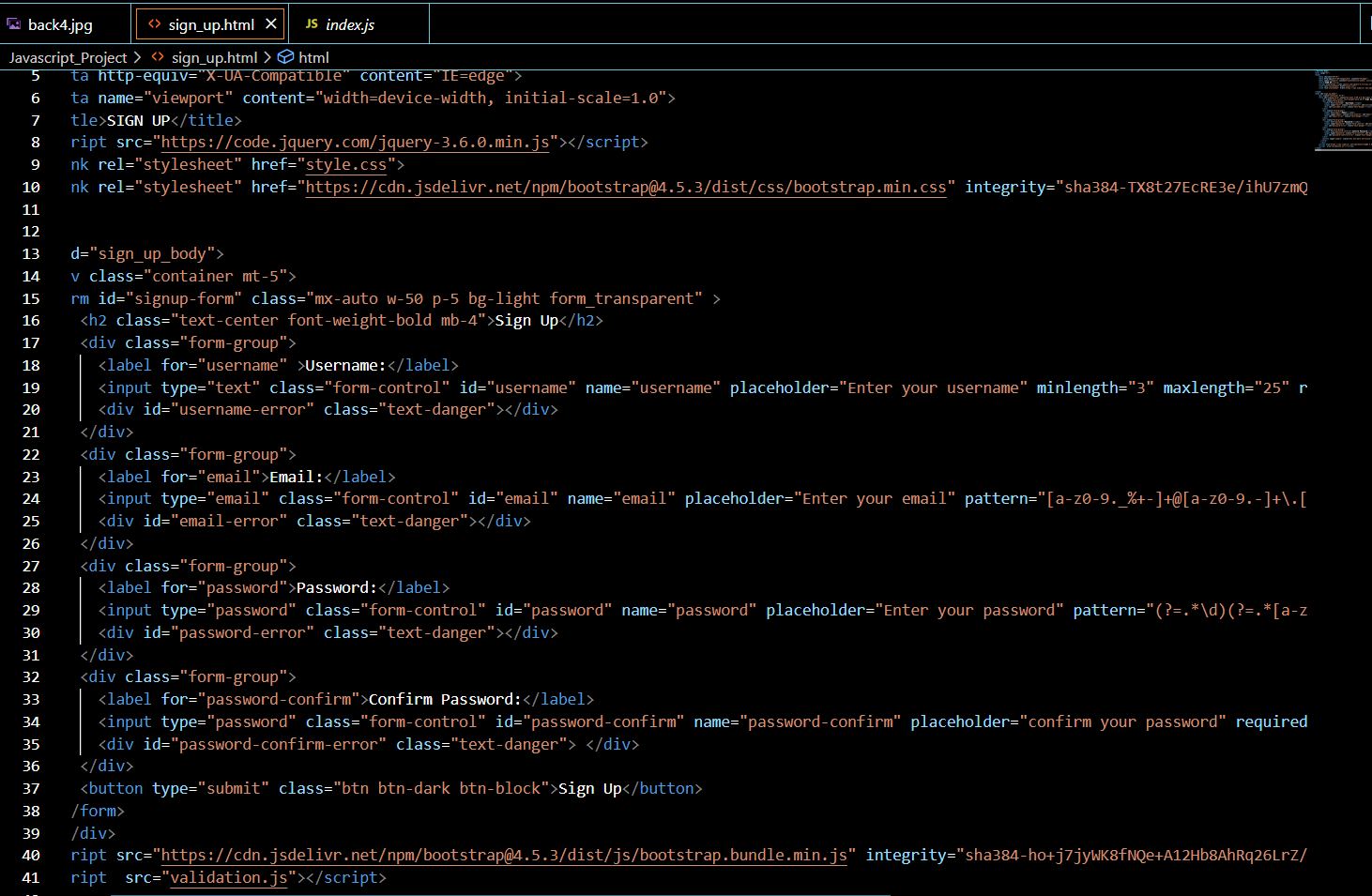
**REALISE PAR: DAIGHAM SALMA 3IIR(G3)**

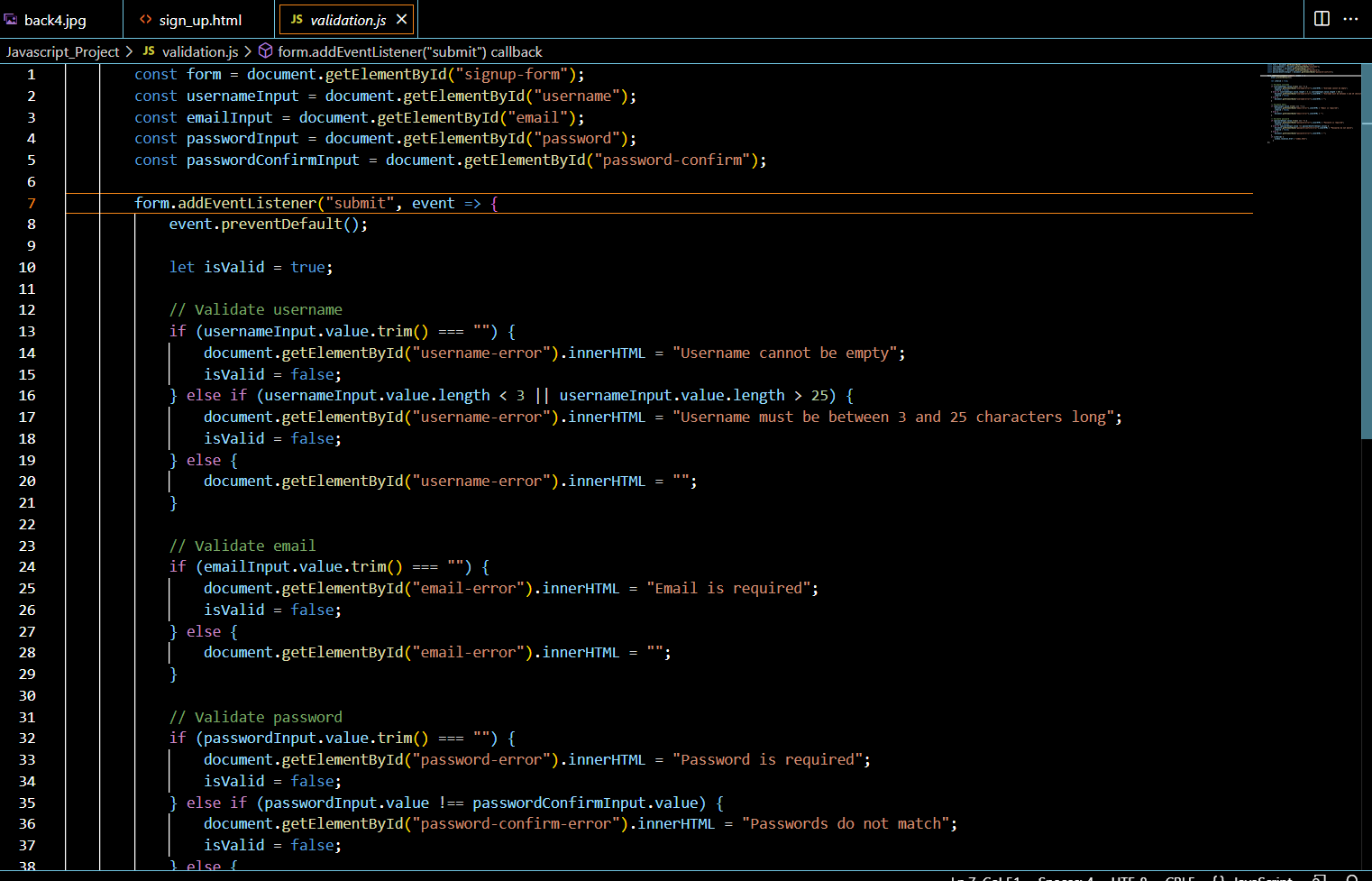
1. **SIGN\_UP.HTML:**

**Compte Rendu du miniProjet 2022-2023**



* Premièrement j’ai créer 4 div pour le “**username”,’email”,”password” et “confirm password”.**
* Chacune est attribué à une classe définie.
* Ensuite, j’ai créer un **button** de type submit appelé ‘**SIGN UP’.**
* J’ ai ajouté **patter**n dans la div de **email et password** pour respecter le format demande dans l’enoncé de ce projet
* Le **minlenght** et le **maxlenght** dans la **div username** sont ajoutés pour assurer le nombre d’element sera entre **3 et 25 caractères**.
* l’ID **"password-error",’username-error’,’password-confirm-error’, ‘email-error’**et la class "**text-danger"** placer directement après les 4 inputs, vont aider à afficher l’erreur à l’utilisateur si les inputs(password,username , confirm password,email)ne sont pas valides.
* La forme est stylé avec **Boostrap** et inclue quelque **Javascript code** qui gère la validation du formulaire:
* La validation est faite lorsqu’on click sur le button à l’aide du fichier **validation.js**

**2-Validation.js:**

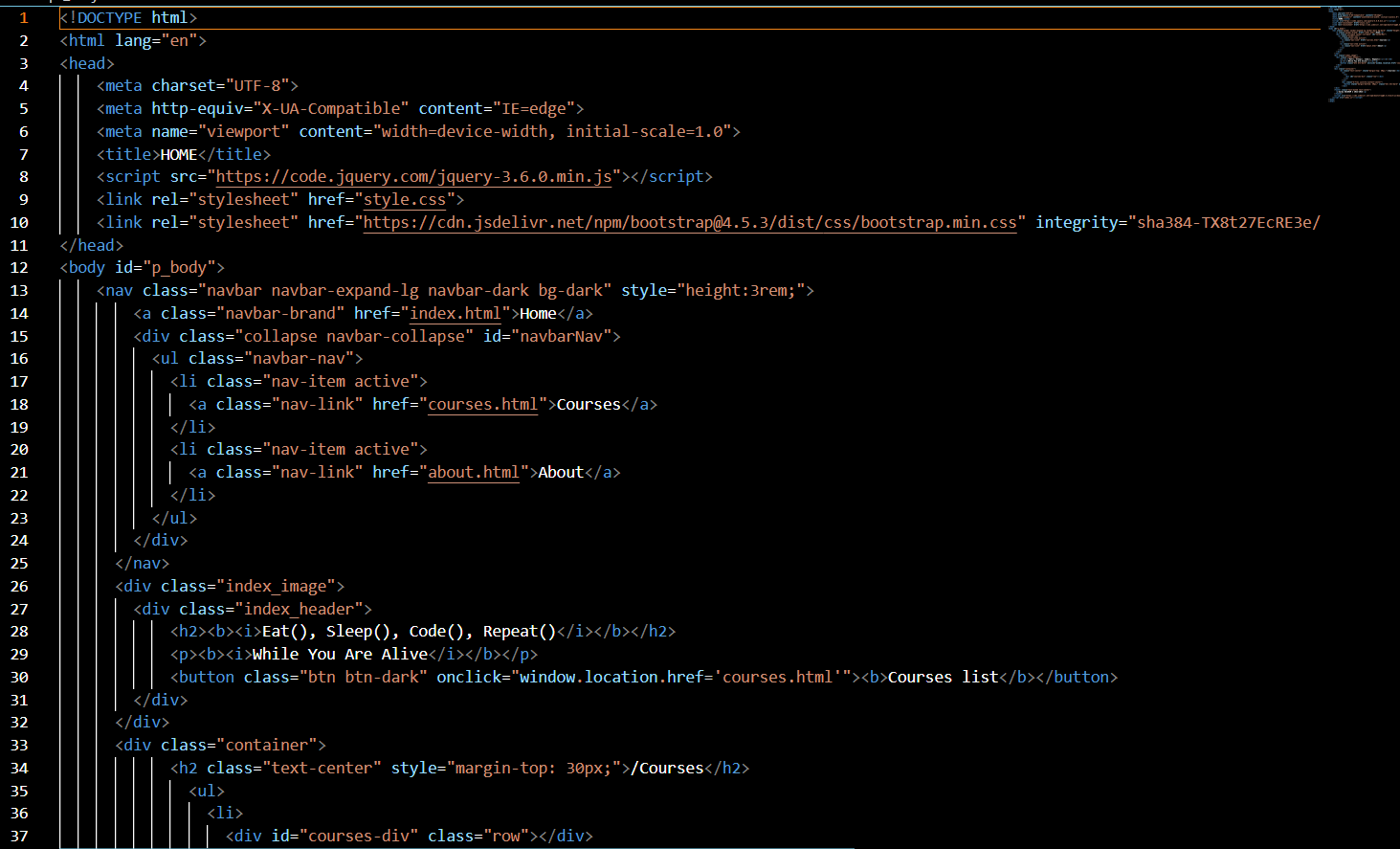
****

* La constante **form** fait reference aux élément avec ID = **“signup-form”** donc un élément de <form>.
* **usernameInput**, **emailInput, passwordInput,et passwordConfirmInput** sont des references aux éléments avec lID est **“username”,”email”,”password”,et “password-confirm”***{ les éléments des input}.*
* La méthode **event.preventDefault()** est utilisée pour empêcher le comportement de soumission de formulaire par défaut de se produire. Ceci est fait pour que le formulaire ne soit pas soumis tant qu'il n'a pas été correctement **validé** par le code qui suit.
* La variable **isValid** est également définie et initialisée à **true**. Cette variable sera utilisée pour savoir **si le formulaire est valide ou non**. Si l'un des contrôles de validation **échoue**, **isValid** sera défini sur **false**.
* Si la valeur de l'élément **usernameInput** est une chaîne **vide** après avoir appelé la méthode trim dessus, un message disant "***Username ne peut pas être vide"*** sera afficher a laide de l'innerHTML d'un élément avec l'identifiant **"username-error**".
* Si la valeur n'est pas une chaîne vide, il vérifie alors si **la longueur** de la valeur est **inférieure à 3 caractères ou supérieure à 25 caractères**. Si l'une de ces conditions est vraie, il définit l'innerHTML de l'élément avec l'identifiant "***username-error***" sur un message indiquant ***"Le nom d'utilisateur doit comporter entre 3 et 25 caractères"***. Si aucune de ces conditions **n'est vraie**, il définit l'innerHTML de l'élément avec l'identifiant "**username-error**" sur une chaîne vide.
* *//validate email*: utiliser pour valider le champs de de saisie de l email ; si la

valeur de l’entrée avec l'id ***"emailInput***" est vide ou se compose uniquement de caractères d'espacement. Si c'est le cas, il définit l'innerHTML d'un élément avec l'id "***email-error***" sur la chaîne "*Email is required"* et définit la variable **"isValid"** sur **false**. Si l'entrée email n'est pas vide, elle efface l'innerHTML de l'élément "email-error".

* *//Validate password*: vérifie la valeur du “password” et de "passwordConfirmInput". Si la valeur de "passwordInput" est vide ou se compose uniquement de caractères d'espacement, il définit l'innerHTML d'un élément avec l'id "password-error" sur la chaîne "Password is required" et définit la variable "isValid" sur false.
* Si la valeur de "passwordInput" est égale à la valeur de "passwordConfirmInput", cela efface l'innerHTML de l'élément "password-error".
* Enfin on vérifie la valeur de la variable "**isValid"**. Si "**isValid**" est **true** il redirige la page en cours vers une page avec l'URL *"index.html"*. Si "**isValid**" est **false**, ce bloc de code ne fait rien.

**3-index.html:**



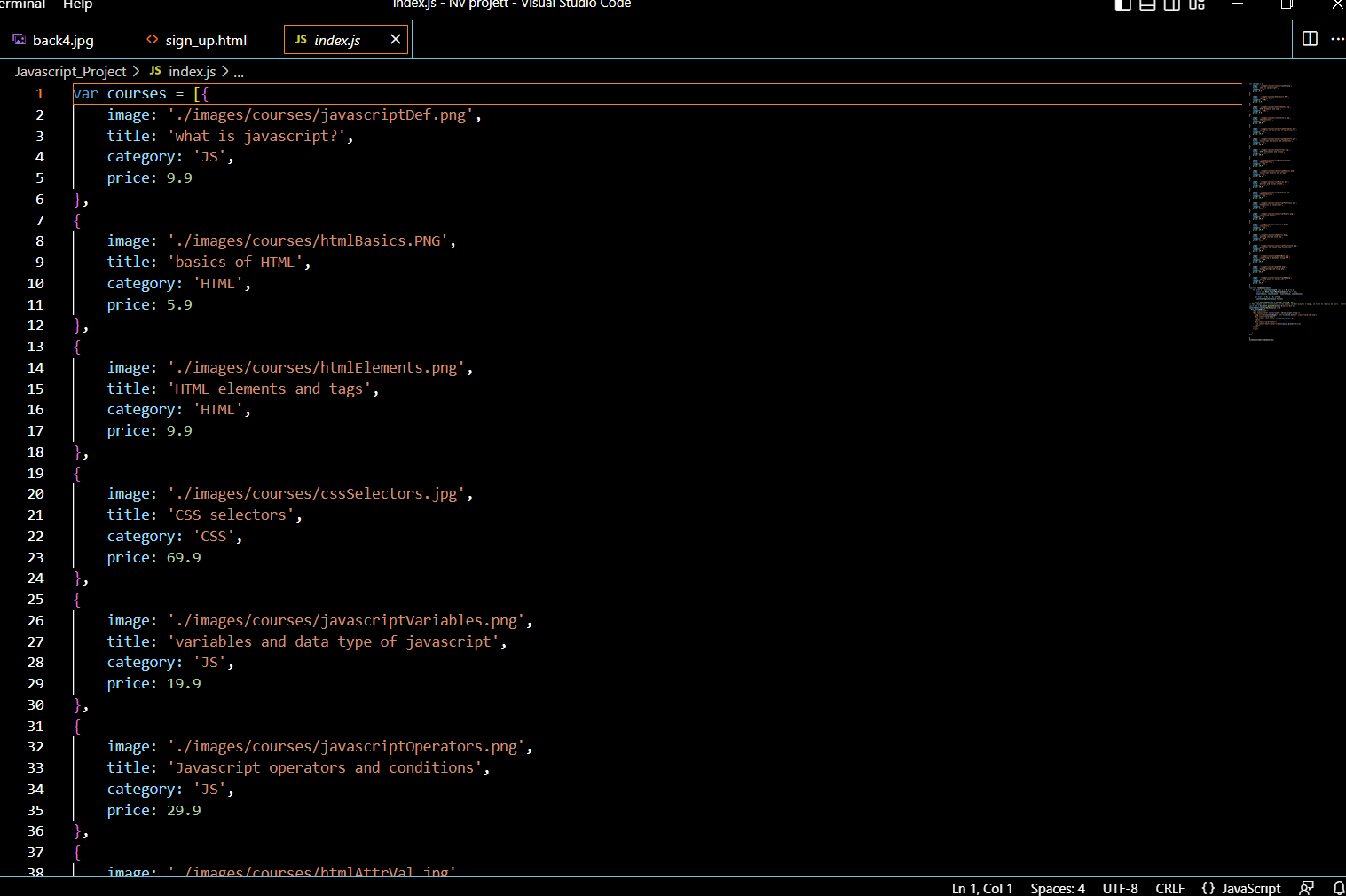


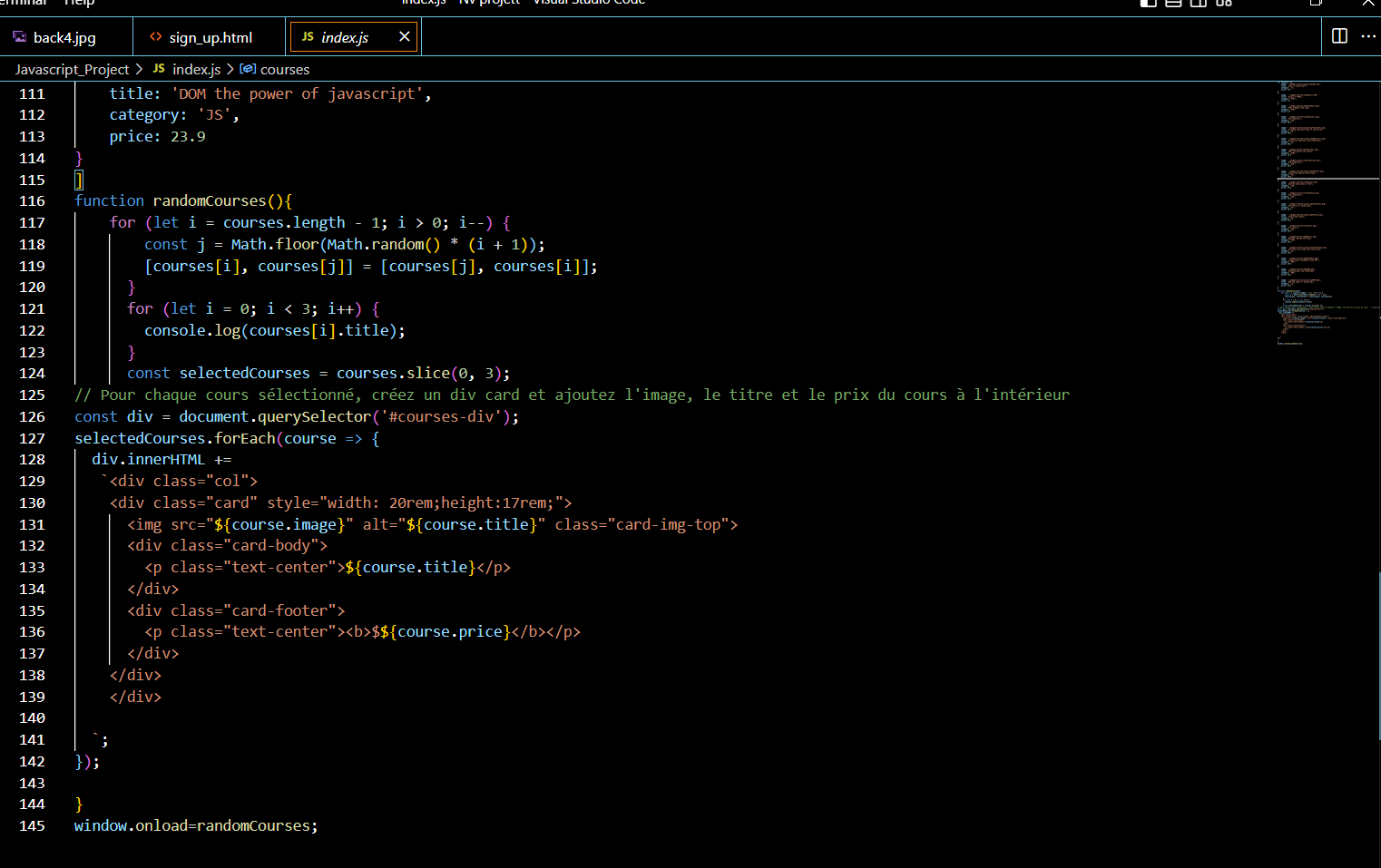
-J’ai d’abord creer une barre de navigation avec des liens vers les pages "Cours" et "courses”et”about” , une image avec du texte superposé, une liste de cours et un pied de page avec un avis de droit d'auteur.

-j ai ainsi integrer les liens vers des ressources externes telles que *jQuery, Bootstrap et un fichier JavaScript local appelé* ***"index.js****"*.

- le bouton avec la classe "btn-dark" est cliqué, l'utilisateur est redirigé vers une page appelée "*courses.html"*

**4-index.js:**





-J’ai crée un tableau d'objets représentant des cours, chaque objet contenant des informations sur le cours, telles que son image, son titre, sa catégorie et son prix.

-La fonction **randomCourses** permet de mélanger le tableau des cours, puis enregistre le titre des 3 premiers éléments du tableau mélangé dans la console. Il crée également un nouveau tableau appelé **selectedCourses** qui contient les 3 premiers éléments du tableau des cours mélangés.

-Une fois le tableau mélangé, la fonction enregistre le titre des 3 premiers éléments du tableau dans la console à l'aide d'une boucle for. Il crée ensuite un nouveau tableau appelé **selectedCourses** qui contient les **3 premiers éléments** du tableau des cours mélangés à l'aide de la méthode slice().

- un élément div avec un identifiant courses-div,va parcourir un tableau de cours sélectionnés, créant un élément div pour chacun avec la classe "card". Il définira la largeur et la hauteur de la carte et ajoutera une image, un titre et un prix pour chaque cours à l'intérieur de la carte.

- la fonction **randomCourses** est **exécutée**.

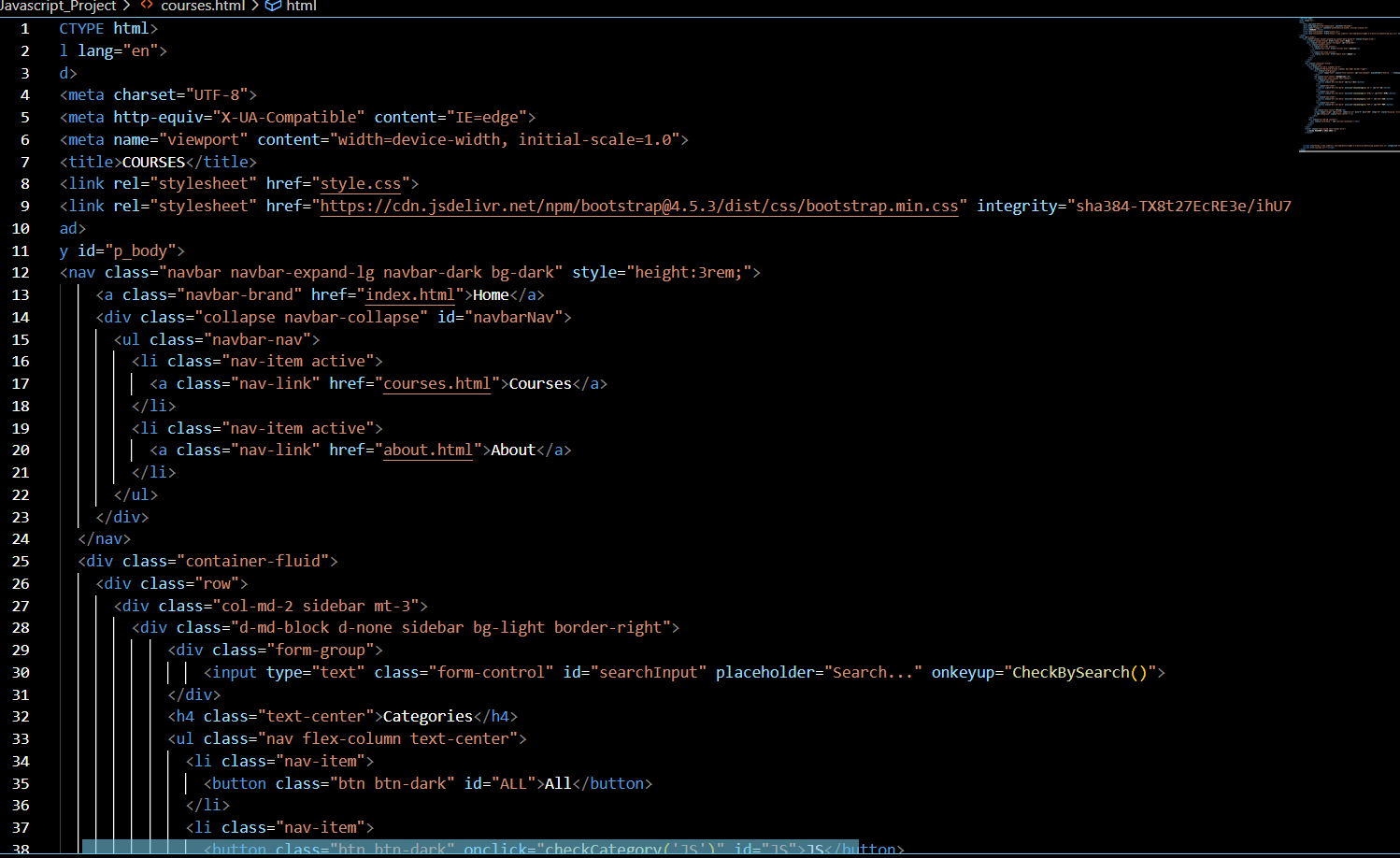
**5-courses html:**

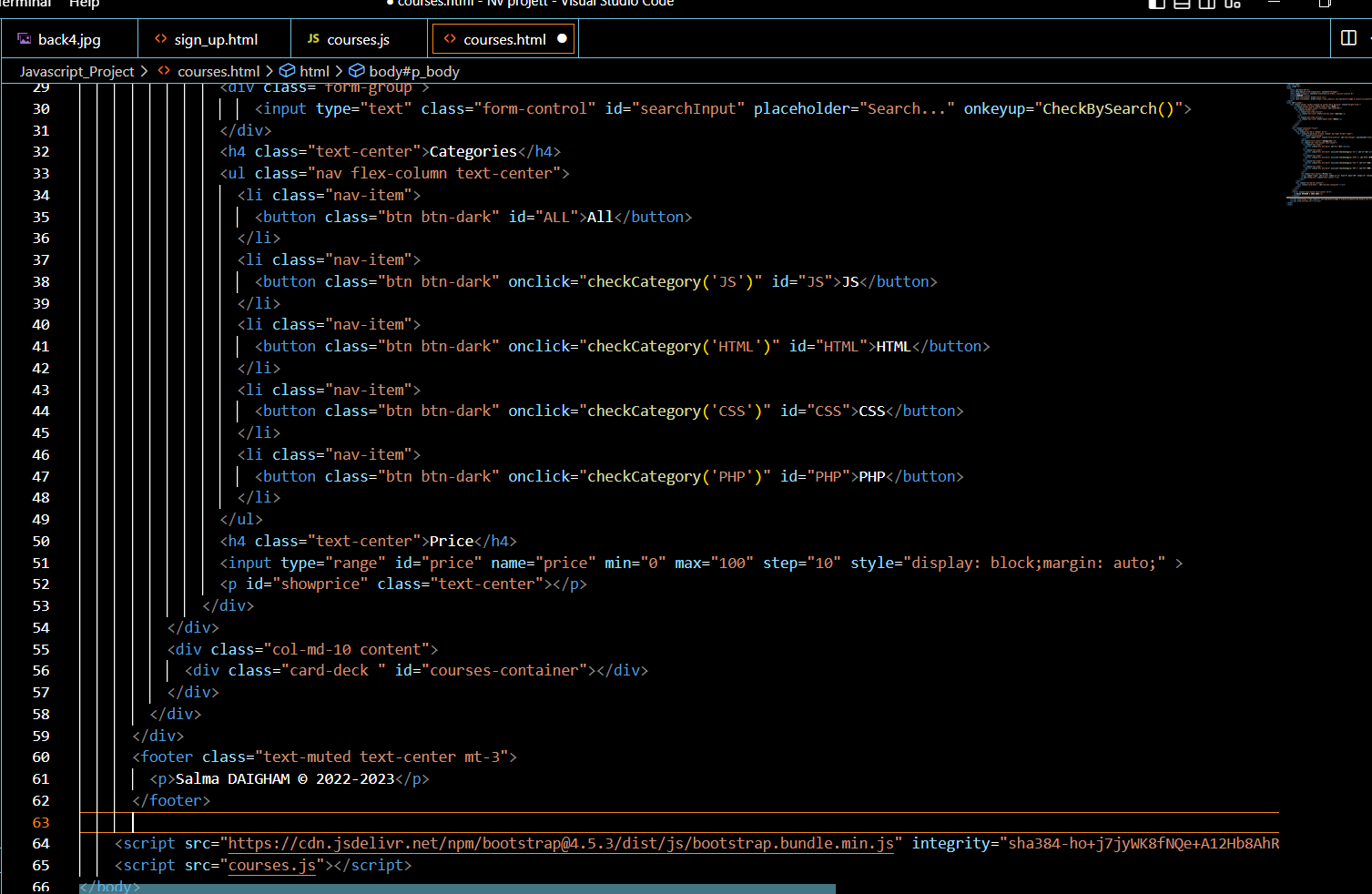
**-**J ai créer une barre de navigation en haut et une liste de cours dans la zone de contenu principale. La barre de navigation contient des liens vers les pages "**Accueil", "Cours" et “about".**

-Les cours sont affichés dans une grille à l'aide d'un élément "**card-deck"** avec l'ID "courses-container".

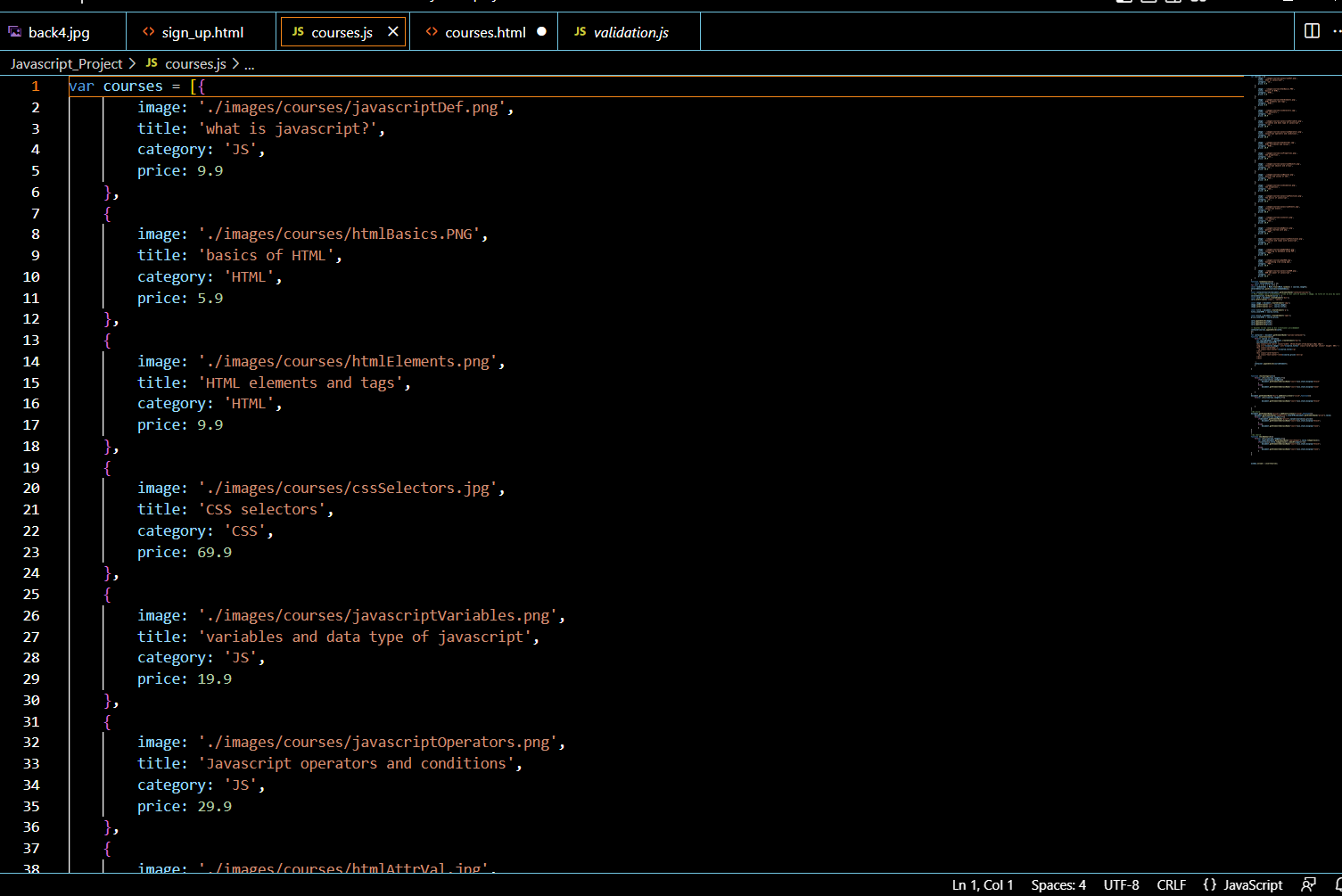
-j ai creer également une barre latérale sur la gauche qui permet aux utilisateurs de filtrer les cours par **catégorie** (***JavaScript, HTML, CSS, PHP***) et **par prix.**

-Le filtrage est effectué à l'aide de fonctions JavaScript qui sont déclenchées par des clics de bouton et des modifications de l'élément d'entrée de plage. La page comporte également un pied de page avec un avis de droit d'auteur.





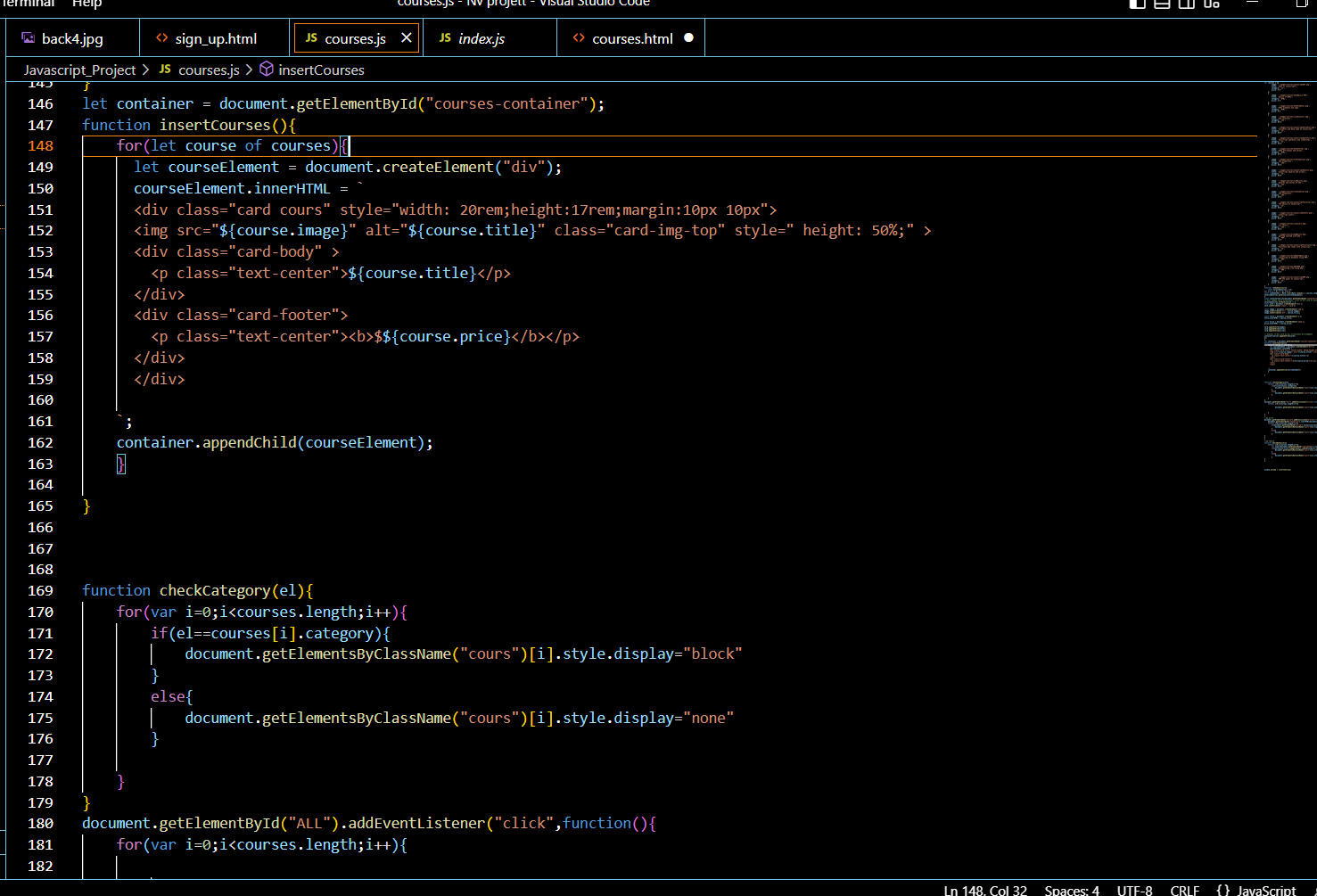
**6-courses.js:**

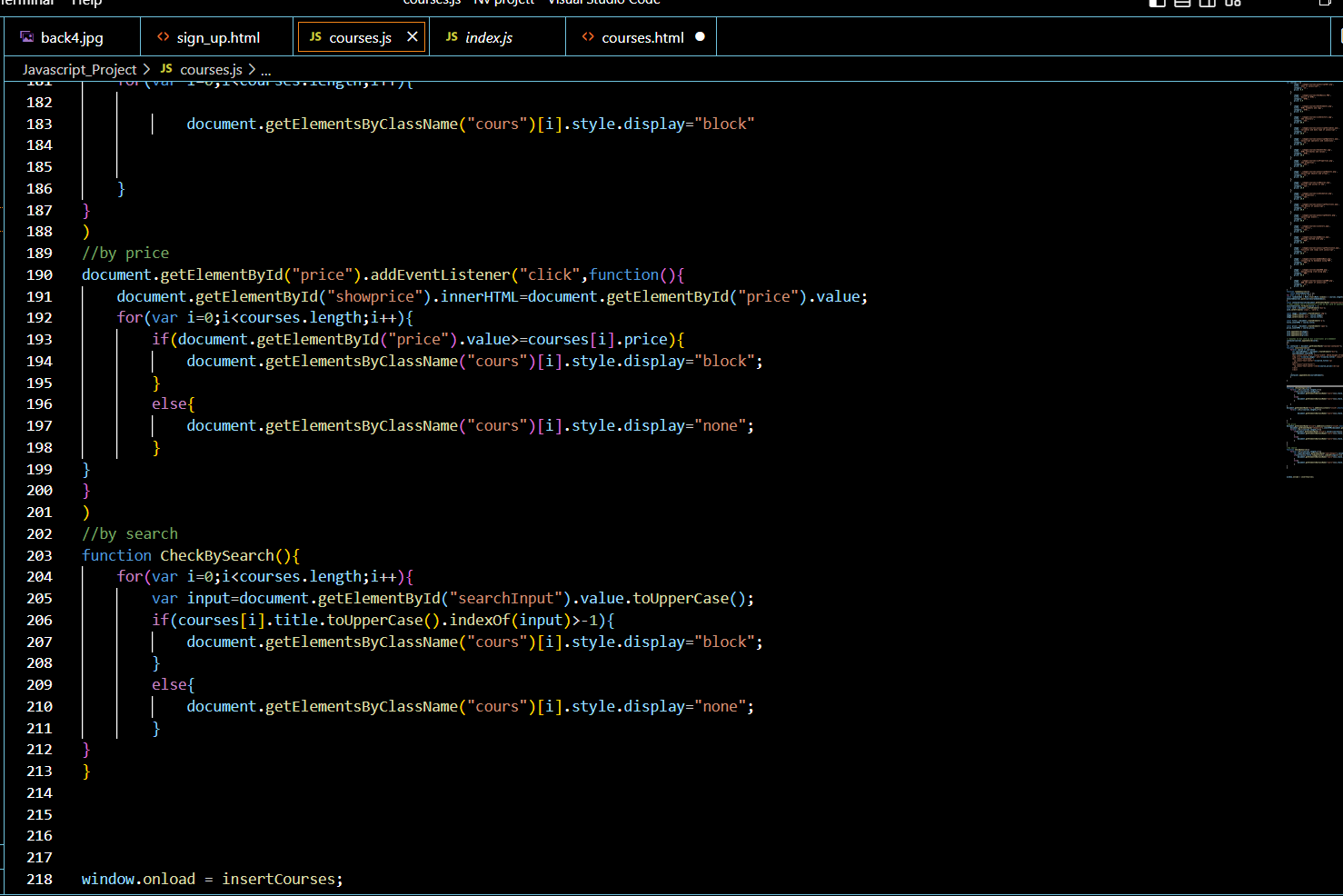


-J’ai crée un tableau d'objets représentant des cours, chaque objet contenant des informations sur le cours, telles que **son image**, s**on titre**, s**a catégorie** et **son prix**.



-





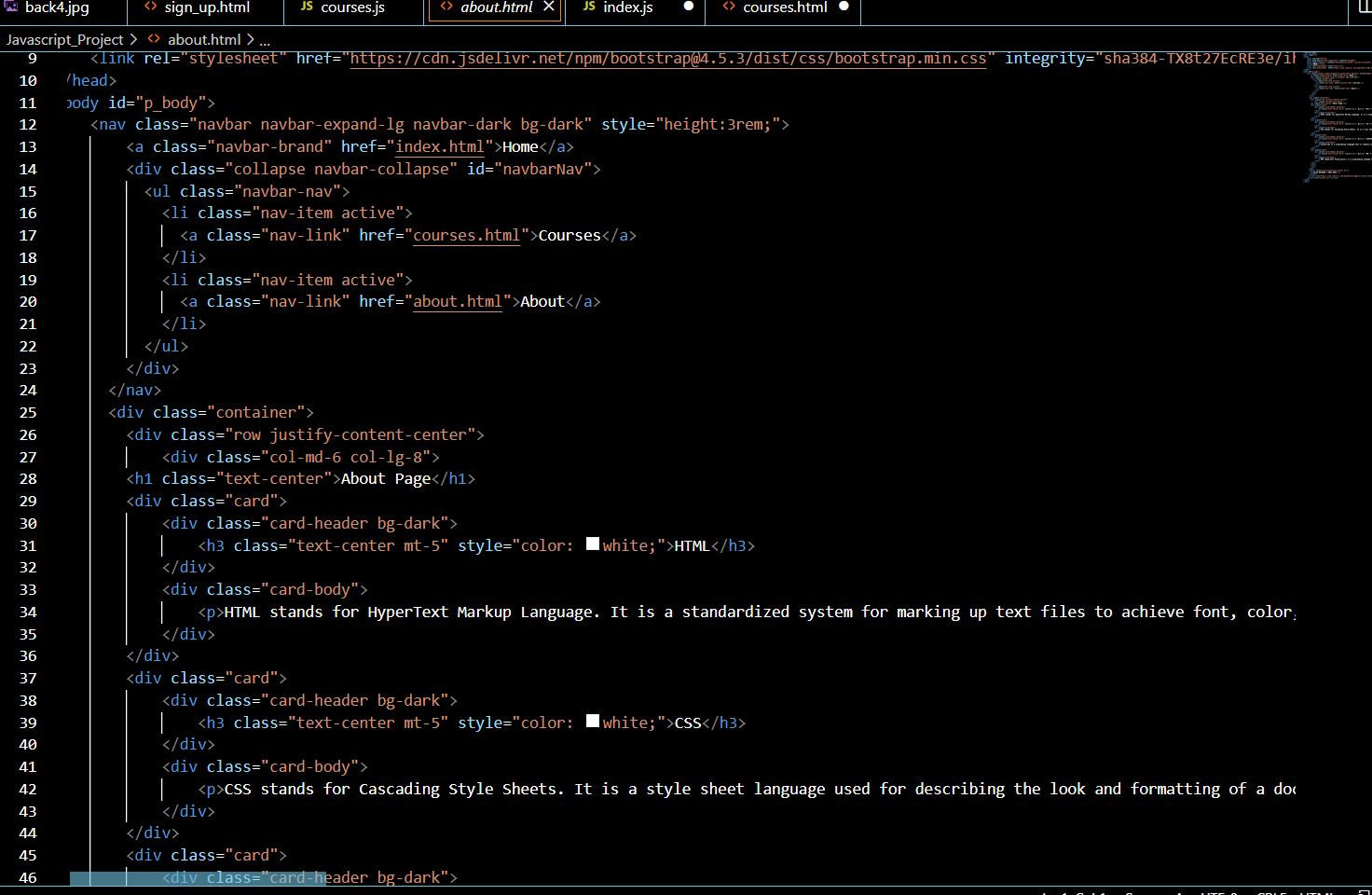
-*//By price*:On a ajouter un écouteur d'événement à un élément avec un identifiant de prix, qui, lorsqu'il est cliqué, parcourt le tableau courses et vérifie la valeur de l'élément avec l'identifiant **price**. Si la valeur de cet élément est supérieure ou égale au prix du cours, il définit la propriété de style d'affichage de l'élément avec la classe cours à bloquer, sinon il la définit à aucun.

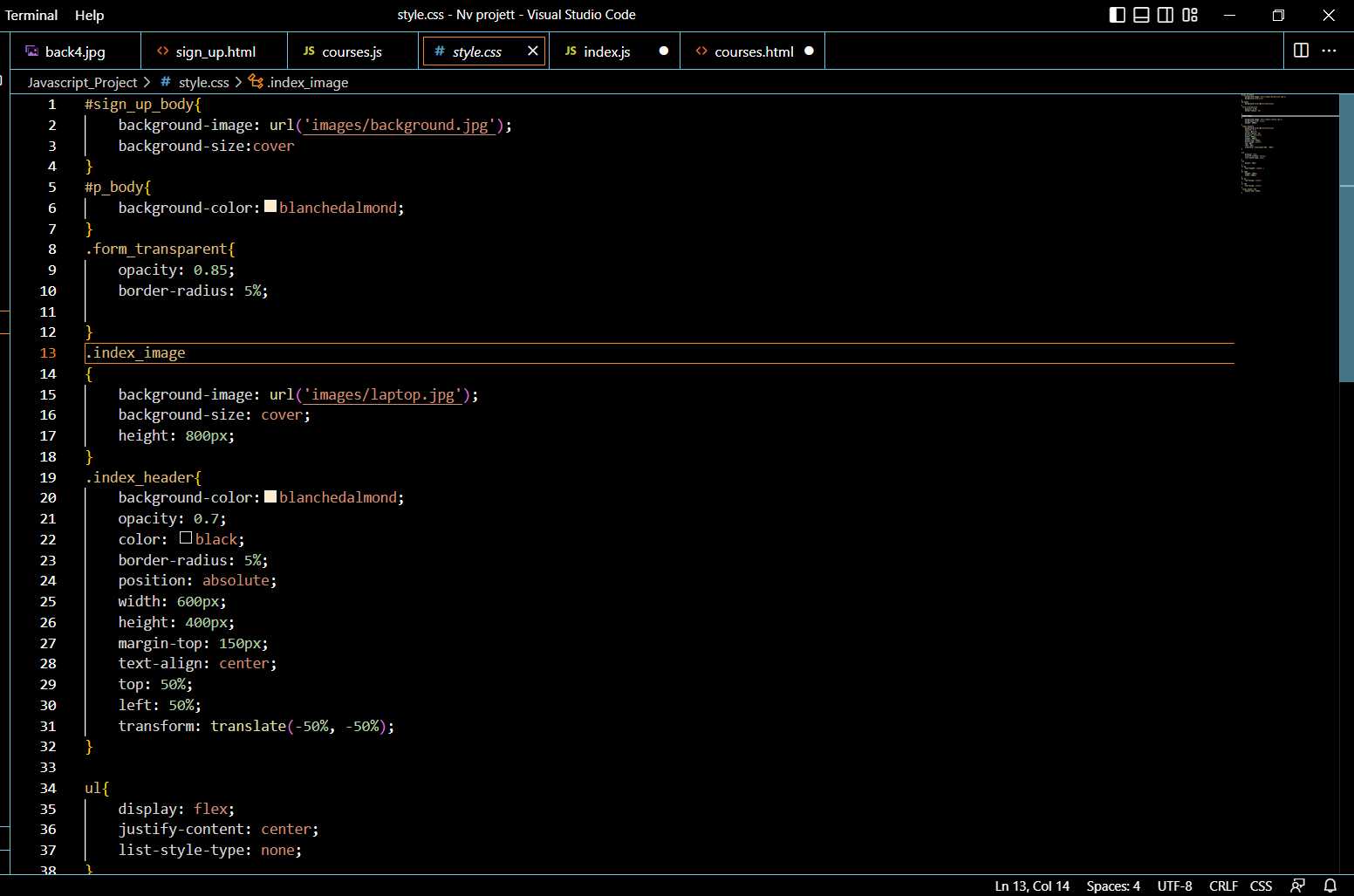
*-//By search*: on a définit également une fonction **CheckBySearch** qui parcourt le tableau courses et vérifie si le titre du cours contient la valeur d'un élément d'entrée avec l'id searchInput. Si c'est le cas, il définit la propriété de style d'affichage de l'élément avec la classe cours sur block, sinon il la définit sur none.

-l**a fonction Checkcategory**: est définie pour prendre  **un élément e1**. Il parcourt ensuite un tableau de cours et vérifie **si e1 est égal à la catégorie du cours**. Si c'est le cas, il définit la propriété de style d'affichage d'un élément avec la classe cours à bloquer. Si ce n'est pas le cas, il définit la propriété de style d'affichage de l'élément sur none.

**7-about.html:**

-j ai ajouter plusieurs div contenant des informations sur différentes technologies de développement Web (*HTML, CSS, JavaScript et PHP* ). Chaque carte comprend un en-tête avec le nom de la technologie et un corps avec des informations à son sujet. Il y a aussi une barre de navigation en haut de la page qui comprend des liens vers la page d'accueil et deux autres pages (**cours et about**).



**8\_style.css:**

-Plusieurs modifications de style sont appliquées à différents éléments d'une page Web. Par exemple l’ajout d’un background image à *“sign.up”*

-les modifications de la forme dans ce travail ont été établi a l’aide de **boostrap**

FIN

DAIGHAM SALMA