**Mini-projet JavaScript**

*Sujet*

On s'intéresse à la publication d'un document annoté tel que celui qui apparaît à la figure ci-dessous. Ce document est constitué d'un texte principal, et de commentaires, qui apparaissent sur la droite. Chacun de ces commentaires porte sur une portion du texte. Les portions commentées apparaissent comme soulignées dans le texte. Lors du survol d'un commentaire, la portion de texte qu'il commente apparaît en surbrillance (voir deuxième image de la figure) ;

En haut de la page on trouve trois éléments de type *button*.

1. le premier a pour texte ***Aérer***. Un clic sur ce bouton a pour effet d'augmenter la marge interne de tous les éléments p. Chaque clic augmente de *5px* la valeur de cet espacement dans les 4 directions.

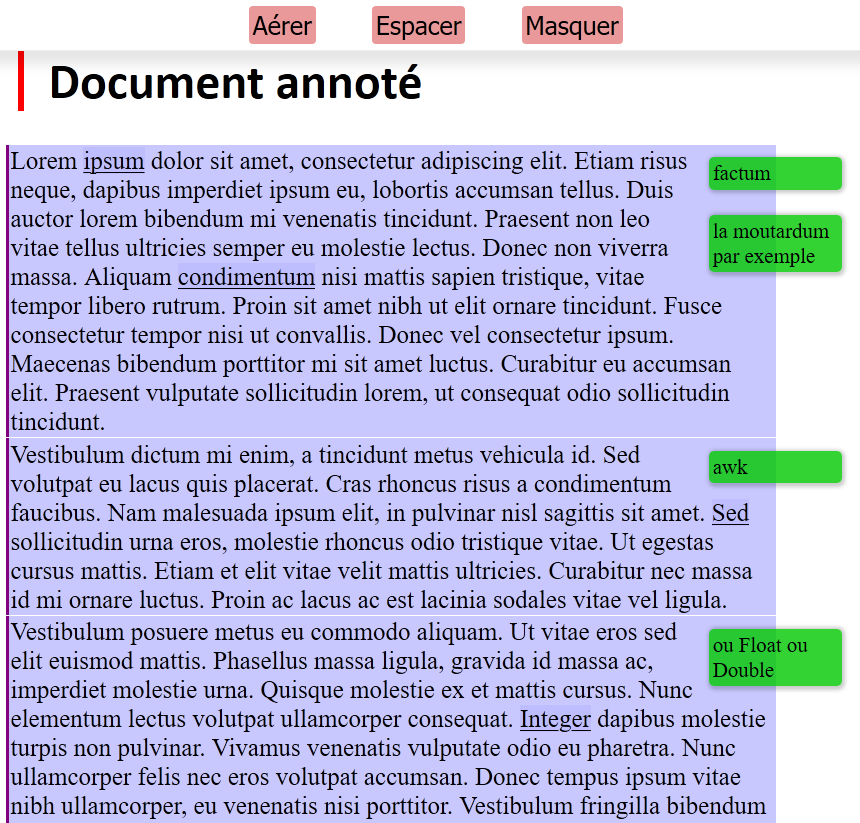
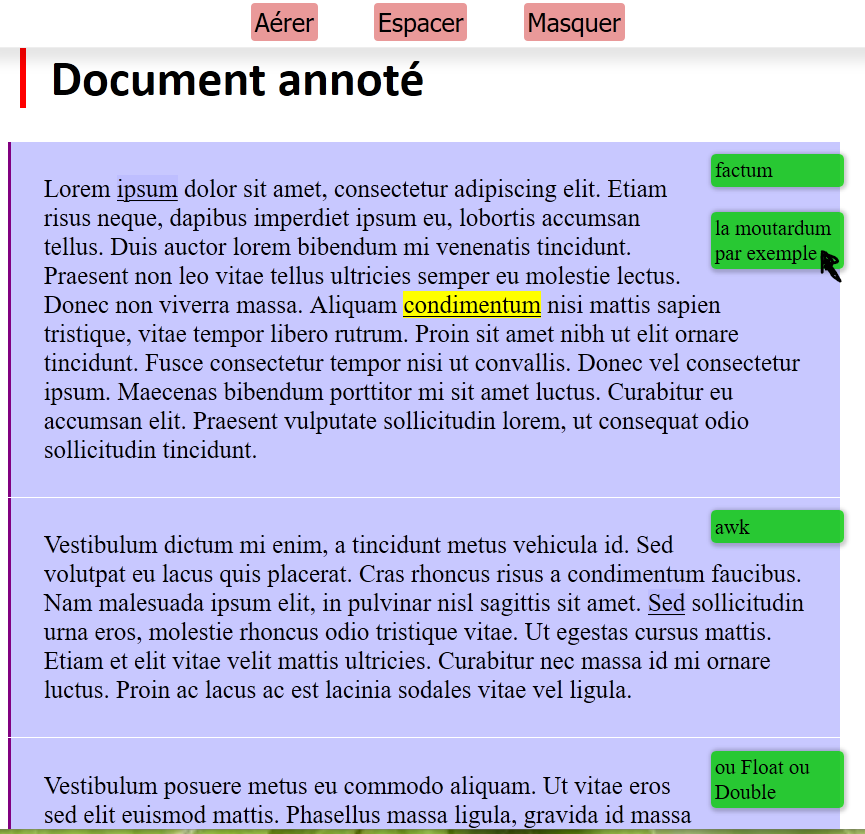
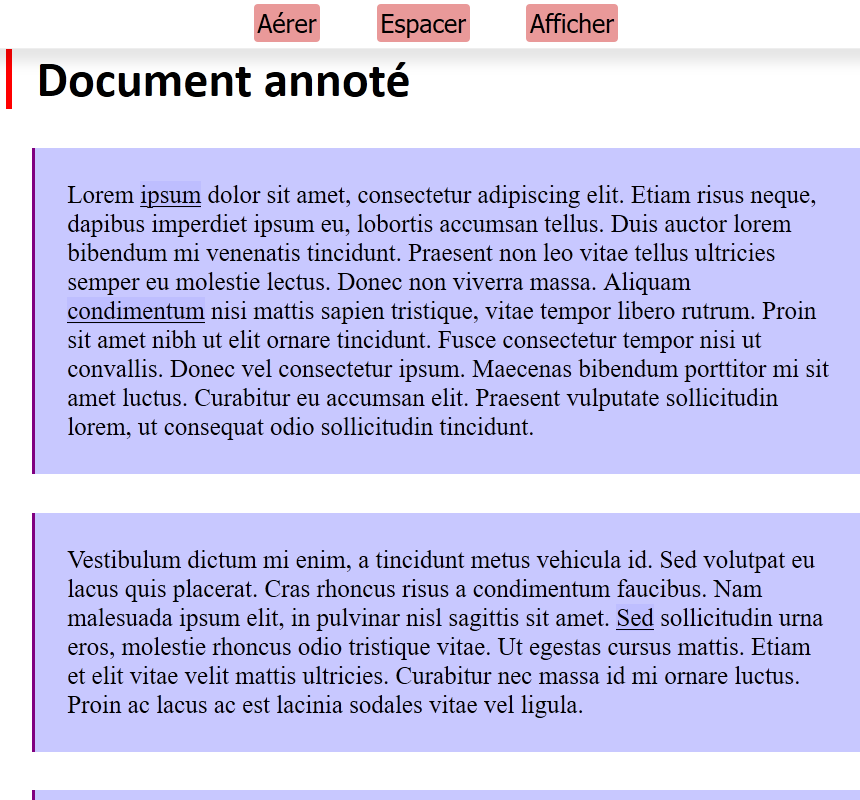
La deuxième image de la figure montre le résultat obtenu après plusieurs clics sur ce bouton à partir de la situation initiale affichée dans l'image sur la gauche.

1. le deuxième a pour texte ***Espacer***. Un clic sur ce bouton a pour effet d'augmenter la marge externe de tous les éléments p. Chaque clic augmente de *5px* la valeur de cet espacement dans les 4 directions.

La troisième image de la figure montre le résultat obtenu après plusieurs clics sur ce bouton à partir de la situation affichée dans l'image au centre.

1. le troisième a pour texte ***Masquer***. Un clic sur ce bouton a pour effet :
   1. de masquer tous les commentaires ;
   2. de remplacer le texte ***Masquer*** de cet élément par le texte ***Afficher*** ;
   3. le prochain clic sur cet élément aura pour conséquence de rétablir l'état initial (commentaires affichés, texte ***Masquer*** et réinstallation du comportement initial du clic).

La troisième image de la figure montre le résultat obtenu après un clic sur ce bouton.

        La première image représente la situation initiale, la seconde après plusieurs clics sur le bouton *Aérer* et pendant le survol par la souris du second commentaire, la troisième après plusieurs clics sur le bouton *Espacer* et masquage des commentaires (clic sur troisième bouton).

Les éléments correspondant à un commentaire sont tous de classe *"comment"* et ont un id de la forme *i-comment* où *i* correspond au numéro du commentaire, le premier ayant le numéro 1.

Les éléments correspondant aux portions commentées de texte sont tous de classe *"annote"* et ont un id de la forme *i-annote* où *i* correspond au numéro du commentaire qui lui est associé.

Ainsi le code html du second commentaire, survolé par la souris dans la seconde image de la figure est :

|  |  |
| --- | --- |
| 1  2  3 | <div class="comment" id="2-comment">    la moutardum par exemple  </div> |

et celui du texte associé est :

|  |  |
| --- | --- |
| 1 | ...  Aliquam <span class="annote" id="2-annote">condimentum</span> nisi mattis... |

On constate bien que leurs deux id commencent par le même numéro (ici *2*). Dans la suite, on supposera sans avoir besoin de le vérifier qu'à chaque commentaire correspond bien une portion annoté du texte, et réciproquement.

**Travail à réaliser**

Le travail que vous devez réaliser consiste à écrire le code javascript permettant de mettre en œuvre les comportements décrits ci-dessus.

Vous devez utiliser les fichiers contenus dans l'archive [**fichiers-mini-projet-js2020.zip**](https://www.fil.univ-lille1.fr/~routier/enseignement/licence/tw1/tdtp/mini-projet/js-2020/fichiers-mini-projet-js2020.zip) fournie. Cette archive contient :

* le fichier **mini-projet-js-2020.html** que vous renommerez en ***votreNom*-projet.html** mais dont vous ne modifierez pas le contenu,
* le fichier **style/style-projet-js.css**, vous ne devez pas le modifier non plus,
* le fichier **scripts/projet-js.js** que vous compléterez avec le code javascript que vous écrirez.  
  Toutes les fonctions que vous écrirez dans ce fichier devront être documentées et leur code commenté quand nécessaire.

1. Consultez le code html fourni pour identifier les différents éléments et en particulier leurs id.
2. Codez une fonction getNumber() qui à partir d'une chaîne de caractères passée en paramètre permet de récupérer l'entier en tête de cette chaîne (donc ici i) ?

Par exemple :

|  |  |
| --- | --- |
| 1 | getNumber("2-comment")     // -> 2 |

1. On appelle surligne la fonction *listener* qui est exécutée lors du survol d'un commentaire.

Codez cette fonction qui a pour effet de donner la valeur *"yellow"* à la couleur d'arrière plan du texte correspondant au commentaire qui est survolé et qui a donc déclenché l'événement.

Codez une fonction qui permet de rétablir la couleur d'arrière plan initiale du texte annoté lors de l'arrêt du survol à la souris d'un commentaire.

Faites le nécessaire pour que le comportement attendu lors du survol des commentaires soit actif au chargement de la page.

1. Ecrivez le code javascript nécessaire pour mettre en œuvre le comportement relatif aux clics sur le bouton *Masquer*.

Ce comportement doit être actif au chargement de la page.

1. Complétez votre code en définissant les fonctions qui correspondent au traitement à exécuter lors d'un clic sur les boutons *Aérer* et *Espacer*, puis faites le nécessaires pour que ces comportements soient actifs au chargement de la page.
2. Rendez votre travail sous la forme d'une archive. Le nom de cette archive sera ***votreNom*-projetJS.zip**. Cette archive contiendra un répertoire dont le nom sera ***votre-nom*-projet**. Le contenu de ce répertoire sera organisé ainsi :
   * un fichier **lisezmoi.txt** avec vos nom et prénom. Ce fichier mentionnera pour chacune des questions précédentes si elle a été traitée ou non et les éventuels problèmes de votre solution par rapport au cahier des charges demandé. Si vous avez ajouté des fonctionnalités, vous les détaillerez également dans ce fichier.
   * les fichiers mentionnés en introduction.

Tous vos fichiers seront codés en UTF-8. Les noms des fichiers (y compris leurs extensions) seront en minuscules.