L2 MI - 2023/2024 UE Développement Web David Lesaint

## TP JS 7 - Manipulation statique du DOM

Décompressez l'archive déposée sur Moodle pour ce TP. Le dossier résultant contient différents fichiers à réutiliser ou à compléter. Pensez à consulter le site MDN, et pour visualiser ce qui est attendu, ce démonstrateur.

## Exercice 1. Triangle de Pascal

L'objectif est de produire un fichier **pascal.html** qui génére le triangle de Pascal sous forme d'un tableau de dimension  $(N+1)\times (N+1)$  où  $0\leq N\leq 7$  est un paramètre choisi par le visiteur. Pour tout k et n vérifiant  $0\leq k\leq n\leq N$ , la cellule située sur la ligne de rang n et la colonne de rang k a pour valeur le coefficient binomial C(n,k). Les Figures 1 et 2 illustrent le résultat attendu pour N=0 et N=7, respectivement. Par exemple, la cellule (2,1) située sur la 3ème ligne et la 2<sup>nde</sup> colonne a pour valeur C(2,1)=2 ainsi qu'illustré en figure 2. La valeur d'une cellule est obtenue en sommant celle du dessus (valeur 0 si elle est vide ou n'existe pas) et celle située à gauche de cette dernière (valeur 0 si elle est vide ou n'existe pas).

Le fichier **pascal.html** importe le fichier **pascal.js** et contient un champ de type **number** suivi d'un bloc **div**. Deux gestionnaire d'évènements sont implémentés à la fin du fichier **pascal.js**. Le premier se déclenche au chargement de la page, accède à la valeur N prédéfinie du champ libellé N, et appelle avec cette valeur la fonction **creerTableau** qui se chargera de générer le tableau HTML correspondant au tableau  $(N+1)\times(N+1)$  des coefficients C(n,k). Le second se déclenche à chaque changement de valeur du champ N, supprime le tableau existant et rappelle **creerTableau** pour regénérer le tableau en fonction de la nouvelle valeur de N. Implémentez la fonction **creerTableau** en trois étapes :

- 1. Génération d'un tableau HTML à N lignes et N cellules par ligne à insérer dans le bloc div.
- 2. Remplissage des cellules du tableau HTML située sous la diagonale en appliquant les règles de calcul suivantes :
  - C(n,0) = 1 pour 0 < n < N.
  - C(n, n) = 1 pour 0 < n < N.
  - C(n,k) = C(n-1,k-1) + C(n-1,k) pour  $0 \le k \le n \le N$ .
- **3.** Ajout de l'abbréviation HTML de titre C(n,k) à la cellule (n,k) pour tout  $0 \le k \le n \le N$ . Chaque abbréviation apparaîtra au survol de sa cellule ainsi qu'illustré en Figure 2 pour la cellule (2,1) de valeur 2.

 $N = \boxed{0 \ \odot}$ 

1

FIGURE 1 - C(n, k) pour  $0 \le k \le n \le 0$ .

FIGURE 2 – C(n, k) pour  $0 \le k \le n \le 7$ .

## Exercice 2. Gouvernements

Le fichier **gouvernements.html** affiche un tableau HTML. L'objectif est de formatter ce tableau en fonction d'évènements utilisateur. Vous compléterez à cet effet le fichier **gouvernements.js**.

- 1. Créez un tableau qui stocke chaque triplet d'entiers apparaissant sur une ligne du tableau HTML dans un tableau JS à 3 éléments. Ce tableau ne doit contenir ni les en-têtes, ni les noms de continents.
- 2. Créez un tableau qui stocke chaque ligne du tableau HTML, sauf la ligne d'en-têtes, sous la forme d'un objet ayant 4 propriétés dénommées continent, republic, monarchy et other.
- 3. Implémentez la fonction effacerFormattage() qui applique à toute cellule du tableau HTML contenant un entier la mise en forme suivante :
  - son fond est blanc,
  - le texte est en noir, et
  - la graisse (weight) des chiffres est normale.
- 4. Implémentez la fonction formaterMinMax(ncol) qui prend en entrée un numéro de colonne du tableau HTML (ncol) et applique aux 2 cellules contenant les valeurs minimale et maximale de la colonne la mise en forme suivante :
  - le texte est en gras,
  - le texte est de couleur rouge pour la valeur minimale et vert clair pour la valeur maximale,
  - le fond est de couleur rose pour la valeur minimale et vert pour la valeur maximale.

Appuyez-vous sur l'un ou l'autre des tableaux produits lors des questions précédentes pour l'implémentation. La Figure 3 illustre le résultat obtenu sur la colonne Other.

5. Remplacez le texte des en-têtes du tableau HTML Republic, Monarchy et Other par des boutons (éléments button ou input de type button). Chaque bouton affiche le même texte que l'en-tête qu'il remplace. Le gestionnaire d'événement pré-enregistré sur chaque bouton se chargera d'appeler les fonctions effacerFormattage et indiquerMinMax pour la colonne concernée à chaque clic visiteur.

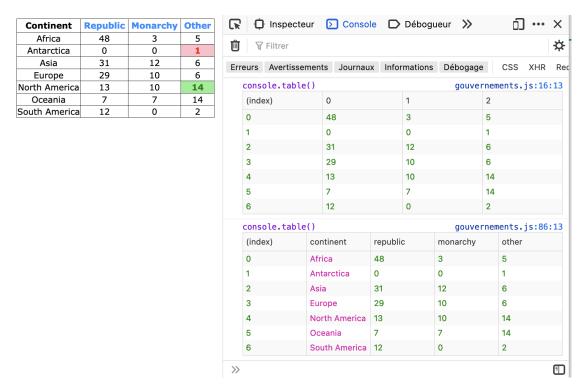
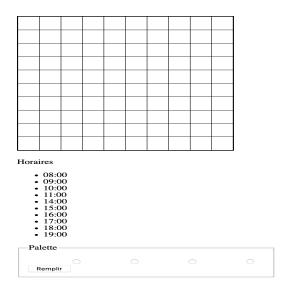


FIGURE 3 – Page web et console après clic sur l'entête de la colonne "Other"

## Exercice 3. Horaires

Le fichier horaires.html importe horaires.js. Complétez ce dernier pour obtenir la page illustrée en Figure 5.

- 1. Ajoutez une ligne d'en-tête au tableau HTML dont les cellules seront une copie des items de la liste : le noeud texte du i-ème item sera copié dans celui de la i-ème cellule. Supprimez la liste une fois la copie effectuée.
- 2. Le fichier horaires.js définit deux tableaux dénommés t\_labels et c\_labels. Accédez aux valeurs du premier pour créer le contenu texte des 4 éléments HTML label. Accédez aux valeurs du second pour définir la couleur de fond CSS de ces 4 éléments. Par exemple, A éviter et orange seront respectivement le contenu texte et la couleur de fond du troisième label.
- 3. Modifiez les 4 boutons radio en affectant à leur attribut HTML name la même valeur preference. Affectez à l'attribut HTML value du i-ème bouton le i-ème élément du tableau c\_labels. Par exemple, red sera la valeur du second bouton.
- 4. Ajoutez un bouton HTML de type submit, de nom ok et de valeur OK à la suite du bouton existant.
- 5. Ajoutez un élément HTML de type form et sans attributs contenant un clone complet de l'élément fieldset.



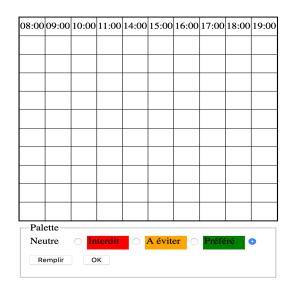


Figure 4 – Page sans exécution JS

Figure 5 – Page avec exécution JS