

Certification IT Specialist JavaScript

RS :

Les candidats à cet examen démontreront qu'ils peuvent reconnaître, écrire et déboguer du code JavaScript qui résoudra logiquement un problème

 INFORMATION
TECHNOLOGY
SPECIALIST



Objectifs

Opérateurs,
méthodes et
mots clés
JavaScript

- 1.1 Compléter et déboguer le code qui utilise l'affectation et l'arithmétique des opérateurs : Affectation, incrémentation, décrémentation, addition, soustraction, division, opérateurs de multiplication, de module, d'affectation composée (+=, -=, *=, /=, %=)
- 1.2 Appliquer les bonnes pratiques JavaScript : commentaires, indentation, conventions de nommage, no script, constantes, mots clés réservés, mot clé du débogueur, définition des points d'arrêt, console.log
- 1.3 Evaluer l'utilisation de scripts internes et externes : Quand utiliser, comment utiliser et que se passe-t-il lorsque les scripts sont utilisés à plusieurs niveaux
- 1.4 Implémenter la gestion des exceptions : essayer, attraper
- 1.5 Compléter et déboguer le code qui interagit avec l'objet Browser modèle (BOM) : Affichage des boîtes de dialogue, détermination de la taille de l'écran

Variables, types
de données et
fonctions

- 2.1 Déclarer et utiliser des variables de types de données primitifs : Nombre, booléen, chaîne, nul, indéfini, type d'opérateur, vérification de type fonctions, utiliser la conversion entre les types de données (parseInt, parseFloat) formatage des nombres, opérations sur les chaînes, eval(), toFixed(), toLocaleString (), toprecision(), guillemet simple vs guillemet double (imbrication), initialisation
- 2.2 Déclarer et utiliser des tableaux : Tableaux à une dimension ; tableaux multidimensionnels ; itération ; initialisation ; définir, trier et rechercher un tableau, décaler les méthodes ; propriétés de longueur ; accéder à un élément du tableau
- 2.3 Compléter et déboguer le code qui utilise des objets : Propriétés, méthodes, instanciation, date de l'objet, récupération de la date et de l'heure, localisation du format de date (MM/JJ vs JJ/MM), addition et soustraction

Implémentation du contrôle de flux

- 2.4** Compléter et déboguer le code qui utilise des fonctions mathématiques intégrées ; random, round, abs, floor, ceil, min, max, pow, sqrt
- 2.5** Fonctions complètes et de débogage qui acceptent les paramètres et renvoient les valeurs : code réutilisable, portée locale vs globale, redéfinition des variables, paramètres de passage, valeur vs référence, valeurs de retour

Décisions et boucles

- 3.1** Evaluer les expressions qui utilisent des opérateurs logiques et de comparaison. !=, <, >=, >, !=, &&, ||
- 3.2** Terminer et déboguer les déclarations de décision : Alternative simple (si), alternative double (si sinon), alternative multiple (switch)
- 3.3** Compléter et déboguer les boucles : for, for in, while, do while, break, continue

Modèle d'objet de document

- 4.1** Identifier et construire l'arbre DOM (Document Object Model) : fenêtre, document, corps, autres éléments HTML
- 4.2** Identifier et gérer les événements de document, de formulaire, de clavier et de souris, onload, onfocus, onblur, onchange, onkeypress, onclick, onmouseout, onmouseover
- 4.3** Compléter et déboguer le code qui sort dans un document HTML : document write, innerHTML, textContent
- 4.4** Compléter et déboguer le code qui localise, modifie et ajoute du HTML éléments et attributs des documents : getElementById, getElementsByTagName, getElementsByClassName, setAttribute, createElement
- 4.5** Créer des événements à l'aide de gestionnaires d'événements et d'écouteurs : Événements DOM, événement d'attribut HTML, addEventListener

Formulaires HTML

- 5.1** Compléter et déboguer le code qui récupère l'entrée du formulaire et définit les valeurs de champ du formulaire ; identifier le chemin DOM ; obtenir des valeurs de différents types d'éléments ; pré remplissage, masquage et mise à jour de la valeur
- 5.2** Compléter et déboguer le code qui effectue la validation d'entrée, comparaisons de chaînes, Not-A-Number (NaN),
- 5.3** Décrire le processus de soumission du formulaire onSubmit, POST vs. GET, cibles potentielles pour la soumission