|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰1**  **Formation:** DD 201  **Prof:** SAID MAHDAD  **Module:** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

1. **Sur le bureau, Créer un dossier « Front-end »**
2. **Réaliser une page web suivante nommée « ser1tp1.html » :**



1. **Ajouter les autres balises que vous connaissez à ce tableau et enregistrez votre travail.**

**TP2 :**

1. **Dans une page web nommée « ser1tp2.html », Réaliser les deux listes suivantes :**

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰2**  **Formation:** DD 201  **Prof:** SAID MAHDAD  **Module:** développer en Front-end  **Nom et prénom :** |

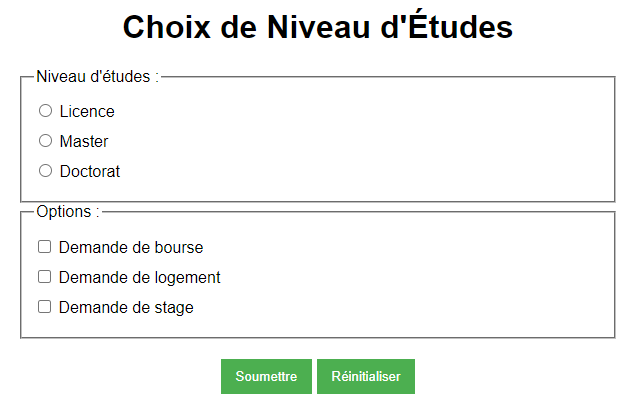
**TP 1 :**

1. **Réaliser la page web « ser2tp1.html »:**



**TP 2:**

1. **Réaliser la page web « ser2tp2.html »:**

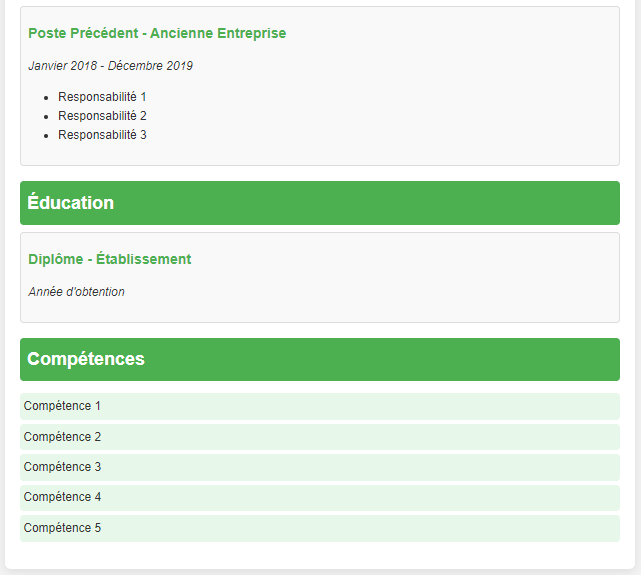


|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰3**  **Formation:** DD 201  **Prof:** SAID MAHDAD  **Module:** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

1. **En utilisant HTML et CSS, réaliser une page web(CV) sous le nom « ser3tp1.html » :**





|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰4**  **Formation:** DD 201  **Prof:** SAID MAHDAD  **Module:** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

1. **Créer une page web interactive « ser4tp1.html » où les étudiants peuvent tester leurs connaissances avec un quiz simple. Cette page contiendra des questions à choix multiple (QCM) et utilisera du HTML et CSS, avec une touche de JavaScript pour gérer les réponses.**



**TP2 :**

1. **A partir de la page web précédente, créer une nouvelle page web « ser4tp2.html » en Ajoutant plus de questions.**
2. **Afficher des messages en fonction du score**

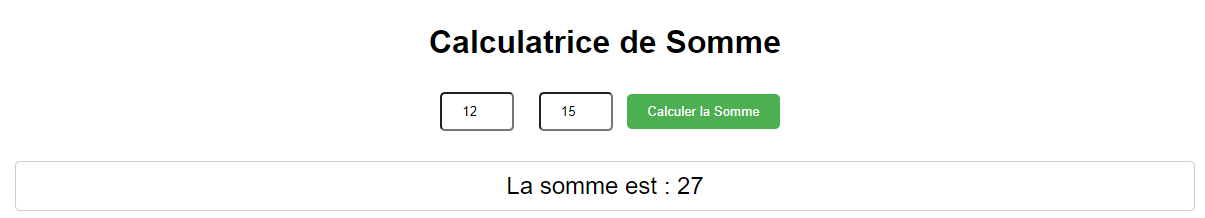
**(par exemple, "Bien joué !" ou "Essayez encore !").**

1. **Enregistrer les scores dans le stockage local pour un quiz futur.**

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰5**  **Formation:** DD 201  **Prof:** SAID MAHDAD  **Module:** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

**Créez un formulaire avec deux champs de saisie pour entrer deux nombres, affichez la somme des deux nombres.**

****

**TP 2 :**

**Créez un formulaire de connexion avec un champ pour le nom d'utilisateur et un champ pour le mot de passe. Vérifiez que les champs ne sont pas vides et affichez un message d'erreur si l'un d'eux est vide. La bordure de la zone non remplie doit être en rouge.**

****

**Dans le cas contraire, affiche un message « Succès » dans une boite de dialogue**

**TP 3:**

**Créez un formulaire avec un champ pour le mot de passe. Lorsque l'utilisateur saisit son mot de passe, vérifiez la longueur et affichez un message indiquant si le mot de passe est "faible", "moyen", ou "fort".**

****

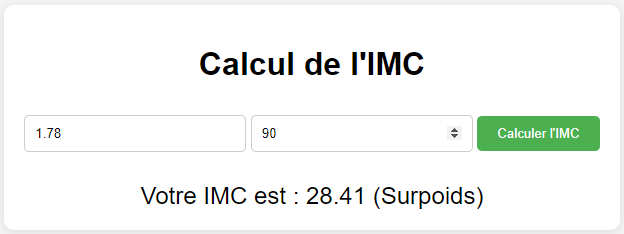
|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰6**  **Formation:** DD 201  **Prof:** SAID MAHDAD  **Module:** développer en Front-end  **Nom et prénom :** |

**TP 1:**

**Créez un formulaire avec deux champs de saisie : un pour la taille (en mètres) et un pour le poids (en kilogrammes). Lorsque l'utilisateur soumet le formulaire, affichez son IMC, et donner la catégorie.**

**(Insuffisance pondérale** : IMC < 18.5 **;Poids normal** : 18.5 ≤ IMC < 24.9 **;**

**Surpoids** : 25 ≤ IMC < 29.9 **;Obésité** : IMC ≥ 30)



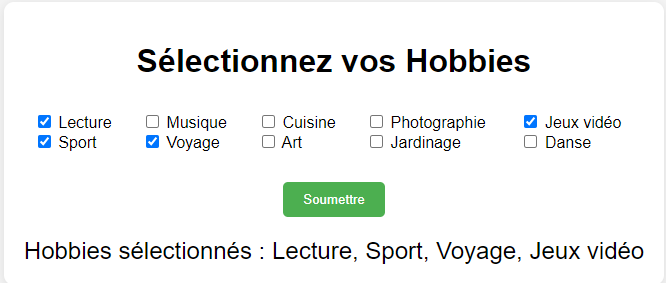
**TP 2:**

**Créez un formulaire avec deux champs : un pour le montant en euros et un pour le taux de change, Lorsque l'utilisateur soumet le formulaire, affichez le montant converti en dirhams.**

****

**TP 3:**

**Créez un formulaire avec une liste de cases à cocher pour sélectionner plusieurs hobbies. Lorsque l'utilisateur soumet le formulaire, affichez les hobbies sélectionnés.**

****

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰7**  **Formation:** DD 201  **Prof:** SAID MAHDAD  **Module:** développer en Front-end  **Nom et prénom :** |

**TP 1:**

**un formulaire avec un champ pour saisir une adresse e-mail. Lorsque l'utilisateur soumet le formulaire, vérifiez si l'adresse e-mail est valide et affichez un message approprié.**



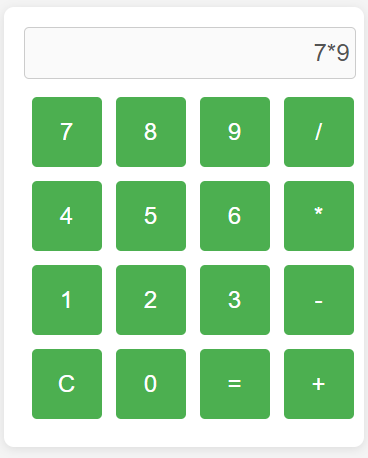
**TP 2:**

**Créer une application de liste de tâches qui inclut la date et l'heure.**

****

**TP 3:**

**Créez la calculatrice suivante :**

****

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰8**  **Formation:** DD 201  **Prof:** SAID MAHDAD  **Module:** développer en Front-end  **Nom et prénom :** |

**TP 1:**

**Créez un formulaire qui compte le nombre de mots, de caractères et de phrases dans un texte que l'utilisateur saisit.**



**TP 2:**

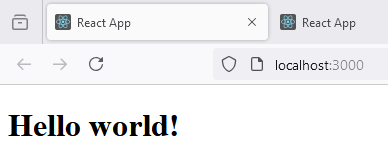
**Créez un formulaire qui compte le nombre total de "J'aime" et de "Je n'aime pas".**



|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰9**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

1. **Installer le *NodeJs* et le *Npm*, puis créer votre première application React**
2. **Réaliser une page qui affiche « Hello, World »**

****

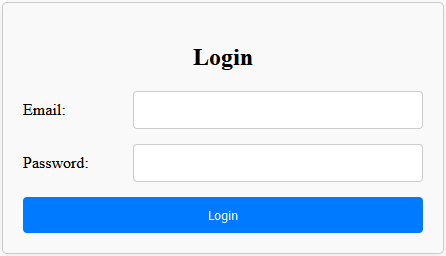
**TP 2 :**

**Réaliser une page suivante :**

|  |  |
| --- | --- |
|  |  |

**TP 3 :**

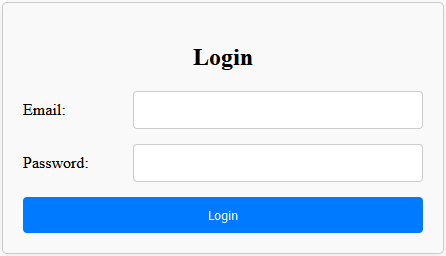
**Réaliser une page suivante :**

****

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰10**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

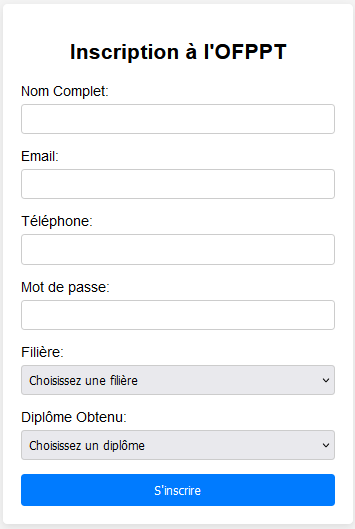
**TP 1 :**

**En utilisant JSX, réaliser la page web suivante :**

****

**TP 2 :**

**En utilisant JSX, réaliser la page web suivante :**

****

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰11**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**Exercice 1 :**

**Convertissez les composants suivants écrits en JSX en utilisant** *React.createElement* **:**

|  |  |
| --- | --- |
| ***const Greeting = () => {***  ***return <h1>Bienvenue à l'OFPPT!</h1>;***  ***};*** |  |
| ***const TodoList = () => {***  ***const todos = ['Apprendre React',***  ***'Pratiquer les exercices', 'Construire un projet'];***  ***return (***  ***<ul> {***  ***todos.map(todo => <li key={todo}>{todo}</li>)}***  ***</ul>***  ***);***  ***};*** |  |
| ***const UserProfile = ({ name, age }) => {***  ***return (***  ***<div>***  ***<h2>{name}</h2>***  ***<p>Âge: {age}</p>***  ***</div>***  ***);***  ***};*** |  |

**Exercice 2 :**

**Convertissez les composants suivants en JSX**

|  |  |
| --- | --- |
| ***const App = () => {***  ***return React.createElement(***  ***'div',***  ***null,***  ***'Bienvenue à l\'OFPPT!');***  ***};*** |  |
| ***const App = () => {***  ***return React.createElement(***  ***'div',***  ***null,***  ***React.createElement(***  ***'h1',***  ***null,***  ***'Mon Application'),***  ***React.createElement(***  ***'p',***  ***null,***  ***'Ceci est un paragraphe.')***  ***);***  ***};*** |  |

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰12**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

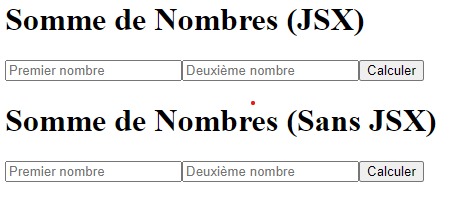
**Exercice 1 :**

**Convertissez le composant suivant en JSX**

|  |  |
| --- | --- |
| ***const ShoppingCart = () => {***  ***return React.createElement(***  ***'div',***  ***null,***  ***React.createElement(***  ***'h2',***  ***null,***  ***'Mon Panier'),***  ***React.createElement(***  ***'ul',***  ***null,***  ***React.createElement(***  ***'li',***  ***null,***  ***'Produit 1'),***  ***React.createElement(***  ***'li',***  ***null,***  ***'Produit 2'),***  ***React.createElement(***  ***'li',***  ***null,***  ***'Produit 3')***  ***)***  ***);***  ***};*** |  |

**Exercice 2 :**

**Réaliser la page suivante :**

**

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰13**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

**Créer la page suivante qui contient 5 composants:**

****

**TP 2 :**

**Créer la page suivante :**

**

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰14**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

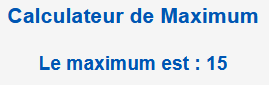
**TP 1 :**

**Créez un composant SumCalculator qui prend trois nombres en tant que props (num1, num2, num3) et affiche leur somme dans un composant enfant DisplaySum.**

****

**TP 2 :**

**Créez un composant MaxCalculator qui prend trois nombres (num1, num2, num3) et affiche le maximum parmi ces trois dans un composant enfant DisplayMax.**

****

**TP 3 :**

**Créez un composant VowelCounter qui prend une chaîne de caractères (text) et affiche le nombre de voyelles dans un composant enfant DisplayVowelCount.**

**

**TP 4 :**

**Créez un composant UnitConverter qui prend une distance en kilomètres (kilometers) et affiche sa conversion en miles dans un composant enfant DisplayMiles.**

**

**TP 5 :**

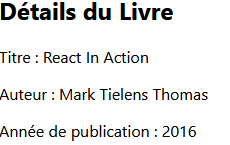
**Créez un composant PositiveChecker qui prend un nombre (number) et affiche si ce nombre est positif, négatif ou zéro dans un composant enfant DisplaySign.**

**

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰15**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

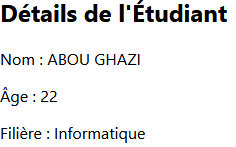
**TP 1 :**

**Créez un composant fonctionnel sous le nom Book qui reçoit un objet book en props et affiche les détails du livre (titre, auteur et année de publication)**

****

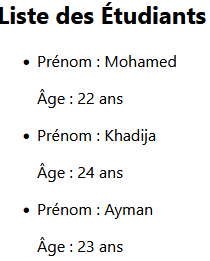
**TP 2 :**

**Créez un composant de classe sous le nom Student qui reçoit un objet student en props et affiche les informations de l'étudiant (nom, âge, filière).**

****

**TP 3 :**

**Créez un composant fonctionnel StudentList qui reçoit un tableau students en props et affiche une liste d'étudiants. Chaque étudiant doit afficher son prénom et son âge.**

**

**TP 4 :**

**Créez un composant de classe sous le nom BookList qui reçoit un tableau books en props et affiche la liste des livres. Chaque livre doit afficher son titre et son auteur.**

**

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰16**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

**Créez deux composants (de classe et fonctionnel) qui réalisent la somme de deux nombres saisis au clavier**

****

**TP 2 :**

**Créez un composant fonctionnel qui réalise la somme de deux nombres saisis au clavier après avoir cliqué un bouton**

**

**TP 4 :**

**Créez le composant qui affiche le formulaire suivant :**



|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Récapitulatif**  **Formation : DD 201**  **Prof : SAID MAHDAD**  **Module : développer en Front-end**  **Nom et prénom :** |

**Exercice 1 :**

* + 1. **Quelle commande est utilisée pour créer un nouveau projet React ?**

|  |  |
| --- | --- |
| * + npm create react-app my-app   + npm start react-app my-app | * + npx create-react-app my-app   + npm install react-app my-app |

* + 1. **Qu'est-ce que JSX ?**
  + Une bibliothèque JavaScript
  + Une syntaxe de JavaScript permettant d'écrire des balises HTML dans le code JavaScript
  + Un framework pour les interfaces utilisateur
  + Une fonction JavaScript permettant d'intégrer CSS dans React
    1. **Quel mot-clé est utilisé pour créer un composant de classe en React ?**

|  |  |
| --- | --- |
| * + constructor   + class | * + function   + component |

* + 1. **Comment appelle-t-on une fonction qui renvoie du JSX ?**

|  |  |
| --- | --- |
| * Un composant de classe * Un composant fonctionnel | * Une fonction utilitaire * Une méthode de classe |

* + 1. **Comment passe-t-on des données d'un parent à un composant enfant ?**

|  |  |
| --- | --- |
| * state * props | * context * setState |

* + 1. **Lequel des éléments suivants est utilisé pour définir les props d’un composant fonctionnel en React ?**

|  |  |
| --- | --- |
| * this.props * Les paramètres de la fonction | * state * render |

* + 1. **Lequel des suivants est un avantage des composants fonctionnels sur les composants de classe ?**

|  |  |
| --- | --- |
| * + Ils sont plus complexes   + Ils nécessitent moins de code | * + Ils sont plus lents à exécuter   + Ils ne supportent pas les hooks |

* + 1. **Quelle est la fonction principale de render() dans un composant de classe ?**
* Gérer les événements utilisateur
* Déclencher un cycle de vie
* Retourner du JSX pour afficher l'interface utilisateur
* Charger les données externes
  + 1. **Quelle méthode est utilisée pour retourner du contenu dans un composant de classe ?**

|  |  |
| --- | --- |
| * render() * display() | * show() * return() |

* 1. **Quel est le rôle principal des props dans React ?**

|  |  |
| --- | --- |
| * Gérer l'état interne d'un composant * Passer des données du parent à l'enfant | * Ajouter des événements de clics * Gérer les appels d'API |

**Exercice 2 :**

* + 1. **Décrivez le processus d'installation de React et de création d'une nouvelle application React.**

* + 1. **Qu'est-ce que JSX, et pourquoi est-il utilisé dans React ? Donnez un exemple de JSX.**

* + 1. **Quelle est la différence entre un composant de classe et un composant fonctionnel dans React ? Donnez un exemple de chaque.**

* + 1. **Comment passe-t-on un objet comme prop à un composant enfant en React ? Montrez-le avec un exemple.**

* + 1. **Expliquez comment on peut passer un tableau comme prop et comment il est utilisé dans le composant enfant. Donnez un exemple.**

**Exercice 3 :**

1. **Créez un composant fonctionnel appelé WelcomeMessage qui prend un prop name et affiche "Bienvenue, [name]".**
2. **Créez un composant de classe appelé UserCard qui prend deux props : name et age, et qui affiche un petit paragraphe contenant ces informations.**
3. **Créez un composant fonctionnel appelé FruitList qui prend un tableau de fruits en prop et qui affiche chaque fruit dans une liste non ordonnée (<ul>).**
4. **Créez un composant de classe appelé ProductCard qui prend trois props d’un objet Produit : name, price, et description, et qui affiche ces informations dans une carte de produit stylée avec des balises HTML comme <h2>, <p>, etc.**
5. **Créez un composant fonctionnel appelé GreetingList qui prend un tableau de noms en prop et affiche un message de bienvenue pour chaque nom sous forme de liste d'éléments <h3>.**

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Mini-Projet1**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**Projet : Application de Publication avec Réactions et Commentaires (Similaire à Facebook)**

**Objectif :**

Créer une application React simple qui permet de publier du texte, d'ajouter des réactions ("J'aime", "Je n'aime pas") et de gérer les commentaires pour chaque publication. Le design de l'application doit être basique mais agréable, avec un affichage centré et des fonctionnalités interactives.

**Fonctionnalités requises :**

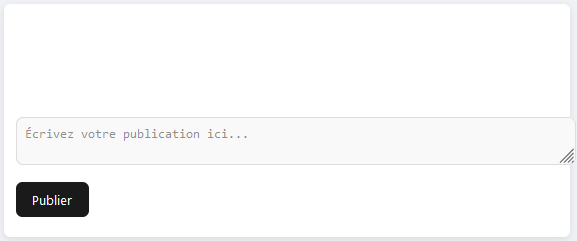
1. **Zone de saisie de la publication** :
   * L'utilisateur doit pouvoir saisir un texte dans une zone de texte pour créer une publication.
   * Après avoir cliqué sur le bouton "Publier", la publication doit s'afficher en dessous de la zone de saisie.
   * La zone de saisie doit rester active pour permettre d'ajouter d'autres publications sans supprimer les précédentes.
2. **Affichage de la publication** :
   * Chaque publication doit être affichée avec un texte aligné à gauche, et les publications doivent apparaître sous la zone de saisie.
   * Les publications doivent inclure trois boutons :
     + **"J'aime"** : Incrémente le nombre de "J'aime".
     + **"Je n'aime pas"** : Incrémente le nombre de "Je n'aime pas".
     + **"Commenter"** : Permet d'afficher une zone de texte pour écrire des commentaires.
3. **Commentaires** :
   * En cliquant sur "Commenter", une zone de saisie doit apparaître pour permettre à l'utilisateur de saisir un commentaire.
   * Les commentaires doivent être publiés après avoir cliqué sur le bouton "Ajouter commentaire".
   * Les commentaires doivent être affichés en dessous de la publication .

**Technologies :**

* **React.js** : Utiliser React pour la gestion des composants et des états.
* **CSS** : Utiliser un fichier CSS pour styliser l'application.

**Captures d’écran :**

**1)-**



**2)-**



**3)-**

****

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Mini-Projet3**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**Projet : Application de Gestion de Tâches (Todo List)**

**Objectif :**

Créer une application React qui permet de gérer des tâches (todo list) où l'utilisateur peut ajouter, marquer comme terminée, supprimer, et filtrer des tâches. L'interface sera simple mais interactive, avec des actions pour organiser les tâches quotidiennes.

**Fonctionnalités Requises :**

1. **Ajouter une tâche** :
   * L'utilisateur doit pouvoir saisir une tâche dans une zone de texte.
   * Un bouton **"Ajouter"** permet d'ajouter cette tâche à la liste.
   * Une fois ajoutée, la tâche est affichée dans une liste avec un bouton pour la marquer comme terminée ou la supprimer.
2. **Marquer une tâche comme terminée** :
   * Chaque tâche de la liste doit avoir un bouton **"Terminer"** qui, lorsqu'il est cliqué, change l'apparence de la tâche (par exemple, texte barré ou couleur différente) pour indiquer qu'elle est terminée.
3. **Supprimer une tâche** :
   * Chaque tâche doit également avoir un bouton **"Supprimer"** pour retirer la tâche de la liste.
4. **Filtrer les tâches** :
   * Ajouter des boutons de filtrage pour afficher :
     + Toutes les tâches.
     + Les tâches terminées.
     + Les tâches non terminées.

**Technologies à utiliser :**

* **React.js** : Utilisation des composants et des hooks pour gérer les tâches.
* **CSS** : Styliser l'application pour rendre l'interface agréable.

**Captures d’écran**

|  |  |
| --- | --- |
|  |  |

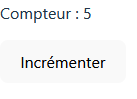
**Fonctionnalités supplémentaires (optionnel) :**

* **Édition d'une tâche** : Permettre à l'utilisateur de modifier une tâche existante.
* **Priorités** : Ajouter une fonctionnalité pour assigner une priorité (basse, moyenne, haute) à chaque tâche et les organiser en fonction de la priorité.
* **Affichage des statistiques** : Montrer combien de tâches qui sont terminées et combien restent à faire.

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰17**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

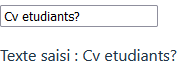
**TP 1 :**

**Créer un composant qui affiche un compteur. Le bouton "Incrémenter" augmente le compteur de 1 à chaque clic**



**TP 2 :**

**Créer un composant qui permet à l'utilisateur de saisir du texte dans un champ input et affiche ce texte sous l'input en temps réel.**



**TP 3 :**

**Créer un composant avec deux boutons, "Rouge" et "Bleu". Chaque bouton change la couleur de fond de la page en conséquence**



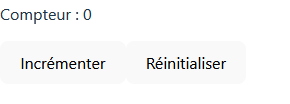
**TP 4 :**

**Créer un composant avec un bouton qui change son état entre "Visible" et "Invisible", ce qui permet de cacher ou montrer un texte**

|  |  |
| --- | --- |
|  |  |

**TP 5 :**

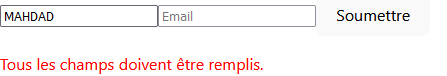
**Créer un compteur qui permet de l'incrémenter et de le réinitialiser à zéro via deux boutons séparés.**

****

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰18**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

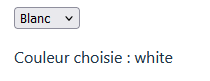
**TP 1 :**

**Créer un formulaire avec deux champs : "Nom" et "Email". Affiche un message d'erreur si l'un des champs est vide lorsque l'utilisateur clique sur "Soumettre".**

****

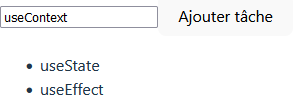
**TP 2 :**

**Créer un formulaire avec une liste déroulante. L'utilisateur peut choisir une couleur, et la couleur de fond de la page change en fonction de l'option sélectionnée.**



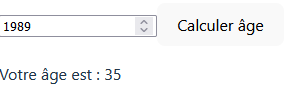
**TP 3 :**

**Créer une simple application de to-do-list où l'utilisateur peut ajouter des tâches. Chaque tâche saisie s'affiche dans une liste.**

****

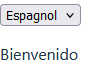
**TP 4 :**

**Créer un composant où l'utilisateur saisit son année de naissance et le composant affiche l'âge calculé en fonction de l'année actuelle.**

****

**TP 5 :**

**Crée un composant qui permet à l'utilisateur de choisir une langue dans un menu déroulant (par exemple : "Français", "Anglais", "Espagnol") et qui affiche un message de bienvenue dans la langue sélectionnée**

****

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰19**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

**Créer un composant qui affiche "Bienvenue!" une seule fois, au moment où le composant se charge.**

**TP 2 :**

**Créer un compteur avec un bouton pour augmenter la valeur. Utilise useEffect pour afficher la valeur du compteur dans la console chaque fois qu’elle change.**

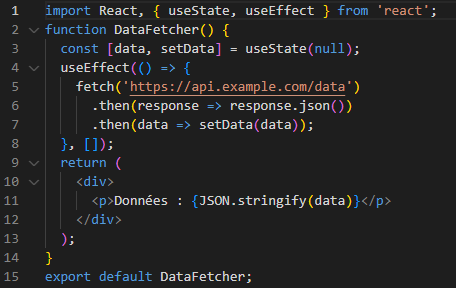
**TP 3 :**

**Créer un compteur avec un bouton pour augmenter la valeur. Utilise useEffect pour changer le titre de la page à chaque fois que le compteur est incrémenté**

****

**TP 4 :**

**Créer un composant qui récupère des données depuis une API au chargement et les affiche.**

****

**TP 5 :**

**Crée un composant qui charge des données depuis un fichier JSON externe et affiche chaque élément dans une liste.**

****

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Mini-Projet2**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**Projet :Application de Quiz sur React avec JSON**

**Objectif :**

Créer une application React qui propose une série de questions à choix multiples à l'utilisateur. Les questions seront stockées dans un fichier JSON et chargées dynamiquement dans l'application. À la fin du quiz, un score basé sur les réponses fournies sera affiché à l'utilisateur. Le design sera basique mais agréable, avec des fonctionnalités interactives.

#### Fonctionnalités Requises :

1. **Affichage des questions** :
   * Chaque question doit être affichée avec des réponses à choix multiples (au moins 4 réponses par question).
   * L'utilisateur doit pouvoir sélectionner une réponse pour chaque question.
   * Les questions seront chargées depuis un fichier JSON local (pas d'API).
2. **Calcul du score** :
   * Lorsque l'utilisateur a terminé le quiz, son score est calculé en fonction des réponses correctes fournies.
   * Le score final doit être affiché avec un message récapitulatif.
   * Un bouton **"Recommencer"** permet de relancer le quiz avec une réinitialisation du score.

**Technologies à utiliser :**

* **React.js** : Gestion des composants et des états pour afficher les questions et gérer les réponses.
* **JavaScript** : Pour la logique de vérification des réponses et du calcul du score.
* **CSS** : Pour styliser l'application (mise en page, couleurs, animations de survol).
* **JSON**: Le quiz doit utiliser un fichier ***json*** qui contient les questions, les options de réponse, et la bonne réponse.

**Captures d’écran :**

|  |  |
| --- | --- |
|  |  |

**Fonctionnalités supplémentaires (optionnel) :**

* **Chronomètre** : Limiter le temps pour répondre à chaque question ou pour l'ensemble du quiz.
* **Indicateur de progression** : Afficher la progression du quiz, par exemple "Question 3 sur 10".
* **Feedback sur les réponses** : Après avoir soumis une réponse, afficher si elle est correcte ou non avant de passer à la suivante.

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰20**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

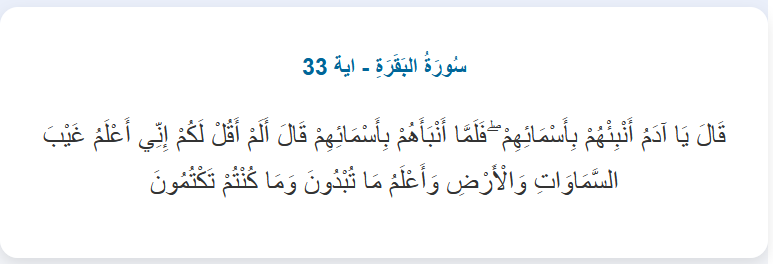
**Créer un composant qui affiche les données météo de la ville de Driouch**

****

**TP 2 :**

**Créer un composant qui retourne des verset de Quran à partir de l’API :**

***API : https://api.alquran.cloud/v1/ayah/${verseNumber}/ar***

****

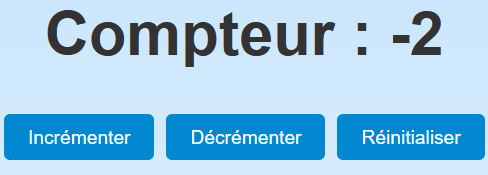
**TP 3 :**

**Créez un composant React qui affiche un compteur avec trois boutons pour :**

**Incrémenter le compteur de 1.**

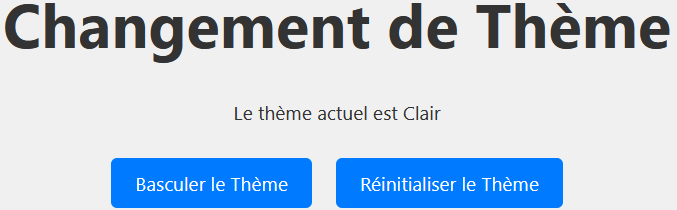
**Décrémenter le compteur de 1.**

**Réinitialiser le compteur à 0.**

****

**TP 4 :**

**Utilisez useReducer pour basculer entre un thème clair et un thème sombre, Ajoutez une option pour réinitialiser le thème à l'état par défaut (clair).**

****

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Mini-Projet3**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**Projet : Application de Gestion de Tâches (Todo List)**

**Objectif :**

Créer une application React qui permet de gérer des tâches (todo list) où l'utilisateur peut ajouter, marquer comme terminée, supprimer, et filtrer des tâches. L'interface sera simple mais interactive, avec des actions pour organiser les tâches quotidiennes.

**Fonctionnalités Requises :**

1. **Ajouter une tâche** :
   * L'utilisateur doit pouvoir saisir une tâche dans une zone de texte.
   * Un bouton **"Ajouter"** permet d'ajouter cette tâche à la liste.
   * Une fois ajoutée, la tâche est affichée dans une liste avec un bouton pour la marquer comme terminée ou la supprimer.
2. **Marquer une tâche comme terminée** :
   * Chaque tâche de la liste doit avoir un bouton **"Terminer"** qui, lorsqu'il est cliqué, change l'apparence de la tâche (par exemple, texte barré ou couleur différente) pour indiquer qu'elle est terminée.
3. **Supprimer une tâche** :
   * Chaque tâche doit également avoir un bouton **"Supprimer"** pour retirer la tâche de la liste.
4. **Filtrer les tâches** :
   * Ajouter des boutons de filtrage pour afficher :
     + Toutes les tâches.
     + Les tâches terminées.
     + Les tâches non terminées.

**Technologies à utiliser :**

* **React.js** : Utilisation des composants et des hooks pour gérer les tâches.
* **CSS** : Styliser l'application pour rendre l'interface agréable.

**Captures d’écran**

|  |  |
| --- | --- |
|  |  |

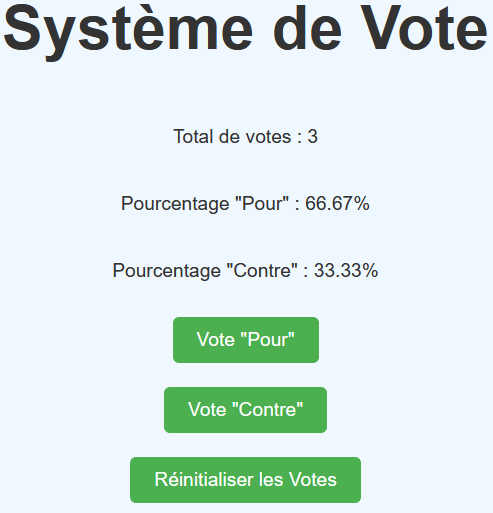
**Fonctionnalités supplémentaires (optionnel) :**

* **Édition d'une tâche** : Permettre à l'utilisateur de modifier une tâche existante.
* **Priorités** : Ajouter une fonctionnalité pour assigner une priorité (basse, moyenne, haute) à chaque tâche et les organiser en fonction de la priorité.
* **Affichage des statistiques** : Montrer combien de tâches qui sont terminées et combien restent à faire.

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰21**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

**En utilisant Usereducer créer le composant suivant :**

****

**TP 2 :**

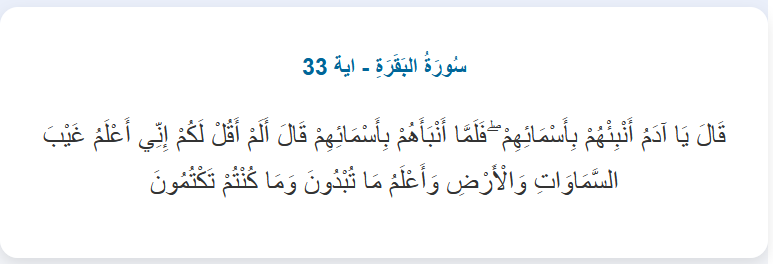
**Créer un composant qui retourne les données Météo suivantes(en utilisant Axios)**

******

***API :* *https://api.open-meteo.com/v1/forecast?latitude=${10}&longitude=${10}&current\_weather=true***

**TP 3 :**

**Créer un composant React qui affiche les versets du Quran(en utilisant Axios)**

****

***API : https://api.alquran.cloud/v1/ayah/${40}/ar***

**TP 4 :**

**Créer un composant React qui affiche le conseil du jour(en utilisant Axios)**

****

***API :* *https://api.adviceslip.com/advice***

**TP 5 :**

**Créer un composant React qui affiche les information des pays(en utilisant Axios)**

****

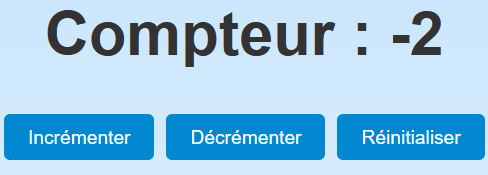
***API :* *https://restcountries.com/v3.1/all***

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰22**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

**Contexte : développer une application simple en utilisant React et Redux pour gérer l’état d’un compteur, avec trois boutons pour :**

* **Incrémenter le compteur de 1.**
* **Décrémenter le compteur de 1.**
* **Réinitialiser le compteur à 0.**

****

**TP 2 :**

**Contexte : développer une application simple en utilisant React et Redux pour gérer un message. Cette application permettra d'afficher, de modifier et de réinitialiser ce message.**

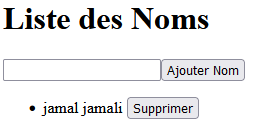
**Objectifs :**

1. **Afficher le message actuel à l'écran.**
2. **Modifier le message en saisissant un texte dans un champ d'entrée.**
3. **Effacer le message en réinitialisant son contenu à une chaîne vide.**

****

**TP 3:**

**développer une application simple en utilisant React et Redux pour gérer les noms. Cette application permettra d'ajouter, d’afficher et de modifier les noms.**



**TP 4:**

**Créer une petite application React qui utilise Redux pour gérer deux aspects de l'état de l'application :**

**Un message affiché dans l'interface utilisateur que l'utilisateur peut modifier.**

**Un compteur qui permet à l'utilisateur d'incrémenter et de décrémenter une valeur.**

|  |  |
| --- | --- |
| **INSTITUT SPECIALISE DE TECHNOLOGIE APPLIQUEE DRIOUCH** | **Série N⁰23**  **Formation :** DD 201  **Prof :** SAID MAHDAD  **Module :** développer en Front-end  **Nom et prénom :** |

**TP 1 :**

**Description :**  
Créez une application pour gérer une liste d'utilisateurs avec les fonctionnalités suivantes :

* Ajouter un utilisateur
* Supprimer un utilisateur
* Afficher les détails d'un utilisateur



**Étapes :**

1. Créez un **reducer** qui gère une liste d'utilisateurs
2. Implémentez trois actions : ADD\_USER, REMOVE\_USER, et SELECT\_USER.
3. Affichez les utilisateurs et leurs détails dans des composants connectés à Redux.



**TP 2 :**

**Description:**  
Développez une application où les utilisateurs peuvent :

* Ajouter des articles au panier
* Supprimer des articles du panier
* Afficher le total des articles

**Étapes :**

1. Créez un **reducer** pour gérer le panier.
2. Configurez un **store** et connectez-le à un composant React.
3. Gérez les actions ADD\_TO\_CART et REMOVE\_FROM\_CART.

