

TP4

Découverte Programmation Web

Concepts: CSS: pseudo-classes, pseudo-éléments, flexbox, dégradés (gradient), ombres (text-shadow, box-shadow), polices personnalisées, responsive-design (media queries)

Objectifs du TP :

- Améliorer le design du site avec des effets CSS3 (ombres, dégradés)
- Utiliser les flexbox et adapter le site en fonction des résolutions d'écran
- Personnaliser le site avec une police qui n'est pas installée sur l'ordinateur par défaut.

Travail de la séance :

- **Le titre du site :**
 - Ajoutez une ombre de texte au titre de votre site, vous avez le choix de la couleur (nous déclinons toute responsabilité en cas de crise d'épilepsie due à des couleurs trop flashies...)
- **Les liens :**
 - Jusqu'à présent chaque lien survolé et visité avait un style par défaut donné par le navigateur. Maintenant vous savez comment les changer (pseudo-classe **hover** et pseudo-classe **visited**). A vous de mettre le style qui convient à votre site sur vos liens.
- **Le menu :**
 - Adieu notre menu de la préhistoire, nous allons le remettre au goût du jour grâce à un joli dégradé et des ombres. Tout d'abord, nous allons réorganiser le menu afin de rassembler la partie vidéo et photos dans un menu **médias**. Cela veut dire qu'il va y avoir un sous menu (ou plus si vous souhaitez réorganiser différemment) contenant les vidéos et les images.
 - Le sous menu sera seulement affiché lorsqu'on passe le curseur sur le menu parent (ie: lorsque l'on passe sur le menu **médias**, on affichera le sous menu en dessous contenant les **images** et les **vidéos**). (Utilisez la pseudo-classe **hover**).
 - Mettez ensuite un gradient linéaire (à vous de choisir les couleurs) en fond de votre menu et sous menu (**linear-gradient**).
 - Ajoutez une ombre à votre menu (**box-shadow**).
 - Pour séparer les différents titres de votre menu, mettez une bordure droite d'1px de la couleur de votre choix à chaque titre sauf le dernier (utilisez **:not(:last-child)**, une autre solution est d'utiliser le signe +...).

- **L'aside :**

- Arrondissez les bords de l'élément **aside** contenant une liste sur votre page d'accueil. A vous de voir si vous souhaitez mettre le même arrondi pour tous les bords (**border-radius**).

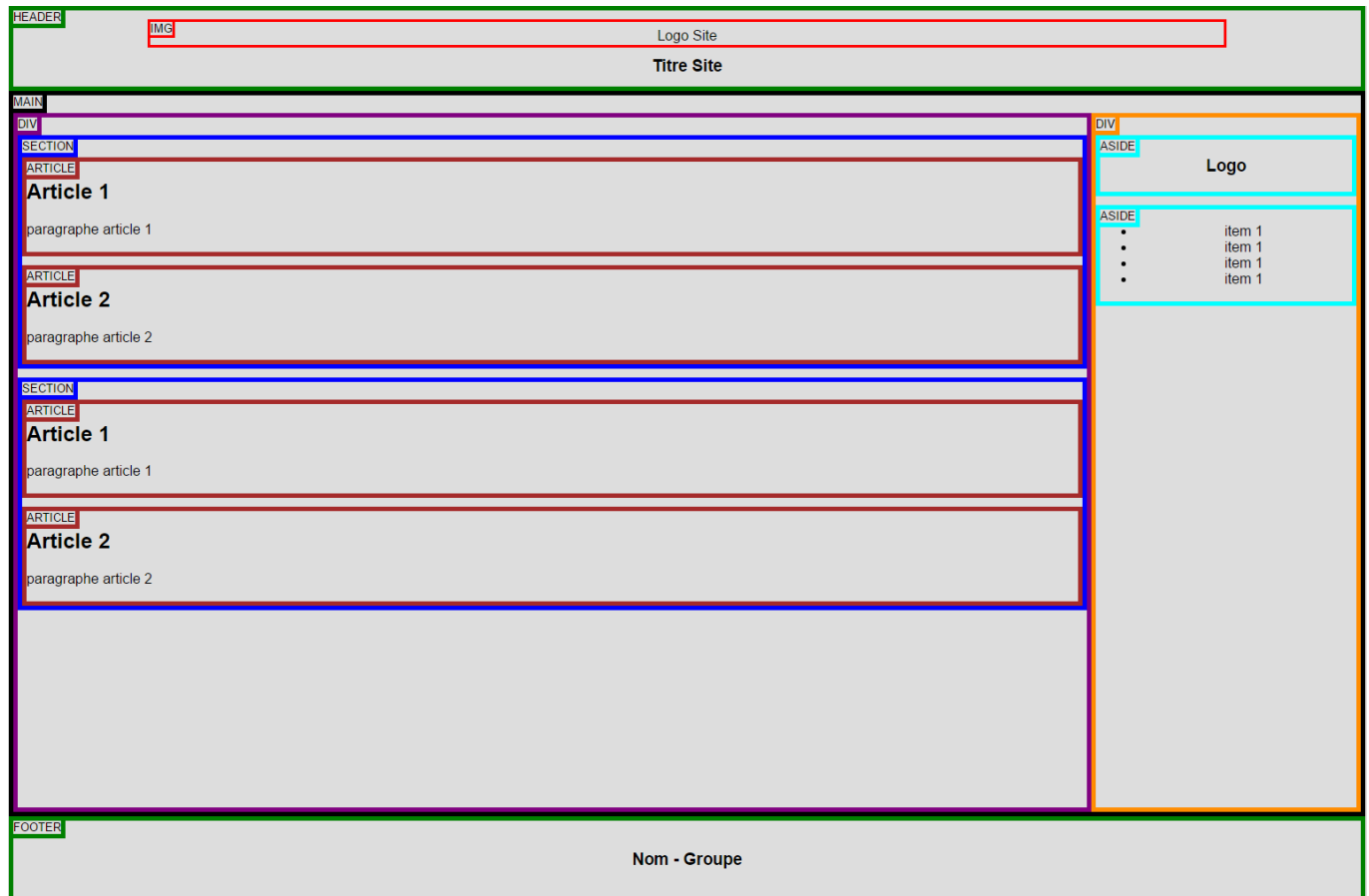
- **Flexbox**

- Jusqu'à maintenant, votre site utilisait une structure avec du positionnement classique pour vos éléments HTML (**display : block, inline-block, float, position**). Vous allez changer le positionnement pour les éléments HTML qui composent la structure globale de toutes vos pages, il est temps d'utiliser les flexbox. Actuellement, votre structure ressemble normalement à ceci :

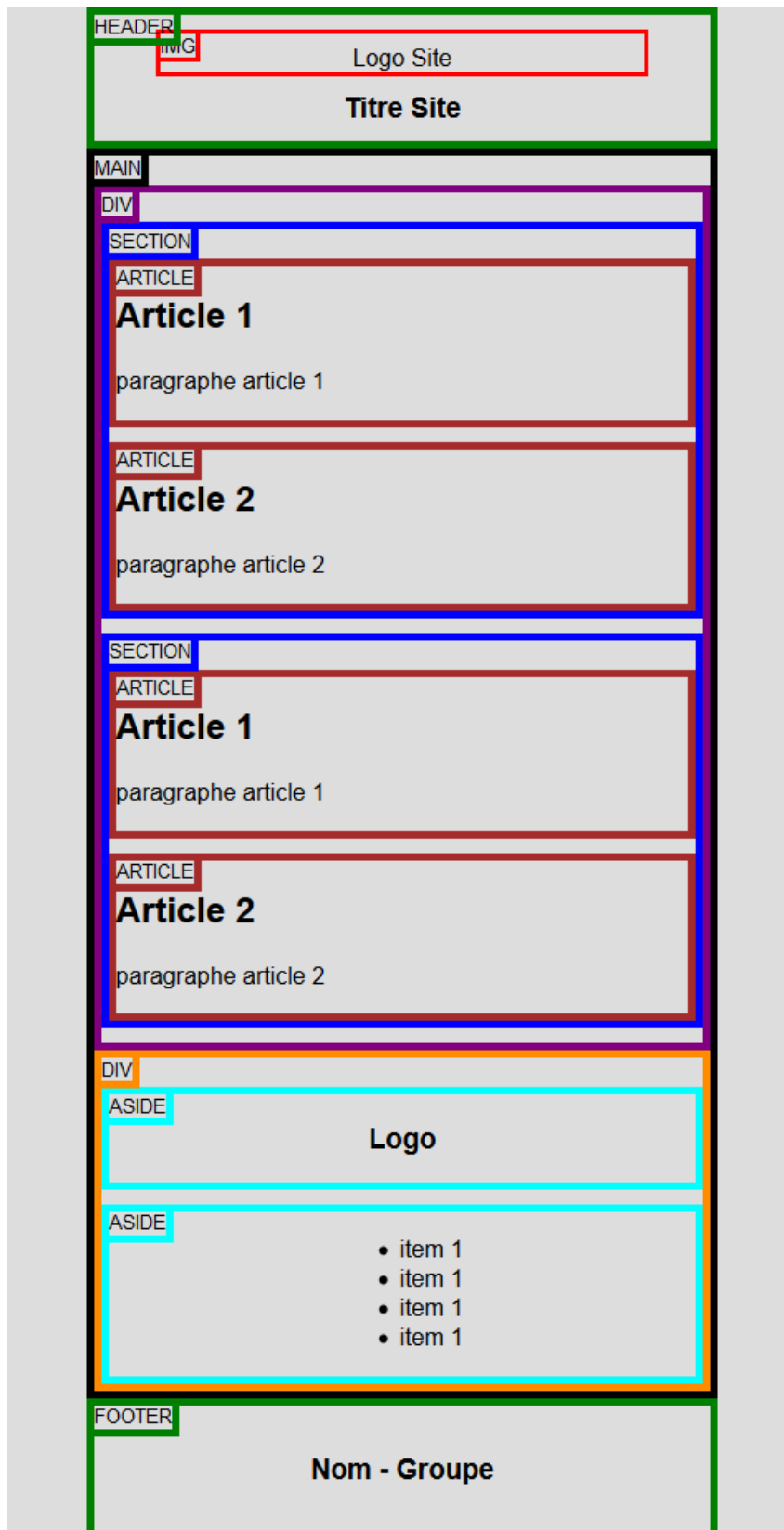


- Sur les images, chaque container possède des bordures de couleur différente afin de les distinguer, bien entendu vos container n'ont pas besoin de posséder de bordure...
Comme on peut le constater, si le contenu de votre site ne prend pas toute la hauteur de votre page, le **footer** ne se trouve pas en bas de la page. Il est possible de forcer le **footer** en bas de la page peu importe le contenu de votre site. Pour cela nous allons utiliser les flexbox. Vous allez devoir modifier votre CSS pour votre structure afin de les utiliser.

(NB : Afin de garder une trace du code CSS précédemment utilisé, commentez les lignes de CSS qui ne sont plus utiles). Sans avoir besoin de modifier votre code HTML (si votre site respecte la structure HTML de la capture d'écran), vous devez arriver au résultat suivant :



- Avant de commencer à coder, visualisez quels éléments doivent être de type **flex**, quelle direction faut-il leur attribuer et ainsi utiliser le bon axe d'alignement... (**body** est aussi un élément HTML...). Au sein des containers en **flex**, qui doit pouvoir s'agrandir afin de prendre l'espace restant ?
- **Vous avez dit responsive ?**
 - Dans notre page d'accueil il y a notre **division** contenant les deux **sections** d'actualités et notre **division** contenant les **asides**. On peut constater qu'en réduisant la largeur de la fenêtre, les **sections** et les **asides** vont se réduire, mais sur des petits écrans la lecture des articles ne sera pas aisée au vu de la largeur du container. Utilisez les **media-queries** pour que lorsque la largeur de l'écran passe en dessous des **800px** la **division** contenant les **asides** s'affiche en dessous de la **division** des deux **sections**, ainsi les **articles** dans les **sections** prendront toute la largeur de l'écran, et les deux **asides** se trouveront en dessous prenant également toute la largeur de l'écran. Pour résumer, voici le résultat obtenu :



- **La police (d'écriture bien entendu...)**
 - Vous voulez rendre votre site un peu plus unique ? Quoi de mieux que de mettre une police qui correspond au thème du site.

Choisissez une police sur un site type **fontsquirrel.com** et appliquez la à votre site (**@font-face**).

- Essayez de choisir une police avec le plus de compatibilité au niveau des navigateurs (autrement dit chargez les fichiers nécessaires pour que la police fonctionne sur plusieurs navigateurs).
- Si vous ne trouvez pas votre bonheur vous pouvez également piocher dans la liste des **Google Fonts**, l'intégration de ces polices ne nécessite pas de téléchargement.

- **Si vous avez le temps...**

- Pour utiliser les pseudo-éléments, dans la page des images, faites en sorte que le titre de l'image soit affiché en dessous de l'image lorsque vous la survolez. Rappel, vos images se trouvent dans un lien, il faudra que l'attribut titre soit dans la balise de votre lien. (Utilisez **:hover :after**, avec un **content :attr(title)**)