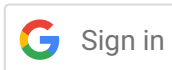


## Bonjour, Demo User #919

- Tous documents et usage de Google Chrome autorisés.
- Communication ou usage d'un moyen de communication (email, chat, sms...) => note de 0.
- Copiage => note de 0 pour les étudiants concernés.
- Lisez attentivement les énoncés et choix proposés.
- Il est recommandé d'utiliser la console JavaScript pour tester vos solutions.

À vous de jouer !



## QCM

### Question 1 (1 points)

```
function bonjour() {  
  return 'bonjour !';  
}  
var resultat = bonjour();
```

Qu'est-ce que ce code ne contient PAS:

- Affectation
- Définition de fonction
- Appel de fonction
- Concaténation

### Question 2 (1 points)

```
for (var i = -1; i < 1; i++) {  
  bonjour();  
}
```

Combien de fois la fonction `'bonjour'` va-t-elle être appelée ?

- 0 fois
- 1 fois
- 2 fois
- 3 fois

### Question 3 (1 points)

```
if (a === 1) {  
  if (b === 1) {  
    // code sans importance  
  } else {  
    // code sans importance  
  }  
} else {  
  // code sans importance  
}
```

Si on voulait représenter ces conditions sous forme d'un arbre de décision, combien contiendrait-il de niveaux ?

- 1
- 2
- 3
- 4

### Question 4 (1 points)

Supposons que nous ayons un tableau `fruits` ayant la valeur suivante:

```
[ 'kiwi', 'prune', 'abricot', 'fraise' ]
```

Que se passe-t-il si on exécute l'instruction suivante:

```
fruits[3] = 'litchi';
```

- "litchi" va être ajouté à la fin du tableau
- "litchi" va remplacer l'abricot
- "litchi" va remplacer la fraise
- cette expression vaut false

### Question 5 (1 points)

Supposons que nous ayons un tableau `fruits` ayant la valeur suivante:

```
[ 'kiwi', 'prune', 'abricot', 'fraise' ]
```

Que se passe-t-il si on exécute l'instruction suivante:

```
fruits.splice(1, 0, 'mûre');
```

- un élément "mûre" sera inséré entre kiwi et prune
- un élément "mûre" sera inséré juste avant kiwi
- le deuxième élément (prune) va être remplacé par "mûre"
- le tableau restera inchangé

## Exercices de codage

### Question 6 (3 points)

## Condition simple

Supposons qu'une variable `monNumero` ait été créée et contienne un nombre.

Implémenter une condition qui affiche `'plus grand que 4'` dans la console, si `monNumero` est strictement supérieur à `4`.

Saisissez votre code Javascript ici

```
if (monNumero > 4) {  
  console.log('plus grand que 4');  
}
```

---

### Question 7 (3 points)

## Condition avancée

---

Nous allons écrire un programme qui donne la météo à l'utilisateur, pour la ville de son choix.

Ce programme devra:

- Inviter l'utilisateur à saisir le nom d'une ville;
- Dans le cas où l'utilisateur a saisi `roubaix`, répondre `nuageux` ;
- Dans le cas où l'utilisateur a saisi `marseille`, répondre `soleil` ;
- Sinon, répondre `mauvaise saisie` .

Les réponses sont à afficher à l'aide de la fonction `alert()` . Respecter les textes fournis à la lettre. (y compris la casse, espaces et ponctuation)

Saisissez votre code Javascript ici

```
var ville = prompt();  
if (ville === 'roubaix') {  
  alert('nuageux');  
} else if (ville === 'marseille') {  
  alert('soleil');  
} else {  
  alert('mauvaise saisie');  
}
```

---

### Question 8 (3 points)

## Fonctions

---

Définir une fonction `multiplier` qui retourne le résultat de la multiplication des trois nombres passés en paramètres.

Exemple d'appel: `multiplier(1, 3, -2)`; doit retourner `-6` (`1 * 3 * -2`).

Saisissez votre code Javascript ici

```
function multiplier(a, b, c) {  
  return a * b * c;  
}
```

---

### Question 9 (3 points)

## Boucles

---

Définir une fonction `nFois` qui prend comme paramètres un nombre `n` et une chaîne de caractères `message`, et retourne une chaîne de caractères contenant `n` fois `message`, en utilisant la concaténation.

Exemple d'appel: `repeteChaine(3, 'bonjour!');` doit retourner `'bonjour!bonjour!bonjour!'`.

Saisissez votre code Javascript ici

```
function nFois(n, message) {  
  var result = '';  
  for (var i=0; i<n; i++) {  
    result = result + message;  
  }  
  return result;  
}
```

---

### Question 10 (3 points)

## Tableaux

---

Définir une fonction `tableauContient` qui prend deux paramètres:

- `tableau` : un tableau de nombres
- `val` : un nombre

...et retourne:

- `true` s'il existe au moins un élément ayant la valeur `val` dans le tableau `tableau`,
- ou `false` sinon.

### Exemples d'appels:

- `tableauContient([1, 2, 3], 2);` doit retourner `true` .
- `tableauContient([1, 2, 3], 4);` doit retourner `false` .

Saisissez votre code Javascript ici

```
function tableauContient(tableau, val) {  
    return tableau.indexOf(val) !== -1;  
}
```

---

## Rendu de copie

La copie doit être rendue **avant** l'heure de fin d'examen, telle qu'annoncée par l'enseignant.

Une fois votre copie rendue, vous ne pourrez plus modifier vos réponses.

✓ RENDRE LA COPIE