Démo + explications React Native pour débutant :

<https://www.youtube.com/watch?v=K3D2rjAUQ3o>

Le code react native se compose principalement de composants => un composant est une fonction qui renvoi un rendu visuel (JSX (ressemble à du HTML)) (6:58 dans la vidéo)

Créer le répertoire de travail :

* *npx create-react-app <nom\_du\_répertoire>*

/!\ à installer dans le répertoire de travail quand on fait un projet en react (dans vsc) :

* *npm i -s react-dom react-router-dom node-sass@4.14.1*

Dans notre projet on doit avoir un répertoire **nodes\_modules** (contient toutes les librairies nécessaires pour le projet, **public** pour le stockage d’éléments statiques utiles pour le projet (images par exemple) et le fichier index.html qui contiendra tout le code qui fera fonctionner l’interface web et c’est plus exactement à l’intérieur de la balise <div /> de <body /> qui contiendra tout le code react et **src** (16:46 explications détaillées).

Ce qui serait intéressant c’est de créer une dossier “pages” dans le **src** qui va contenir tous les composants qui vont faire office de pages web.

Télécharger les plugs-in “Reactjs code snippets” et “Simple react snippets” afin d’avoir des propositions de code lorsque l’on commence à écrire un bout de code.

=> pour tester :

* allez dans le script App.js
* supprimez tout le code
* tapez “rsc” puis Entrée, le plug-in aura permi d’écrire une partie du code

(Ctrl + Z Pour revenir en arrière.)

Faire une navigation : (31:30)

/!\ si ‘react-router-dom’ >= v6.0 ⇒ remplacer Switch par Routes et puis

<Route path='/' component={Home}/> par <Route path='/' element={<Home />}/>

Quand on tape localhost:3000 dans la barre d’URL, le navigateur exécutera le code html de la page Home de notre répertoire.

Le but à présent est de pouvoir naviguer de page en page sans avoir à écrire l’url à la main, c'est-à-dire passer par des liens.

On va créer dans notre src un dossier “components” qui va contenir le code de la barre de navigation :

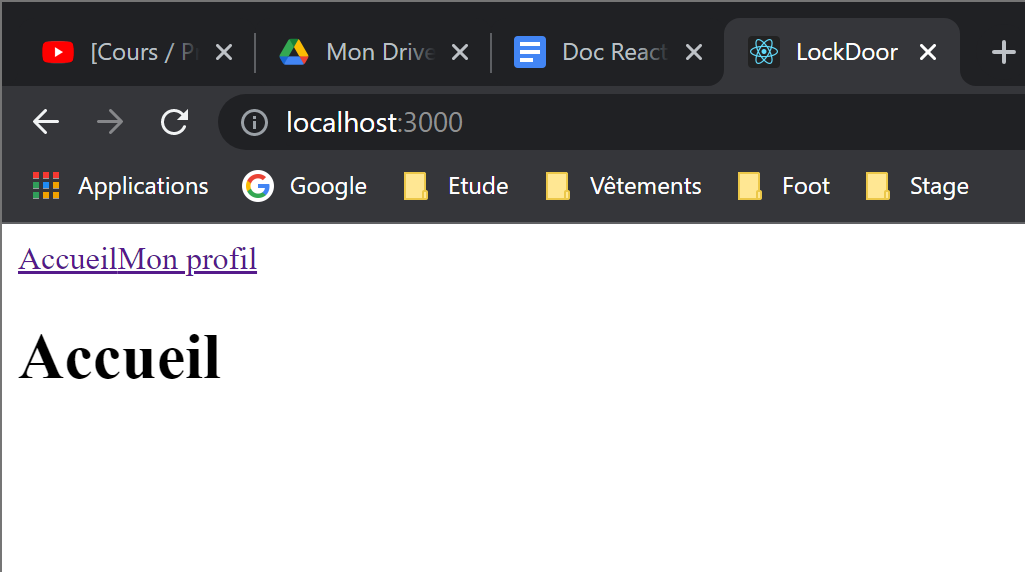
(rsc → raccourci pour créer un nouveau components)

(38:13) importer la librairie “NavLink” et faire les lien qui nous intéresse sans la division NavLink : <NavLink exact to="/myProfil">

Mon Profil

</NavLink>

Ensuite, pour chaque page créer, il faut importer le composant “Navigation” que l’on vient de créer dans chaque page juste après chaque balise div.



On aperçoit donc la barre de navigation en haut de la page web. Maintenant on va s’intéresser à comment identifier le lien sur lequel on a cliqué. En d’autres termes, comment voir visuellement la page sur laquelle on est (41:38) :

On ajoute activeClassName="nav-active" dans notre code Navigation.js dans la balise NavLink. Puis on va, dans le src créer un dossier “style” qui va nous permettre d’y stocker des fichiers .scss, on va pouvoir créer un index.scss et directement après, importer ce fichier dans l’index.js du projet.

Puis copier coller ce [code](https://github.com/JustFS/react-basics/blob/d335e8eef55ba201ca08b6d653c1e50adcd64599/styles/_settings.scss) dans \_settings.scss afin de rendre la page plus esthétique. Il faut importer “\_settings.scss” dans l’index.scss puis rafraîchir la page web pour constater que le style à bien été pris en compte.

On peut maintenant se créer un petit dossier components dans le dossier style qui va contenir le style de chaque composants de la page web en l'occurrence ici, on veut styliser la barre de navigation, pour cela on va aller dans le dossier components que l’on vient de créer et y ajouter un fichier “\_navigation.scss”

Puis mettre le code [suivant](https://github.com/JustFS/react-basics/blob/d335e8eef55ba201ca08b6d653c1e50adcd64599/styles/components/_navigation.scss) dans ce fichier, cela permet d’identifier quelle page est active dans notre barre de navigation.

/!\ Ne pas oublier les différents import nécessaires dans les components récemments créés.

Gestion des images sur la page web (53:11)

npm i -s axios ⇒ permet de faire des fetch

trier (1:33:04)

transformer l’array en objet + voir le code de Profil.js pour les axios et les Array

reprendre à 1:49:00