Support de soutenance

Bonjour à tous,

Permettez-moi de vous présenter le projet que j'ai réalisé pour l'imprimerie familiale Print It. Print It est une entreprise familiale spécialisée dans l'impression personnalisée sur une variété de supports tels que cartes, flyers, affiches, et plus encore. Fondée il y a plusieurs décennies, Print It a su se démarquer par la qualité de ses impressions et son service clientèle exemplaire. Dans le cadre de ce projet, mon rôle était de moderniser le site internet de Print It en remplaçant une bannière statique par un carousel interactif. Cette mise à jour visait à offrir aux visiteurs du site une expérience plus attrayante et dynamique, mettant en valeur le savoir-faire de l'entreprise et sa capacité à répondre aux besoins variés de sa clientèle.

1. \*\*Ajout du code HTML et CSS :\*\*  
  - J'e n’ai pas eu a ajouté les balises HTML nécessaires pour créer la structure du carousel.  
  - J'ai stylisé le carousel en utilisant du CSS déjà fourni afin de mettre les flèches pour lui donner un aspect visuel attrayant et en adéquation avec l'identité de Print It.

2. \*\*Gestion des événements avec les Event Listeners :\*\*  
  - J'ai mis en place des Event Listeners sur les flèches de navigation du carousel.  
  - J'ai veillé à différencier le clic sur la flèche de gauche de celui sur la flèche de droite pour déclencher les actions correspondantes.

3. \*\*Affichage des bullet points :\*\*  
  - J'ai ajouté des indicateurs visuels sous forme de bullet points pour signaler la position de l'image actuellement affichée dans le carousel.  
  - J'ai veillé à ce que le bullet point correspondant à l'image en cours soit mis en évidence.

4. \*\*Changement dynamique des images et du texte associé :\*\*  
  - J'ai programmé le carousel pour qu'au clic sur les flèches de navigation, l'image et le texte associé changent de manière fluide.  
  - J'ai veillé à ce que le texte affiché corresponde à l'image en cours de visualisation dans le carousel.

5. \*\*Défilement infini :\*\*  
  - J'ai mis en place un mécanisme de défilement infini pour le carousel.  
  - J'ai géré les cas particuliers du clic sur la flèche de gauche pour la première image, et sur la flèche de droite pour la dernière image, pour assurer une expérience utilisateur fluide.Chacune de ces étapes a été cruciale pour le bon fonctionnement du carousel et pour offrir aux visiteurs du site une expérience interactive et engageante.

Dans ce code, j'ai mis en place un carousel qui va faire défiler des images lorsque l'utilisateur clique sur des flèches de navigation.

Les étapes de travail :

Ce code est organisé de manière à ce que chaque fonction ait une responsabilité spécifique :

La fonction main() initialise le carousel en appelant les autres fonctions nécessaires.

Les fonctions prevSlide() et nextSlide() sont responsables de la navigation entre les diapositives.

La fonction initDots() initialise les points indicateurs (dots) du carousel.

La fonction updateSlide() met à jour l'image et la légende de la diapositive affichée.

Chaque fonction est nommée de manière explicite pour indiquer ce qu'elle fait, ce qui rend le code facile à comprendre pour d'autres développeurs. De plus, les commentaires sont utilisés pour expliquer les étapes importantes du code.

Tout d'abord, j'ai récupéré les éléments HTML des flèches de navigation, des éléments d'affichage des images et des légendes. Ensuite, j'ai aussi récupéré un conteneur pour les points indicateurs du carousel.

Cette fonction initialise les points indicateurs (dots). Elle boucle à travers chaque élément du tableau slides et crée un élément <span> pour chaque diapositive, lui ajoute la classe "dot", puis l'ajoute au conteneur des points indicateurs (dotsContainer). Ensuite, elle récupère tous les éléments span ayant la classe .dot dans le conteneur dotsContainer et les stocke dans allDots. Enfin, elle ajoute la classe "dot\_selected" au premier point indicateur pour indiquer quelle diapositive est actuellement affichée.

Ensuite, j'ai initialisé les points indicateurs dynamiquement en fonction du nombre de diapositives pour permettre de visualiser quelle image est affichée dans le carousel.

J'ai ensuite attaché des écouteurs d'événements aux flèches pour permettre de changer de diapositive en fonction du clic de l'utilisateur.

Cette fonction ajoute un écouteur d'événement de clic à la flèche gauche (leftArrow). Lorsque cette flèche est cliquée, elle enlève la classe "dot\_selected" de l'élément point indicateur actuel, décrémente l'index de la diapositive courante (currentSlideIndex), vérifie si l'index est inférieur à zéro (si oui, il revient au dernier élément), puis appelle la fonction updateSlide() pour mettre à jour l'affichage de la diapositive.

Cette fonction est similaire à prevSlide(), mais elle est attachée à la flèche droite (rightArrow). Lorsque cette flèche est cliquée, elle enlève la classe "dot\_selected" de l'élément point indicateur actuel, incrémente l'index de la diapositive courante (currentSlideIndex), vérifie si l'index dépasse la longueur du tableau de diapositives (si oui, il revient à zéro), puis appelle la fonction updateSlide() pour mettre à jour l'affichage de la diapositive.

Cette fonction met à jour l'affichage de la diapositive en mettant à jour l'attribut src de l'élément imageSlide avec le chemin de l'image de la diapositive courante, en mettant à jour le contenu HTML de l'élément tagLine avec le sous-titre de la diapositive courante, et en ajoutant la classe "dot\_selected" à l'élément point indicateur correspondant à la diapositive courante.

Pendant la réalisation de ce projet, j'ai rencontré certaines difficultés. Par exemple, j'ai dû comprendre comment manipuler les classes CSS pour changer l'apparence des flèches et des bullets points. J'ai également dû trouver une solution pour adapter le code afin qu'il fonctionne correctement. Ces défis m'ont permis d'approfondir ma compréhension du langage JavaScript et des principes de base de la programmation.[Conclusion]  
Ce projet a été une excellente opportunité pour moi d'explorer et de pratiquer les fondamentaux de JavaScript dans un contexte réel. J'ai appris à construire un carousel interactif à partir de zéro, en utilisant les fonctionnalités de base du langage JavaScript. Je suis satisfait du résultat obtenu et je suis prêt à répondre à vos questions.Merci pour votre attention.

Explication détaillée des termes importants en JavaScript :

const slides : C'est un tableau contenant des objets représentant chaque diapositive du carousel, avec des propriétés telles que "image" (chemin vers l'image) et "tagLine" (sous-titre de l'image).

let et const : J'utilise ces mots-clés pour déclarer des variables en JavaScript. "const" définit une constante (dont la valeur ne peut pas être modifiée) et "let" définit une variable dont la valeur peut être mise à jour.

Arrow Functions : Ce sont des fonctions anonymes plus succinctes et simplifiées que les fonctions traditionnelles en JavaScript.

querySelector : C'est une méthode permettant de sélectionner un élément du DOM en fonction de son sélecteur CSS.

addEventListener : Cette méthode permet d'ajouter un écouteur d'événement à un élément HTML, ce qui permet de réagir à des interactions de l'utilisateur comme un clic.

forEach : Il s'agit d'une méthode d'itération sur un tableau pour exécuter une fonction sur chaque élément du tableau.

innerHTML : C'est une propriété qui permet de définir ou récupérer le contenu HTML d'un élément.

classList : C'est une propriété qui permet de gérer les classes d'un élément du DOM, comme ajouter ou supprimer une classe.

Question 1 : Pourquoi avez-vous choisi de dynamiser la bannière du site avec un carousel ?  
Réponse : J'ai choisi d'utiliser un carousel car cela permet d'afficher de manière attrayante plusieurs images différentes sur une même partie du site. Cela permet également de mieux mettre en avant les produits et services de l'imprimerie familiale.

Question 2 : Pourquoi avez-vous utilisé JavaScript pour réaliser le carousel plutôt qu'une librairie existante ?  
Réponse : Dans le cadre de ce projet, nous devions être capables de programmer en utilisant les fondamentaux de JavaScript, c'est pourquoi j'ai réalisé le carousel manuellement, sans utiliser de librairie. Cela me permet également de mieux comprendre le fonctionnement interne du carousel et d'avoir unôle total sur ses fonctionnalités.

Question 3 : Comment avez-vous difféié le clic sur la flèche de gauche et celui sur la flèche de droite ?  
Réponse : J'ai ajouté des Event Listeners aux deux flèches de navigation et à chaque clic, je vérifie la classe HTML de la flèche qui a été cliquée. En de cette classe, j'ai programmé les actions spécifiques pour passer à la slide précédente ou à la slide suivante.

Question 4 : Comment avez-vous signalé la position de l'image actuellement affichée ?  
Réponse : J'ai ajouté des bullets points en bas du carousel grâce à du code HTML et CSS. J'ai utilisé des classes spécifiques pour indiquer le slide actuel et les autres slides. Au fur et à mesure que le carousel défile, je change dynamiquement les classes pour que le bon bullet point soit signalé.Question 5 : Comment avez-vous géré le défilement infini des images dans le carousel ?  
Réponse : J'ai pris en compte les cas particuliers du clic sur la flèche de gauche pour la première diapositive et celui de la flèche de droite pour la dernière diapositive. Lorsque l'utilisateur clique sur une flèche, je vérifie si l'image suivante ou précédente est la première ou la dernière. Dans ce cas, je fais apparaître respectivement la dernière ou la première image, créant ainsi l'effet de défilement infini.