

RAPPORT DU PROJECT DAW

Membre du groupe :

Hani Mohamed Ishak Section A groupe 4.

Objectif du theme :

Le theme choisi est le E-Commerce ,pour preciser un site d'une petite entrprise qui vend les sneakers ou bien les chassures sportifs.l'objectif derriere la conception de ce site et facilite la tache au client

Language utilises:

_HTML,CSS,JAVASCRIPT,PHP ,SQL(BASE DE DONNEES)

CONCEPTION

HTML

1-Pour commencer on a créer notre barre de recherche avec deux branche ,” navTop ”qui contient l’image du logo du site ,la barre de recherche avec l’image de recherche et pour finir le petit paragraphe contenant ”Limited offer”.Et pour chacune de ces Article a son propre ”div”.

On passe vers le ”navBottom” , on trouve des titre de taille ‘h3’ conteneant les noms de produit

2-On passe vers l’esapce allouer pour les images des produits sous classe ‘slider’,qui contient un ‘div’ qui contient 5 div de classe ‘sliderItem’.chaque div de ces dernier possede une image ,un div vide pour modifier le background,un titre ,un prix ,et un lien sous forme de bouton qui nous transfere vers une partie basse de notre site

3-L'espace allouer pour les caractéristiques de site ou la classe est 'Features'. Il dispose de 4 div de la même classe 'feature', où chacune a un icône, un titre, et une description.

4-on passe vers la partie concernant les produits et les informations sur les produits. On a un div principale de classe 'product' et un id 'product' qui possède une image. Et se compose des mêmes images concernant le 'slider', un autre div de classe 'productDetails' contenant un titre du produit, un prix, une description, un div pour les couleurs de chaque produit, un div pour la taille du produit, un bouton pour l'achat. Un formulaire pour le paiement qui contient un titre, un label pour le nom et le prénom, un tag de 'input' pour marquer ces derniers, un espace pour le numéro de téléphone avec son tag 'input', et on répète avec une Adresse. Et pour finir notre formulaire on ajoute des icônes de la carte de paiement et un input pour entrer le numéro de carte avec un input de type password. Et à la fin un bouton pour soumettre les informations marquer et un span contenant la lettre 'X' pour fermer le formulaire.

5-On passe vers le div de la classe 'gallery' qui contient 3 div de même classe 'galleryItem' qui affiche les images et les exemples des produits sur les modèles où chaque article possède un titre qui est au même temps la description de la photo

6-On passe vers l'espace de nouveautés qui se trouve dans un div de classe 'newSeason' qui possède lui-même de 2div de classe 'nsItem' qui possède une image et une passe des description et un bouton pour choisir le style du produit qui vous transfère vers la page des produits.

7-Et finalement on a le footer du site qui possède les noms des produits, les contacts de l'entreprise et tout ce qui concerne les feedback etc.

JAVA SCRIPTS

.On commence par sélectionner le premier élément de la page ayant comme classe "sliderWrapper" en utilisant la méthode suivante :

La méthode (document.querySelector)

```
-const wrapper = document.querySelector(".sliderWrapper");
```

Et on répète cette méthode mais pour sélectionner tous les éléments ayant comme classe "menuItem" :

(querySelectorAll)

```
const menuItems = document.querySelectorAll(".menuItem");
```

.On définit les produits disponibles en utilisant un tableau d'objets :

```
const products = [
  {
    id: 1,
    title: "Air Force",
    price: 119,
    colors: [
      {
        code: "black",
        img: "./img/air.png",
      },
      {
        code: "darkblue",
        img: "./img/air2.png",
      },
    ],
  },
],
```

```
{
  id: 2,
  title: "Air Jordan",
  price: 149,
  colors: [
    {
      code: "lightgray",
      img: "./img/jordan.png",
    },
    {
      code: "green",
      img: "./img/jordan2.png",
    },
  ],
},
{
  id: 3,
  title: "Blazer",
  price: 109,
  colors: [
    {
      code: "lightgray",
      img: "./img/blazer.png",
    },
    {
      code: "green",
      img: "./img/blazer2.png",
    },
  ],
},
{
  id: 4,
  title: "Crater",
  price: 129,
  colors: [
    {
      code: "black",
      img: "./img/crater.png",
    },
    {
      code: "lightgray",
      img: "./img/crater2.png",
    },
  ],
},
{
  id: 5,
  title: "Hippie",
```

```

    price: 99,
    colors: [
      {
        code: "gray",
        img: "./img/hippie.png",
      },
      {
        code: "black",
        img: "./img/hippie2.png",
      },
    ],
  },
];

```

.On affecte le premier produit du tableau a un variable ayant comme nom « chosenProduct » :

```
let chosenProduct = products[0];
```

.Sélectionner les elements des differantes classes et les assigner au differantes variables :

```

const currentProductImg = document.querySelector(".productImg");
const currentProductTitle = document.querySelector(".productTitle");
const currentProductPrice = document.querySelector(".productPrice");
const currentProductColors = document.querySelectorAll(".color");
const currentProductSizes = document.querySelectorAll(".size");

```

.On ajoute un evenement en cliquant pour chaque element de classe "menuItem"

```

menuItems.forEach((item, index) => {
  item.addEventListener("click", () => {

```

..en cliquant sur un element de la classe "menuItem" on change le slide courant du "wrapper" avec la methode

```
wrapper.style.transform = `translateX(${ -100 * index }vw)`;
```

.On reste toujours dans l'évenement de click.on affecte l'index du produit actuelle a la variable chosenProduct.

```
chosenProduct = products[index];
```

.La meme chose avec le titre du produit ,prix et la couleur:

```
currentProductTitle.textContent = chosenProduct.title;
currentProductPrice.textContent = "$" + chosenProduct.price;
currentProductImg.src = chosenProduct.colors[0].img;
```

.On Ajoute un click evenement pour chaque couleur,

```
currentProductColors.forEach((color, index) => {
  color.addEventListener("click", () => {
```

et on change l'attribut "src" de l'element "currentProductImg" par l'image de la couleur clickée:

```
currentProductImg.src = chosenProduct.colors[index].img;
```

.On ajoute un click event Listner pour chaque element "size"

```
currentProductSizes.forEach((size, index) => {
  size.addEventListener("click", () => {
    currentProductSizes.forEach((size) => {
      size.style.backgroundColor = "white";
      size.style.color = "black";
    });
    size.style.backgroundColor = "black";
    size.style.color = "white";
  });
});
```

.On recupere les premiers elements des classes (".productButton",".payment",".close")et on les affectent aux variables("productButton","payment","close")respectivement.

```
const productButton = document.querySelector(".productButton");
const payment = document.querySelector(".payment");
const close = document.querySelector(".close");
```

.On ajoute un click event listner a l'element "productButton" et on met le type de diplay pour l'element "payment" a 'flex' :

```
productButton.addEventListener("click", () => {
  //type de display pour l'element "payment"//
  payment.style.display = "flex";
});
```

.Ajouter un click event listener a l'element "close" et on change le display a 'none' en cliquant sur l'element :

```
close.addEventListener("click", () => {  
    payment.style.display = "none";  
});
```

PHP

On commence d'abord par verifier si la variable de serveur(REQUEST_METHOD) est egal a « POST » :

```
<?php  
if ($_SERVER["REQUEST_METHOD"] == "POST") {  
    $name = $_POST['name'];  
    $phone = $_POST['tel'];  
    $address = $_POST['address'];  
    $card = $_POST['card'];
```

Si la condition est vraie ,alors on recupere les valeurs depuis(name , phone , address ,card)en utilisant le tableau "\$_POST",et on les stockent dans les variables {\$name,\$phone,\$address,\$card}

.On recupere les valeurs de la requete et on les affectes au variables :

```
$sneaker = $_POST['sneaker'];  
$price = $_POST['price'];  
$color = $_POST['color'];  
$size = $_POST['size'];
```

.On passe au parametres de connexion a la base de donnees :

```
$host = 'localhost'; cette ligne affecte le string `localhost` a la variable  
$host,qui est l'adresse du serveur de la base de donnees.
```

```
$user = 'root';cette ligne affecte le sting 'root ' a la variable $user,qui le  
username utilise pour connecter a la base.
```

```
$password = '';le mot de pass utilise pour connecter a la base donnee.
```

```
$dbname = 'backend';Le nom de base a connecter avec.
```

.On doit maintenant créer un constructeur MySQLi qui utilise pour créer une connection a la base MySQL,et on le donne les parametres suivant :

```
$conn = new mysqli($host, $user, $password, $dbname);
```

.Après on doit vérifier la connexion à la base ,

```
if ($conn->connect_error) {
```

si la condition est vraie donc ,la partie du script ci-dessous va s'exécuter:

```
die("Connection failed: ". $conn->connect_error);} }
```

Cela signifie que lorsque la connexion vers la base a rencontré une erreur ,un message d'erreur sera affiché pour arrêter l'exécution.

```
. $stmt = $conn->prepare("INSERT INTO pay (nom, phone_number, adresse, card_number) VALUES (?, ?, ?, ?)");  
    $stmt->bind_param("ssss", $name, $phone, $address, $card);  
    $stmt->execute();
```

Ces lignes indiquent la préparation d'une requête SQL pour insérer les informations de paiement dans la table 'pay'. La méthode bind_param lie les variables aux paramètres de la requête.

.Après l'insertion des données,un message sera affiché pour indiquer que l'achat est soumis avec succès,ensuite la déclaration préparée et la connexion à la base seront fermées.

```
echo "Thanks for buying from us...";  
    $stmt->close();  
    $conn->close();
```

.Si la requête n'est de type 'POST' ,le script affiche le message suivant :

```
} else {  
    echo "No data received";  
}  
?>
```

BASE DE DONNEES

.On a créé une base de données nommée 'backend' et qui contient deux tables :


```
CREATE DATABASE backend
```

```
CREATE TABLE pay (  
    nom VARCHAR(100) NOT NULL,  
    phone_number int NOT NULL,  
    adresse VARCHAR(255) NOT NULL,  
    card_number int NOT NULL  
);
```

```
CREATE TABLE product(  
    sneaker VARCHAR(100) NOT NULL,  
    price int NOT NULL,  
    color VARCHAR(50) NOT NULL,  
    size INT NOT NULL  
)
```