|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Matière examinée** | ES6, ES7 | **Nom** |  |
| **Date de l’épreuve** | Exo 2 | **Prénom** |  |

**Variable**

1. **Créer une constante avec la valeur 2;**
2. **Peux-t-on modifier la valeur d’une constante après ? Non une constante ne peut être modifier une fois qu’elle à été initialisé.**
3. **Dans quel cas nous pouvons utiliser le mot clé let**

**Le mot clé let peut être utilisé pour une variable qui à vocation à être modifier une fois initialisé (non constante)**

1. **Quelle différence avons-nous entre un let et const ?**

**Après une première instanciation la valeur d’une let peux être modifier ce qui n’est pas le cas pour une const.**

**Arrow**

1. **Créer une fonction arrow qui prend en paramètre un nombre et retourne ce nombre multiplié**
2. **Le mot clé retourne est obligatoire dans une fonction arrow ?**

**Non, dans une fonction arrow le mot clé return n’est pas nécéssaire.**

1. **Créer une fonction arrow qui prend en paramètre un nombre avec une valeur par défaut à 10 et retourne ce nombre multiplié**

**Rest/Spread**

1. **Que veut dire les paramètres Rest et Spread Operator**

**Le paramètre Rest permet de récupérer les arguments passé dans une fonction mais non prédéfini dans un tableau. Le spread lui permet de déversé les valeurs d’un tableau dans les paramètres d’une fonction.**

1. **Déclarer une fonction qui prends 3 paramètre, le troisième paramètre doit être un paramètre Rest.**

**Cette fonction doit additionner toutes les valeurs passées en paramètre et retourner le résultat.**

***Ex: name(a, b, ...x) // name(1, 2, 3, 4, 5, 6)***

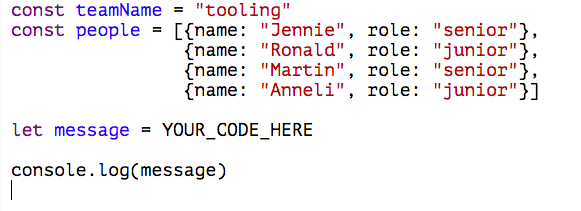
1. **Maintenant au lieu de passer 1,2, 3, 4 en paramètre de votre fonction, créer un tableau contenant les valeurs [1, 2, 3, 4] et passer ces paramètres en mode spread.**
2. **Pourquoi il est intéressant d’utiliser les principes en Rest et Spread ?**

**C’est principes sont intéressante pour des raison de sécurité.**

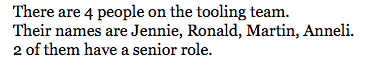
**Templates String**

1. **Créer deux variables qui contiennent les valeurs “hello” et “world”, et maintenant concaténer ces valeurs avec le principe des templates String**
2. **Créer deux variables qui contiennent votre nom et prenom, et maintenant concatener ces valeurs avec le principe des templates String**
3. **Si vous ne devez pas utiliser les templates string comment allez-vous concaténer ces valeurs ?**

**Pour se faire nous utiliserions l’opérateur ‘+’**

****

1. **Maintenant créer plusieurs des templates strings qui doivent retourner les valeurs suivantes**

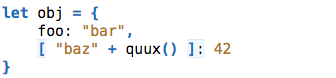
****

**Objects properties**

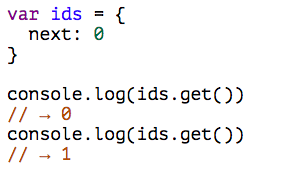
1. **Créer une variable a contenant la valeur “hello world”; maintenant créer un objet avec pour nom “obj” et passer la variable a en mode “property shorthand”**
2. **Créer une variable b contenant la valeur [1, 2, 3]; maintenant créer un objet avec pour nom “obj2” et passer la variable b en mode “property shorthand” dans cette objet**
3. **Maintenant créer un objet “obj3” et passer la deux objets de la question 15 et 16 en mode “property shorthand”**

**Objects properties**

1. **Créer une variable a contenant la valeur “hello world”; maintenant créer un objet avec pour nom “objComputed” et par la suite le contenu de la variable a doit servir de clé dans objet objComputed.**

****

1. **Maintenant, ajouter la fonction “maFunction” dans l’objet objComputed, en utilisant le principe de *Enhanced Object Properties, Method Properties.***
2. **Compléter le code ci-dessous afin d’avoir les résultats affichés en commentaire**

******

**Destruction Assignment**

****

1. ***Créer deux variables a, b contenu dans un tableau en utilisant le principe d’assignement par destruction***

******

1. ***Créer deux variables nom et annee, qui récupèrent les valeurs respectives du film ci-dessus utilisant le principe d’assignement par destruction***
2. ***Créer une variable like, qui récupère la valeur Like du film ci-dessus utilisant le principe d’assignement par destruction et afficher à l’écran.***

****

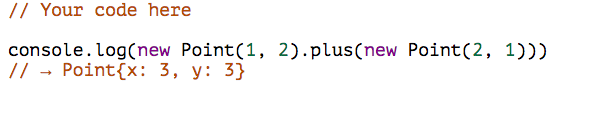
1. ***Pouvez-vous expliquer le code ci-dessus ?***

***Le code ci-dessus recupere les valeurs correspondant aux clés entré en paramètres avant d’en afficher les valeurs associées***

1. ***Créer une fonction qui prends 1 objet javascript, cette objet est détruit avec la variable a. Par la suite, exécuter cette fonction // Ex. maFunction({ a: 1, b: 1});***
2. ***Créer une fonction qui prends 1 objet javascript, cette objet est détruit avec deux variables a et b. Par la suite exécuter cette fonction // Ex. maFunction2({ a: 1, b: 1});***

**Classes**

1. ***Créer une classe vide et exécuter cette classe***
2. ***Créer une classe contenant un constructeur qui prend le paramètres a, b, avec une méthode affiche.***
   1. ***Maintenant, créer une instance de classe avec deux paramètres***
   2. ***Ensuite exécuter la méthode affiche qui doit retourner la valeur a et b multiplié***
3. ***Créer la classe et les méthodes qui doivent retourner cette valeur***

****