

Interface Utilisateur (UI)

Dans ce TP, nous utiliserons la bibliothèque de composants Grommet (basée sur React) pour concevoir une interface utilisateur permettant d'afficher les données du TP précédent. La documentation de Grommet est disponible en ligne :

<https://v2.grommet.io>

Cette bibliothèque permet le développement d'une interface utilisateur moderne, adaptative (*responsive*) et accessible. Un outil en ligne associé, Grommet Designer, permettant la conception visuelle (*visual designer*) d'un prototype est aussi disponible :

<https://designer.grommet.io>

Nous l'utiliserons pour la suite de ce TP :

1. Ouvrir l'exemple Card dans Grommet Designer.
2. Cliquer sur Repeater à gauche et, à droite, donner à count la valeur 1.
3. Cliquer sur Card Box à gauche, puis sur Box^V et reset. Ensuite, à droite :
 - cliquer sur background^V puis color et choisir white
 - cliquer sur round et choisir medium
 - cliquer sur margin et choisir aussi medium
4. Cliquer à nouveau sur "Card Box" à gauche, puis sur Box^V et show code... en haut à droite. La fenêtre qui s'ouvre vous affiche le code React/Grommet correspondant à l'objet actuellement sélectionné.
5. Cliquer sur Card^V puis share à gauche, puis Generate code, et vous obtenez le code complet de l'application. Sélectionner et copier ce code.
6. Créer une application React/Grommet en suivant le tutorial en ligne (cf Annexe).
7. Remplacer le contenu de App.tsx par le code copié auparavant. Tester l'application.
8. Supprimer le thème dans App.tsx et tester à nouveau l'application. Supprimer le paramètre full et tester à nouveau l'application.
9. Cliquer sur Card^V puis share à gauche, puis configure. Dans Data, conserver data comme Name, ajouter comme Source :
<https://jsonplaceholder.typicode.com/users>
10. Cliquer sur Repeater à gauche et, à droite, donner à dataPath à la valeur data (mais laisser la valeur de count à 1).

11. Cliquer sur le nom de la première carte et, à droite :

- donner à `name` la valeur `Name`, puis remplacer le texte par `{name}`
- donner à `name` la valeur `Email`, puis remplacer le texte par `{email}`
- remplacer le texte de la `Description` par `{company.catchPhrase}`

12. Dans l'application, créer un fichier `Users.ts` contenant les déclarations de type et de la constante `users` du TP précédent, et ajouter à la fin les lignes suivantes :

```
export type { Geo, Address, Company, User }
export { users }
```

13. Ajouter les imports correspondants dans `App.tsx` :

```
import type { Geo, Address, Company, User } from './Users'
import { users } from './Users'
```

14. Encadrer la `Box` correspondant à une carte par :

```
{users.map(({ name, email, company }: User) =>
  ...
)}
```

où `"..."` est remplacé par le code de cette `Box`.

15. Revenir à `Grommet Designer` et ajouter un `Accordion` dans le bas d'une carte (juste au-dessous de la `Description`), puis une `AccordionPanel` à l'intérieur. Donner la valeur `Address` au label et remplir le panel avec des champs `Text` contenant l'adresse de l'utilisateur.

16. Déplacer le code pour la `Box` correspondant à une carte dans un fichier `Card.tsx`. Le composant fonctionnel (FC) contiendra alors (en plus des autres imports) :

```
import { FC } from 'react'

const Card: FC<{ name: string, email: string, company: Company }> =
  ({ name, email, company }) => {
    return (
      ...
    )
  }

export default Card;
```

17. Modifier aussi le code de `App.tsx` ainsi :

```
{users.map(({ name, email, company }: User) =>
  <Card name={name} email={email} company={company} />
)}
```

Annexe

La documentation officielle de React se trouve ici :

<https://react.dev/learn/start-a-new-react-project>

Création d'une nouvelle application React/Grommet

Le tutoriel en ligne ci-dessous décrit comment créer un projet React/Grommet :

<https://github.com/grommet/grommet-starter-new-app>

Ce tutoriel utilise `create-react-app`, qui n'est plus maintenu. D'autres outils, plus modernes, sont désormais disponibles comme ceux mentionnés sur la page de React. Nous utiliserons ici une solution alternative, plus adaptée pour ces TP, qui s'appelle « vite » :

<https://vite.dev/guide>

La création d'une application React/Grommet avec vite est décrite dans ce tutoriel :

<https://cedric.cnam.fr/sys/crolard/enseignement/NFP119/grommet-cards-vite.html>

Installation et lancement d'une application React/Grommet

Il est aussi possible d'installer directement une archive pré-existante d'un projet React/Grommet avec la commande suivante :

```
npm install
```

Le lancement de l'application en mode « développement » se fait alors ainsi :

```
npm run dev
```

VS Code doit aussi normalement reconnaître la configuration vite (et proposer les tâches npm correspondantes).