

MODIFIER LA COULEUR D'ARRIÈRE PLAN DYNAMIQUEMENT

Objectifs

Savoir mettre en œuvre un formulaire.

Vérifier la saisie dans un formulaire avec une expression régulière (avec Javascript) .

Savoir modifier le style d'un document.

Énoncé

Vous devez construire un formulaire avec un unique champ de saisie : celui-ci permet à l'utilisateur de saisir un « code couleur » selon les valeurs RGB (Red Green Blue) ou RVB (Rouge Vert Bleu).

Rappel :

- La notation commence par #
- Suivi de 3 doublets de valeurs représentant l'intensité des couleurs rouge, puis vert, puis bleu.
- En hexadécimal, l'intensité varie de 00 (0 en décimal) à ff (255 en décimal).

Exemples :

- #ffffff : blanc
- #000000 : noir
- #ff0000 : rouge

Lorsque l'utilisateur valide sa saisie, vous devez concevoir un script qui testera la valeur saisie avec une expression régulière de la forme `/^[0-9A-Fa-f]{6}$/` (voir les explications ci-après).

Si la saisie n'est pas conforme, l'utilisateur est prévenu, il ne se passe rien de plus.

Si la saisie est conforme, vous utiliserez la valeur saisie pour modifier la couleur d'arrière plan du document (en utilisant les propriétés CSS – voir « Modifier le style CSS d'un nœud » dans le cours).

1. Créez le formulaire .
2. Gestionnaire d'événement : mettez en place un gestionnaire d'événements à la soumission du formulaire, qui invoque la fonction JavaScript qui effectue les vérifications et les traitements . Voir ce tutoriel : http://www.w3schools.com/js/js_form_validation.asp (à adapter !). Rédigez ensuite la fonction JavaScript qui vérifie la saisie selon l'expression régulière donnée, et alerte l'utilisateur en cas d'erreur.
3. Ajoutez au gestionnaire d'événement créé en (2) un appel à une fonction qui permet de changer la couleur de la page.
4. Écrivez cette fonction. Exemple de code JavaScript pour peindre l'élément dont l'identifiant est *toto* (attribut `id="toto"`) en gris :

```
document.getElementById("toto").style.backgroundColor="#CCCCCC" ;
```

L'exercice est terminé lorsque votre formulaire de modification de la couleur d'arrière plan est fonctionnel.

Quelques points à vérifier pour valider votre travail

1. Votre code est-il validé par l'outil en ligne du W3C ? Ou à l'inverse y a-t-il des éléments que vous ne savez pas corriger ou pour lesquels vous ne comprenez pas les avertissements ou erreurs signalées ? Lesquels ?
2. Arrivez-vous à faire la modification de l'arrière plan, sans erreur Javascript ?
3. Que se passe-t-il si la saisie de l'utilisateur n'est pas conforme ?
4. Que se passe-t-il si l'utilisateur appuie sur la touche « entrée » de son clavier plutôt que

d'utiliser la souris pour cliquer sur le bouton du formulaire ? Pourquoi ?

Boîte à outils

Quelques informations sur l'utilisation des expressions régulières avec Javascript :

Pour déclarer une expression régulière

```
var monExp = /monExpressionReguliere/;
```

→ Définit le « masque » de l'expression régulière entre / et /

Exemple d'expression régulière

Masque pour une référence de produits de type :

2lettresMajuscules#2à4chiffresOuLettresMinuscules (par exemple AA#12ra ou AZ#trsd ou AZ#12)
`/^[A-Z]{2}#[0-9a-z]{2,4}$/`

Le masque donné en exemple se lit comme suit

<code>^</code>	Depuis le début de la chaîne
<code>[A-Z]{2}</code>	Un caractère entre A et Z répété exactement 2 fois
<code>#</code>	Un caractère #
<code>[0-9a-z]{2,4}</code>	Un caractère entre a et z ou entre 0 et 9 répété de 2 à 4 fois
<code>\$</code>	La chaîne doit être terminée

La méthode test

C'est une méthode de l'objet RegExp (expressions régulières).

Paramètre : chaîne = la chaîne à tester

S'applique à : une expression régulière

La méthode teste si une chaîne passée en paramètre correspond au masque d'expression régulière défini. Renvoie true si quelque chose a été trouvé et false dans le cas contraire.

Par exemple, si le masque donné précédemment est utilisé :

```
var monExp = /^[A-Z]{2}#[0-9a-z]{2,4}$/;
// monExp.test("AA#12ra" ) renverra true
// monExp.test("1234" ) renverra false
```