre une fenêtre de navigateur.
print()	Imprime le contenu de la fenêtre actuelle.
prompt()	Affiche une boîte de dialogue où l'utilisateur pourra entrer du texte.
resizeBy(x, y)	Agrandi ou diminue la taille de la fenêtre, selon le nombre de pixels spécifiés.
resizeTo(x, y)	Modifie la taille de la fenêtre selon la largeur et la hauteur spécifiée (en pixels).
scrollBy(x, y)	Page le contenu de la fenêtre selon le nombre de pixel spécifié par rapport à la position actuelle
scrollTo(x, y)	Page le contenu de la fenêtre par rapport au début du document.
setInterval()	Appelle périodiquement une fonction ou évalue périodiquement une expression selon un intervalle de temps spécifié en millisecondes.
setTimeout()	Diffère l'appel d'une fonction ou l'évaluation d'une expression en spécifiant le nombre de millisecondes.

Tous les objets JavaScript globaux, les variables ou les fonctions deviennent automatiquement membres de l'objet *window*. Ainsi, les propriétés globales sont des propriétés de l'objet *window*, les méthodes, des méthodes de l'objet *window*.

Le nom complet d'un objet global est de la forme *window. objet*. Cependant, s'agissant d'un objet global, vous pouvez utiliser une syntaxe simplifiée en utilisant uniquement le nom de l'objet.

**Exemples:**

*window.alert()* est équivalent à *alert()* ;

*window.innerHeigh* est équivalent à *innerHeight* ;

*window.document* est équivalent à *document*

Les objets *navigator*, *window*, *screen*, *location* contiennent des informations concernant le navigateur, sa fenêtre d'affichage, l'écran et des informations de localisation.

L'objet *window* est l'objet auquel se rattache toutes les variables, propriétés ou méthodes globales.

L'objet *document*, qui représente le document HTML chargé dans la fenêtre du navigateur, sera décrit dans la prochaine leçon.