

Tableaux à dimensions multiples

Il est fréquent que l'on ait besoin de tableau à dimensions multiples pour gérer des problématiques, notamment en mathématique, en statistique...

JavaScript offre cette possibilité.

1. Syntaxe

Comme pour les tableaux à dimension unique, JavaScript permet de déclarer les tableaux à dimensions multiples de plusieurs façons :

- avec une syntaxe littérale,
- avec une syntaxe dite "Programmation orientée objet".

Avec une syntaxe dite "Programmation orientée objet" (encore appelée JSON - *JavaScript Object Notation*), la déclaration d'un tableau de nom `tabMatrice` de deux lignes subdivisées en quatre colonnes avec initialisation se fait comme suit :

```
/* Déclaration du tableau tabMatrice */
var tabMatrice tableau = new Array();

/* Déclaration de la première "ligne" du tableau tabMatrice */
tabMatrice[0]=new Array()

/* Initialisation des 4 "colonnes" de la première "ligne" */
tabMatrice[0][0] = "Un";
tabMatrice[0][1] = "Deux";
tabMatrice[0][2] = "Trois";
tabMatrice[0][3] = "Quatre";

/* Déclaration de la deuxième "ligne" du tableau tabMatrice */
tabMatrice[1]=new Array()

/* Initialisation des 4 "colonnes" de la deuxième "ligne" */
tabMatrice[1][0] = "Onze";
tabMatrice[1][1] = "Douze";
tabMatrice[1][2] = "Treize";
tabMatrice[1][3] = "Quatorze";
```

2. Exercice n°15 : Mini-tableur

Sujet

Soit le tableau `tb` à deux dimensions comportant quatre lignes et cinq colonnes. Réaliser les traitements suivants :

- saisir au clavier des valeurs dans les trois premières lignes et les quatre premières colonnes (on conserve la dernière ligne et la dernière colonne libres pour des additions de lignes et de colonnes),
- additionner les colonnes en dernière ligne et les lignes en dernière colonne.

```
/* Déclaration de variables locales */
var tb = new Array(5);
var numLigne, numColonne;
var valeur;

/* Déclaration de 5 "colonnes" par "ligne" pour le tableau tb */
for (var numLigne=1; numLigne<tb.length; numLigne++)
{
    /* Création des "colonnes" (numérotées de 0 à 5) */
    tb[numLigne]=new Array(6);
}

/* Initialisation du tableau tb */
valeur = 1;
for (numLigne=1; numLigne<=3; numLigne++)
{
    for (numColonne=1; numColonne<=4; numColonne++)
    {
        tb[numLigne][numColonne] = valeur;
        valeur = valeur + 1;
    }
}

/* Mise à zéro des totaux en ligne n°4 */
for (numColonne=1; numColonne<=5; numColonne++)
{
    tb[4][numColonne] = 0;
}

/* Mise à zéro des totaux en colonne n°5 */
for (numLigne=1; numLigne<=4; numLigne++)
{
    tb[numLigne][5] = 0;
}

/* Détermination des totaux en ligne n°4 et en colonne n°5 */
for (numLigne=1; numLigne<=3; numLigne++)
{
    for (numColonne=1; numColonne<=4; numColonne++)
    {
        /* Totalisation en ligne n°4 */
        tb[4][numColonne] = tb[4][numColonne]
        + tb[numLigne][numColonne];
        /* Totalisation en colonne n°5 */
        tb[numLigne][5] = tb[numLigne][5]
        + tb[numLigne][numColonne];
        /* Totalisation générale en ligne n°4-colonne n°5 */
        tb[4][5] = tb[4][5] + tb[numLigne][numColonne];
    }
}
```

```
/* Affichage du total général */  
/* NB : Total de 78 étant donné la technique de remplissage retenue  
du tableau tb */  
document.write("Total général en tb[4][5] = " + tb[4][5]);
```

Commentaires du code JavaScript

Le tableau `tb` est dans un premier temps déclaré comme étant un tableau à une seule dimension avec cinq cellules, implicitement numérotées de 0 à 4 comme suit :

```
var tb = new Array(5);
```

Vous noterez que la ligne de numéro zéro ne sera pas utilisée par la suite. Ce choix a été fait car l'utilisation ultérieure de cette ligne ne serait pas très intuitive.

Dans un second temps les lignes de ce tableau sont elles-mêmes subdivisées en colonnes, comme ceci :

```
/* Déclaration de 5 "colonnes" par "ligne" pour le tableau tb */  
for (var numLigne=1; numLigne<tb.length; numLigne++)  
{  
    /* Création des "colonnes" (numérotées de 0 à 5) */  
    tb[numLigne]=new Array(6);  
}
```

Là encore, six colonnes sont déclarées (numérotées de 0 à 5) et la colonne numéro zéro sera ignorée par la suite.

L'initialisation des "cellules" du tableau `tb` (uniquement les lignes 1 à 3 et les colonnes 1 à 4) se fait avec la notation suivante :

```
tb[numLigne][numColonne] = valeur;
```

La valeur affectée est une valeur de 1 en progression à partir de 1 (donc 1 à 12). La saisie des données aurait bien sûr pu être effectuée à partir d'une saisie clavier.

Les traitements suivants (mise à zéro de la ligne n°4 et de la colonne n°5) n'amènent pas de commentaires particuliers.

Les totaux en ligne n°4, en colonne n°5 et en cellule [4][5] se font également par un jeu de boucles `for` imbriquées. Le tableau général est naturellement de 78 étant donnée la technique de remplissage du tableau retenue.