**Exercice pour l’encodage URL : Prédiction de la nationalité avec l'API nationalize.io et JavaScript**

**1 Encodage d'un URL**

Avant de commencer, comprenons l'importance de l'encodage d'URL. Lorsque vous travaillez avec des URL, il est essentiel d'encoder les paramètres de requête pour s'assurer que tous les caractères spéciaux sont transformés en un format qui peut être transmis via l'URL sans ambigüité. En JavaScript, vous pouvez utiliser ***encodeURIComponent*** pour cela.

Étapes pour interroger l'API nationalize.io :

https://api.nationalize.io/?name=temblay

***Étape 1 : Récupérer les paramètres de l'URL***

Premièrement, nous devons obtenir les paramètres de recherche à partir de l'URL de notre page web.

// récupérer les paramètres de l'URL

const queryString = window.location.search;

A purple rectangle with white text

Description automatically generated

A screenshot of a computer

Description automatically generated

***Étape 2 : Analyser les paramètres de l'URL***

Ensuite, nous utilisons **URLSearchParams** pour analyser la chaîne de requête.

// placer les paramètres dans un objet URLSearchParams

const urlParams = new URLSearchParams(queryString);

***Étape 3 : Récupérer la valeur du paramètre 'name'\*\****

Nous allons maintenant chercher le paramètre spécifique `name` qui contiendra le nom dont nous voulons prédire la nationalité.

// récupérer la valeur du paramètre 'name'

const nameToSearch = urlParams.get("name");

https://api.nationalize.io/?name=temblay

A screenshot of a computer code

Description automatically generated

// Supposons que queryString contienne la partie query de l'URL,

// par exemple '?name=temblay&age=30'

const queryString = '?name=temblay&age=30';

// Dans un cas réel, vous utiliseriez window.location.search

// Créer un objet URLSearchParams à partir de la chaîne de requête

const urlParams = new URLSearchParams(queryString);

// Récupérer la valeur du paramètre 'name'

const nameToSearch = urlParams.get("name");

// Récupérer la valeur du paramètre 'age'

const ageToSearch = urlParams.get("age");

console.log(nameToSearch); // Affiche 'temblay'

console.log(ageToSearch); // Affiche '30'

A screenshot of a computer program

Description automatically generated

***Étape 4 : Vérifier si le paramètre 'name' est présent***

Nous vérifions si le paramètre 'name' est présent dans l'URL. S'il ne l'est pas, nous informons l'utilisateur.

if (nameToSearch) {

// Si 'name' est présent, nous continuerons avec la requête.

fetchNationality(nameToSearch);

} else {

// Si 'name' est absent, nous informons l'utilisateur.

console.log("Le paramètre 'name' n'est pas présent dans l'URL.");

}

***Étape 5 : Créer une fonction pour interroger l'API***

Maintenant, nous créons une fonction qui prendra le nom et fera une requête à l'API nationalize.io.

function fetchNationality(nameParam) {

// Encodage du paramètre pour s'assurer qu'il est sécurisé pour l'URL.

const encodedName = encodeURIComponent(nameParam);

// Utiliser fetch pour envoyer la requête à l'API.

fetch(`https://api.nationalize.io/?name=${encodedName}`)

.then(response => {

// Vérifier si la réponse est ok.

if (!response.ok) {

throw new Error('Réponse réseau non ok.');

}

return response.json();

})

.then(data => {

// Traiter et afficher la réponse.

console.log(data);

// Ici, vous pouvez ajouter du code pour afficher les données dans votre HTML.

})

.catch(error => {

// Attraper et afficher les erreurs.

console.error('Erreur lors de la requête à l\'API nationalize.io:', error);

});

}

***Étape 6 : Exécuter le script***

Pour que ce script fonctionne, vous devez l'inclure dans une page HTML ou l'exécuter dans la console de votre navigateur alors que l'URL de la page contient le paramètre name. Par exemple : https://api.nationalize.io/?name=temblay

***Conclusion :***

Cet exercice vous a montré comment récupérer un paramètre de l'URL, l'encoder correctement pour une requête HTTP et utiliser cet argument pour interroger l'API nationalize.io afin de prédire la nationalité basée sur un nom. Vous pouvez intégrer ces étapes dans votre application web pour enrichir l'interaction avec vos utilisateurs.