JavaScript

* Le problème de JavaScript c’est la comptabilité dans les différents navigateurs, car on peut trouver qu’un code JavaScript marche sur un navigateur et ne marche pas dans d’autres, mais il Ya plusieurs progrès dans le monde d’informatique pour résoudre ce problème.
* JavaScript est utilisé par les outils de mesure d’audience, grâce à JavaScript, on peut obtenir des informations sur l’origine de la visite d’un site web, durée, fidélité, les connexions utilisés,
* JavaScript est un langage de script, donc, le script est le code source du langage
* JavaScript est interprété, c’est-à-dire, lu, analysé et exécuté directement depuis le code source par le navigateur sans aucune transformation. Il n’y pas la phase de compilation.
* JavaScript est exécuté coté client (langage client), le client ici c’est l’utilisateur. Donc, l’exécution de JavaScript se fait au niveau du navigateur.
* Le défaut de JavaScript, c’est que puisque JavaScript est un langage interprété, le code source est nécessaire pour l’exécution.

L’autre d’défaut, c’est que techniquement, c’est impossible de cacher le code source JavaScript d’un site web

* Les limites de JavaScript, c’est qu’on ne peut pas accéder seulement avec JavaScript directement à un serveur, par exemple un serveur de base de données pour enregistrer et lire des données, donc, on passe par un langage serveur.
* JavaScript ne peut pas accéder aux fichiers stockés dans un poste client, ou de consulter les mots de passe, mail, ect.
* JavaScript n’autorise pas l’installation automatique de programmes exécutables, ou des logiciels, mais il Ya des fonctionnalités dans des navigateurs pour faire ça, comme les composants ActiveX ou des applets JAVA, qui peuvent être intégré d’une façon transparente dans le navigateur web, et installer à votre issu des programmes dans votre machine.
* Contraction des mots « phreaking » « piratage » et « fishing » : ces principes consiste à récupérer des informations personnelles sans l’accord des internautes, fait partie des techniques de piratage.
* Programmation et tests sont deux éléments indissociables. Par exemple pour le développement d’u site web, il faut le tester dans plusieurs navigateurs, notamment, les anciennes et les récentes versions. Egalement, i faut le tester sur plusieurs environnement (Windows, linux, mac os), par exemple, le comportement d’internet explorer en linux n’est pas pareil que celui de Windows. Egalement, tester le site dans Android, iPhone, etc.
* Vu que c’est impossible de faire tous les tests, il faut faire les tests surtout sur les navigateurs les plus courants, comme IE, Firefox ou chrome, et de viser les visiteurs potentiels qui peuvent utiliser le site à travers des outils d’audiences sur les utilisateurs d’un site internet.
* Il ya des sites d’audience gratuit comme « alexa », « google analytics » ou « google trends »
* Auto-remplissage et auto-complétion
* Pour installer une extension « add-on » ou module sur un navigateur web, il faut toujours être prudent avant de les installer, il faut être sûr que c’est extension sont officiels
* L’extension web développer a 2 grands objectifs pour le développeur :
  + Faciliter les tests en désactivant simplement des fonctionnalités
  + Faire apparaître dans la page des éléments non visible du code source
* Developper tool est également un élément très puissant pour faciliter la correction des erreurs de vos pages web
* Google annonce que la rapidité de chargement de vos pages web influe sur le référencement et le positionnement dans les résultats de recherche.
* La bande passante, c’est-à-dire la connexion internet.
* FTP est un outil pour héberger votre site web.
* Développement local : c’est le développement sur votre disque dur
* Une livraison : c’est le fait de mettre un fichier en ligne
* La maintenance évolutive (corrective) : c’est l’ensemble des actions de développement qui vise à corriger des anomalies (bugs)
* Test unitaire : valider une fonctionnalité du site
* Test d’intégration : valider la fonctionnalité dans son ensemble, quand elle est intégrée dans le site complet
* Un fichier livrable : est un fichier qu’il est prêt à être mis en ligne, c’est-à-dire qu’il a subi les étapes de tests et considéré comme apte à être installé sur le site réel
* Un environnement de test : est une copie conforme du site réel, le but de tester les développements avant le passage sur le site réel
* Environnement de production : c’est le site réel
* On planifie généralement les étapes de livraison en production.
* Un fichier est un bien, ayant une valeur ajoutée (visite sur le site, accès à une fonctionnalité, saisie de données

Les bases de JavaScript

De bonnes pratiques :

* + Il faut essayer toujours de générer un code ou un script utile aujourd’hui, dans 2 mois et pourquoi pas dans 5 ans
  + L’indentation
  + Les commentaires (//, /\* \*/)
  + RQ : c’est vrai que les commentaires et l’indentation ajoute quelques octets à la page, c’est-à-dire augmente le poids de la page, mais ils ont un impact négligeable sur le chargement d’une page web
  + Les règles de nommage :
    - Nommage des fonctions : la langue choisie est l’anglais, la première partie de la fonction désigne le type d’action attendue par des préfixes. Voici les préfixes les plus utilisés
      * Is : retourne un résultat vrai ou faux
      * Get
      * Set
      * Add
      * Del
    - Nommage des variables : exemple prixTotal, noCompteur, voici quelques abréviations utilisables :
      * dt : date
      * prix
      * nb : nombre entier
      * no : numéro pour un compteur
      * txt : pour chaine de caractere
      * flag : indicateur de valeur vrai ou faux
  + Règle des pluriels : il faut utiliser le mot au pluriel pour les tableaux
  + Mots réservés
  + Préfixes
* JavaScript n’est pas reconnu par les moteurs de recherche pour le référencement, c’est juste l’HTML qui est connu.
* Il est vivement conseillé pour des raisons de référencement de créer des liens html classique avec la balise <a Href=’page.html’> dans une page de plan de site par exemple.
* Tous les navigateurs utilisent le principe de cache, qui consiste à copier dans une zone de votre disque dur, les fichiers chargés depuis un site web, et ça est important, puisqu’il accélère le chargement de vos pages web.
* Vider le cache est une bonne habitude à prendre quand vous faites des mises à jour sur vos pages. En vidant le cache, vous êtes sûr de charger et d’exécuter la dernière version de vos scripts
* Les variables autorisés en JavaScript doivent contenir les chiffres, les lettres et e caractère spécial Under score (8 du clavier), une variable ne peut pas commencer par u chiffre.
* JavaScript distingue la case (majuscule # minuscule)
* JavaScript est pauvrement typé, c’est-à-dire qu’o peut changer le type d’une variable lors d’un programme, mais il est vivement conseillé d’éviter ça.
* En JavaScript, le type « number »  présente tout (entier, virgule, positif, négatif), il ya pas la notion de taille en JavaScript
* Le type string (chaine de caractère, un seul caractère)
* En JavaScript, une fonction que ne retourne pas de résultat (c’est-à-dire sans le mot clé return) est considérée comme une instruction.
* Portée des variables : locale ou globale
* En JavaScript, les paramètres d’une fonction sont considérés comme des variables locales
* Eviter toujours d’avoir les mêmes noms pour les variables locales, globales et de paramètres pour éviter tout problème de confusion.

JavaScript : Langage Objet

* Il Ya la POO en JavaScript, la seul différence c’est qu’il n y a pas les concepts les plus avancées de la POO comme par exemple l’héritage et les pointeurs.
* Poo nous permet de manipuler des données plus riches et plus complexes.
* En JavaScript, chaque objet lié à un type d’objet, et le type d’objet présente la structure générale des objets.
* En JavaScript, il y a plusieurs objet prédéfinies comme document, les tableaux, les formulaires, et
* Dans chaque classe (type d’objet), on a des propriétés (attributs) et des méthodes (fontions)
* Pour accéder à la propriété d’un objet, les deux écriture sont possible : objet.propriete ou objet[‘’propriete’’]
* Pour les variables de type primitifs, l’affection entre 2 variables, créer deux variables distinctes, c’est-à-dire, si je modifie la valeur d’une variable, l’autre ne sera pas modifié,

Exemple : var txtChaine1 = ‘’salut’’ ; var txtChaine2 = txtChaine1 ;

Ici txtChaine1 et txtChaine2 sont deux variables distinctes