

Atelier 01

Objectif d'apprentissage :

- Etre capable de déclarer des composants sous format fonction et sous format classe
- Utilisation de JSX

Exercice 1 :

1. Créer un nouveau projet React avec l'outil create-react-app
2. Lancer votre projet
3. Dans le fichier App.js, créer les composants suivants :
 - a. Un composant nommé Pht, sous format fonction qui va rendre l'interface suivant :

Prix Hors taxe :

- b. Un composant nommé TVA, sous format fonction qui va rendre l'interface suivant :

Taux TVA :

- c. Un composant nommé Buttons, sous format fonction qui va rendre l'interface suivant :

- d. Un composant nommé Res, sous format fonction qui va rendre l'interface suivant :

Resultat :

- e. Un composant nommé CalculPTTC, sous format fonction qui va appeler les autres fonctions et va rendre l'interface suivant :

Calcul de PTTC

Prix Hors taxe :

Taux TVA :

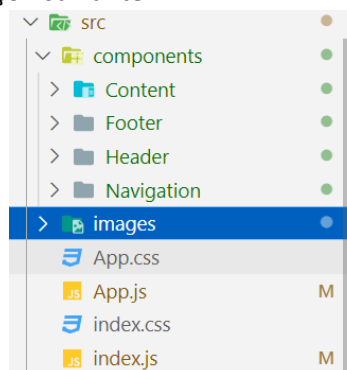
Resultat :

- f. Appeler le composant CalculPTTC dans la fonction App.js
4. Dans le mode développeur, vérifier les composants de votre interface.
 5. Créer un dossier nommé composants dans le dossier source puis tous les composants dans des fichiers séparés.
 6. Ajouter des styles pour afficher votre interface sous la forme suivante :

7. Refaire les mêmes questions précédemment effectuées, en déclarant les composants de la question 3 sous format classe.

Exercice 2 :

1. Créer un nouveau projet React avec l'outil create-react-app
2. Lancer votre projet
3. Structurer votre projet de la façon suivante :



- Dans le dossier src, On crée le dossier components pour déclarer les composants de notre application et le dossier images pour les images qui seront utilisés dans notre application.
- Dans le dossier components créé précédemment, créer 4 dossiers avec le nom : Content, Footer, Header et Navigation.

4. Dans le dossier Header, créer les éléments suivants :
 - a. Un composant nommé Header, sous format fonction qui va rendre l'interface suivante :



- b. Un fichier style nommé header.css qui va déclarer les styles suivants :
 - i. Le conteneur des éléments HTML de ce composant :
 1. Ajouter l'image **logo.png** en arrière-plan de cet élément.

2. Marges intérieurs de 10px
 3. Marges extérieurs gauche et droite de nulle et marges extérieurs gauche et droite de 20px.
 4. Ajoutez une hauteur de 80px.
 5. Ajoutez une bordure normale (solid) autour de l'élément.
 6. Réglez la largeur de la bordure à 3px.
 7. Réglez la couleur de la bordure.
 8. Ajoutez des arrondis de 10px à la bordure
 - ii. Le titre du site
 1. Marge extérieurs de 180px.
 2. Le texte doit être en gras, avec une taille de 20px et avec une couleur de #d47b7b
 - iii. Le slogan
 1. Marge extérieurs de 180px.
 2. La couleur du texte est #2622a2
5. Dans le dossier Navigation, créer les éléments suivants :
- a. Un composant nommé Navigation, sous format fonction qui va rendre l'interface suivante :



- b. Un fichier style nommé navigation.css qui va déclarer les styles de ce composant.
6. Dans le dossier Content, créer les éléments suivants :
 - a. Un composant nommé Content, sous format fonction qui va rendre l'interface suivante :

Titre du content

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna. Nunc viverra imperdiet enim. Fusce est. Vivamus a tellus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin pharetra nonummy pede. Mauris et orci. Aenean nec lorem. In porttitor. Donec laoreet nonummy augue. Suspendisse dui purus, scelerisque at, vulputate vitae, pretium mattis, nunc. Mauris eget neque at sem venenatis eleifend. Ut nonummy.

- b. Un fichier style nommé navigation.css qui va déclarer les styles de ce composant.
7. Dans le dossier Content, créer les éléments suivants :
 - a. Un composant nommé Footer, sous format fonction qui va rendre l'interface suivante :

Copyright ISMO - Année 2022/2023
 - b. Un fichier style nommé navigation.css qui va déclarer les styles de ce composant.
8. Dans le fichier App.js, créer un composant de type fonction pour appeler les autres composants déjà déclarés sous format suivante (Ajouter vos styles dans le fichier App.css) :

