

Exercice 1 (10min) : Questions

1. Les variables existent en Javascript mais n'existent ni en HTML ni en CSS. Vrai ou Faux ?

Faux plutôt, si les variables sont propres aux langages de programmation et non au langage de balisage, il est possible d'utiliser des variables en CSS via les « custom properties »

2. Quelle est la différence entre ces différentes méthodes de stockage (Cookie, SessionStorage & LocalStorage) ?

La principale différence entre cookies, sessionStorage et localStorage, réside dans la durée de vie des données stockées et leur accessibilité. Les cookies ont une durée de vie limitée et sont envoyés au serveur avec chaque requête HTTP, tandis que sessionStorage et localStorage ont une durée de vie plus longue et ne sont pas automatiquement envoyés côté serveur.

3. Quelles sont les règles CSS qui doivent être ajoutées pour appliquer un ellipsis sur un texte (réduire dynamiquement la longueur d'un texte afin qu'il ne dépasse pas son container et sans qu'il n'aille à la ligne) ?

width: 300px;

white-space: nowrap; / empêche le texte d'aller à la ligne */*

overflow: hidden; / cache le texte débordant du conteneur */*

text-overflow: ellipsis; / affiche "..." pour indiquer qu'il y a du texte caché */*

4. Vous êtes face à une liste de 1000 éléments () pour lesquels chacun de ces éléments déclenche au clic une alerte avec le texte du contenu. Comment et pourquoi optimisez-vous l'écoute d'événement dans ce cas ?

On utilise la délégation en appliquant au parent une écoute d'évènement, plutôt que de définir l'action sur chacun des LI. Cela optimise le code.

5. Quelles sont les différences entre les types de variable suivants : var, let et const ?

Les variables var et let sont sensiblement pareilles, mais la portée du let (existant depuis ES6) est préférable quand on l'a cantonnée à un corps de fonction par exemple, elle est plus locale, là où var peut bien être utilisée de façon globale sur toute une application. La constante avec const enfin, stocke également des valeurs, mais de façon constante, il est impossible de réassigner leur référence par la suite dans le code.

6. Quels sont les différents avantages à l'utilisation des outils de développement "Performances" et "Mémoire" dans les navigateurs Firefox et Chromium ?

Les outils de développement "Performances" et "Mémoire" dans les navigateurs Firefox et Chromium offrent des fonctionnalités avancées pour aider les développeurs à améliorer les performances, à identifier les problèmes de code, à déboguer plus facilement et à améliorer la stabilité de l'application. Ces outils sont indispensables pour les développeurs web professionnels, notamment pour des raisons de SEO.

7. Quelle a été votre tâche la plus ardue face à laquelle vous vous êtes confronté au cours de votre expérience professionnelle ?

Curieusement ou pas, la tâche la plus ardue a été de réaliser une popup qui peut apparaître sur toute l'application, au bout d'un certain temps d'inactivité, et qui nous déconnecte de force après un autre délai avec des setInterval, useContext (React) et autres.

8. Quelle est la dernière chose que vous avez apprise récemment ?

Le git rebase afin de travailler en équipe avec gitHub, les références de branche et un historique sans toucher à la production avec ce qu'on appelle une Code review.

9. Cette année, quelle techno aimeriez-vous apprendre et utiliser ?

J'aimerais me parfaire dans l'univers JavaScript que ce soit en vanilla, avec une librairie ou différents framework.