

01. JavaScript Drum Kit



Simulez une page de batterie.

Lorsque l'utilisateur appuie sur les touches ASDFGHJKL du clavier, les boutons correspondant aux lettres de la page deviennent plus grands et plus lumineux, et le son de batterie correspondant retentit.

Points clés à retenir

1. Événements clavier
2. Jouer le son
3. Changer de style

Décomposition par étapes

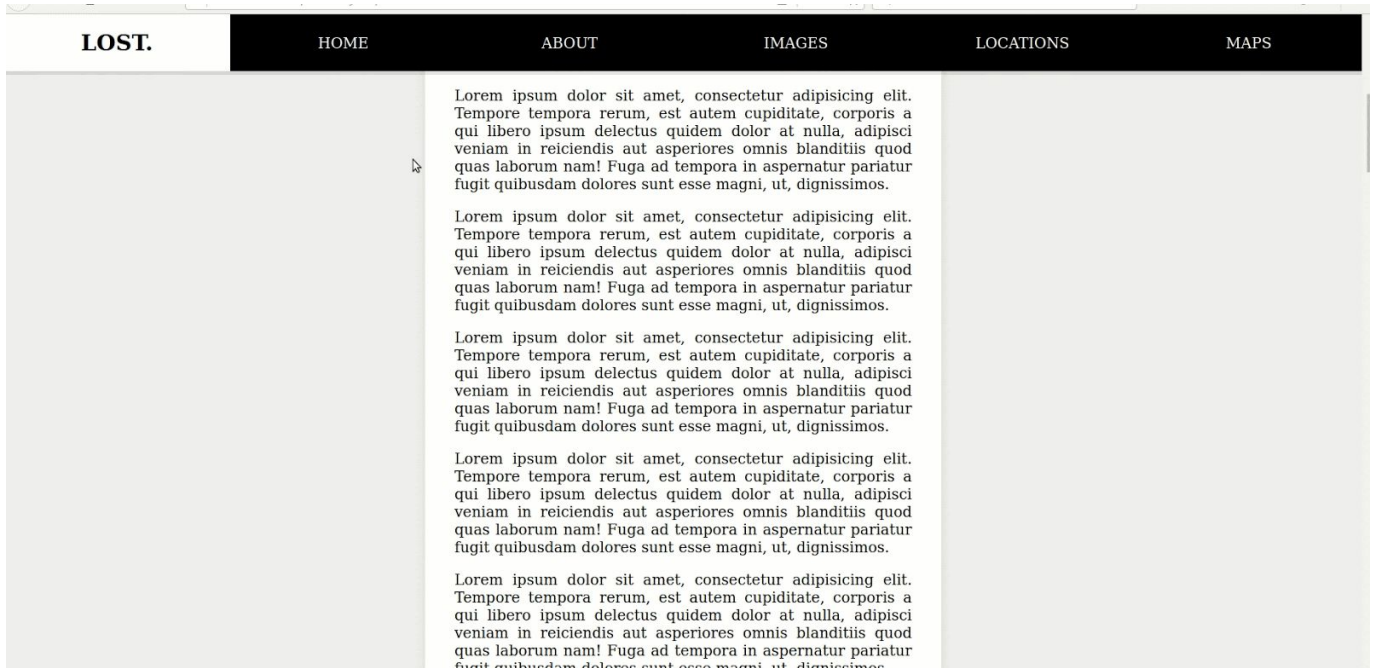
1. Ajouter un écouteur d'événements clavier `.keydown`
2. Créer une fonction déclenchant l'audio pour la data-key correspondante.
3. Ajouter la class `.playing` à la div `.key` activée.
4. Ajouter un gestionnaire d'événements pour enlever le style à la fin de la transition (cf : dans le css, la class `.key` contient une propriété transition de `.0.07s`).

Comment mapper les touches du clavier aux boutons de page?

Comment s'assurer que lorsque la touche est maintenue enfoncée, un son de batterie continu peut être joué immédiatement ?

Comment restaurer les boutons de la page ?

24. Sticky nav



Aujourd'hui, nous créons une barre de navigation qui est placée statiquement sous une image d'en-tête.

Ensuite, lorsque l'utilisateur fait défiler vers le bas et que la barre de navigation frappe le haut de l'écran, il doit passer à un emplacement fixe en haut du navigateur.

Maintenant que nous savons ce que nous voulons réaliser, nous pouvons commencer.

1. Obtenez la position de la barre de navigation par rapport à l'ensemble du document et enregistrez-la

- `offsetTop`

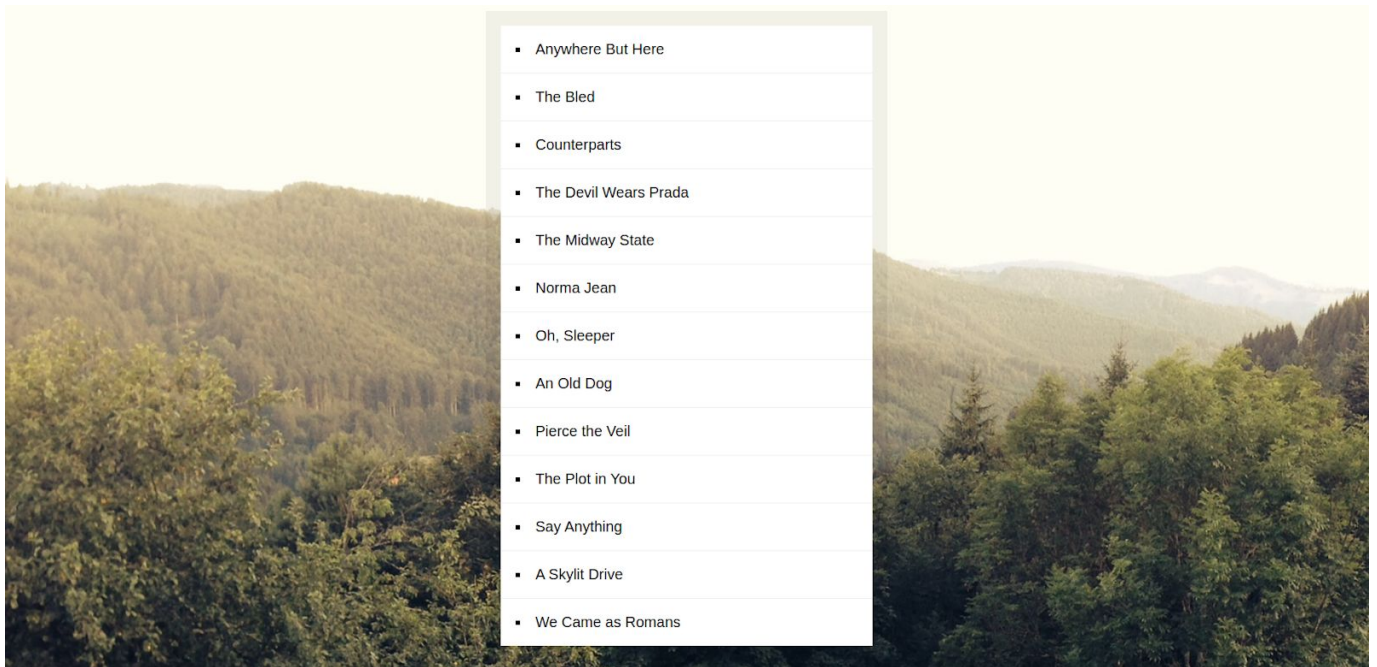
2. Écrivez une fonction qui change dynamiquement le style de la barre de navigation

- `max-width` de la nav
- ajout de la class pour fixer le logo

3. Liez la fonction à l'événement de défilement de la fenêtre

- `window.scroll`

17 : Sorting Band Names without articles



Un élément de liste non ordonné est fourni dans le fichier index.html et un tableau de chaînes est fourni dans le script.

Pouvez-vous trier cette liste ?

Cela nécessite la suppression de certain préfixe de la chaîne : The, A et An puis de trier les phrases par ordre alphabétique

Afficher le résultat classés.

5 : Galerie de panneaux / Flex

Cliquez sur une image pour l'agrandir et déplacez le texte du haut et du bas de l'image.

Après avoir cliqué sur l'image agrandie, l'image est réduite et le texte en haut et en bas de l'image est compressé.

Dans les styles CSS correspondants, la durée de l'animation et d'autres caractéristiques ont été définies.

Il vous suffit de terminer la mise en page et la surveillance des événements dans différents états.

Fonctionnalités impliquées

- affichage: flex
 - flex-direction
 - justifier-contenu
 - align-items
- transformer: translateX / translateY
- `element.classList.toggle()`
- événement `transitionend`

02 : JS + CSS Clock

Le document HTML a été donné, il y a trois cadrans et les trois pointeurs correspondant. Ajoutez simplement des effets CSS et mettez à jour l'état du pointeur dynamiquement avec JavaScript.



Points clés à retenir

1. Style de pointeur sur le cadran: effet de rotation
2. Obtenez en temps réel
3. Changer l'état du pointeur toutes les secondes

Fonctionnalités impliquées:

- transform-origin
- transform: rotate()
- transition
- transition-timing-function: cubic-bezier(x, x, x, x)
- setInterval(callback, time)
- new Date()

12 : Key sequence detection

Tout d'abord, qu'est le CODE KONAMI ? Il s'agit d'une séquence de boutons qui sont utilisés pour déverrouiller des œufs de Pâques ou activer un code de triche (#sims).

Vous pouvez en savoir plus sur le [Code Konami](#) mais c'est une séquence comme: `↑ ↑ ↓ ↓ ← → ← → B A`

La documentation initiale fournit un script de [Cornify.com](#)

Appeler `cornify_add()` va insérer une icône dans le DOM.

Le but de ce défi est de générer un nouvel effet Cornify lorsque le "Mot de passe", une chaîne de caractère prédéfinies, est découverte.



1. Spécifiez une chaîne qui déclenche des effets spéciaux
2. Ecouter l'événement `keydown`
3. Capturer toutes les `key`, nous les stockons dans un tableau et les comparons à notre code secret.
4. Nous devons modifier le tableau avec la méthode `splice` lorsqu'il atteint la longueur maximale de notre code secret afin de ne pas créer par inadvertance un enregistreur de frappe.
5. Si nous trouvons une correspondance entre le code secret, nous pouvons utiliser le script Cornify pour afficher des licornes et des arcs-en-ciel aléatoires sur la page.

10 – Hold Shift and Checkboxes

<input checked="" type="checkbox"/>	This is an inbox layout.
<input checked="" type="checkbox"/>	Check one item
<input checked="" type="checkbox"/>	Hold down your Shift key
<input type="checkbox"/>	Check a lower item
<input type="checkbox"/>	Everything in between should also be set to checked
<input type="checkbox"/>	Try do it without any libraries
<input type="checkbox"/>	Just regular JavaScript
<input type="checkbox"/>	Good Luck!
<input type="checkbox"/>	Don't forget to tweet your result!

Quand une case à cocher est sélectionnée, le texte du paragraphe apparaîtra barré.

L'effet net est de fournir une opération de sélection multiple après avoir appuyé sur la touche Maj.

Guide de processus

Obtenez tous `<input>` les éléments, et ajouter des écouteurs d'événements

```
const boxes = document.querySelectorAll('.inbox input[type="checkbox"]');
```

1. `boxes.forEach(box => box.addEventListener('click', handleCheck));`
2. Écrivez la logique de traitement dans `handleCheck`

Résoudre des idées

Tout d'abord, que se passe-t-il lorsque vous appuyez sur la touche Maj pour effectuer plusieurs sélections ? Le point clé est que A et B délimitent une plage.

1. Vérifier l'élément A
2. Shift
3. Vérifiez à nouveau l'élément B
4. Tous les éléments entre AB sont sélectionnés

16 - CSS Text shadow / Mouse Move Effect

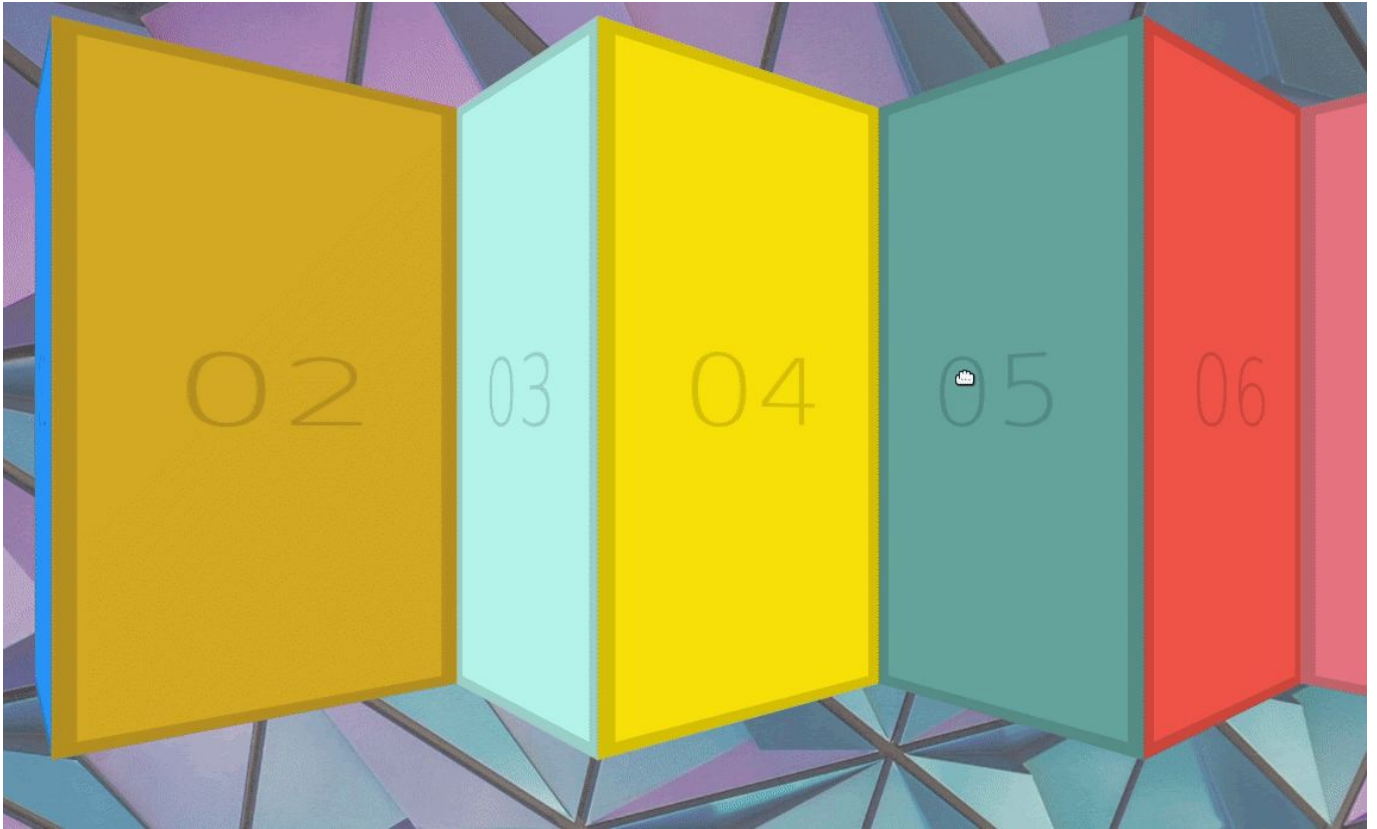


L'objectif est de faire en sorte que l'ombre du texte de notre en-tête suive la souris autour de la fenêtre.

Point de connaissance

- `text.style.textShadow`
- `mousemove`

27 -Click and drag to scroll



Indices pour atteindre l'effet souhaité :

Détecter les événements de souris

Obtenir les données initiales (est cliqué, x initial, défilement Y initial)

Faire glisser la piste

Mettre à jour la position du défilement

13 – Slide in on Scroll

veniam in reiciendis aut asperiores omnis blanditiis quod
quas laborum nam! Fuga ad tempora in aspernatur pariatur
fugit quibusdam dolores sunt esse magni, ut, dignissimos.

Lorem ipsum dolor sit amet, consectetur adipisicing elit.
Tempore tempora rerum, est autem cupiditate, corporis a
qui libero ipsum delectus quidem dolor at nulla, adipisci
veniam in reiciendis aut asperiores omnis blanditiis quod
quas laborum nam! Fuga ad tempora in aspernatur pariatur
fugit quibusdam dolores sunt esse magni, ut, dignissimos.

Lorem ipsum dolor sit amet, consectetur adipisicing elit.
Tempore tempora rerum, est autem cupiditate, corporis a
qui libero ipsum delectus quidem dolor at nulla, adipisci
veniam in reiciendis aut asperiores omnis blanditiis quod
quas laborum nam! Fuga ad tempora in aspernatur pariatur
fugit quibusdam dolores sunt esse magni, ut, dignissimos.

Lorem ipsum dolor
sit amet,
consectetur
adipisicing elit.
Voluptates,
deserunt facilis et
iste corrupti omnis
tenetur est. Iste ut
est dicta dolor
itaque adipisci,
dolorum minima,
veritatis earum

Point de connaissance

Propriétés CSS impliquant le contrôle des images:

- translateX Pour contrôler les mouvements gauche et droit
- scale Pour contrôler le zoom

Attributs impliquant la taille de la page:

- window.scrollY Le nombre de pixels du document a défilé du haut
- window.innerHeight viewport Hauteur pièce
- ele.height Hauteur d'élément
- ele.offsetTop Distance entre le haut de l'élément actuel et le haut de son élément offsetParent.

Pour réduire la fréquence de surveillance des événements, utilisez la méthode anti-rebond.

Résoudre des idées

1. Obtenez tous les éléments d'image dans la page
2. Écouteur d'événement de défilement
3. Acquisition et traitement des dimensions
4. Jugement de condition pour faire défiler jusqu'à la zone spécifiée