

- 1- En esta práctica, he tomado como ejemplo el ejercicio anterior en donde se crea varias funciones para luego ser llamadas a la función principal.

```

function UsersList() { Show usages new *
  const users: {[role: string, name: string]: {id: number, name: string}} = [
    { id: 1, name: 'Jose', role: 'Web Developer' },
    { id: 2, name: 'Estefania', role: 'Web Designer' },
    { id: 3, name: 'Rubén', role: 'Team Leader' },
    { id: 4, name: 'alex1', role: 'Team' },
    { id: 5, name: 'alex2', role: 'Team Leader' }
  ]

  return (
    <>
      <p>Lista de usuarios activos:</p>
      <ul className="list-alum">
        {users.map((user) => (
          <li key={user.id}>
            {user.name} - {user.role}
          </li>
        ))}
      </ul>
    </>
  )
}

function App() { Show usages ± alex0 *
  return (
    <div>
      <lowerText />
      <Element />
      <usersList />
    </div>
  )
}

```

- 2- Para añadir los estilos, solo he tenido que añadirles nombre a las variables del HTML, para luego ser llamadas en el nuevo archivo CSS y poder añadirles el estilo.

```

.text-lowercase{
  color: red;
}
.welcome{
  color:blue;
}
.list-text{
  color:blueviolet;
  font-size: 50px;
}
.list-alum{
  color: green;
  font-size: 30px;
}

```

## 1- PREGUNTAS EXTRAS

¿Qué es JSX con tus palabras?.

Es una extensión de sintaxis de JavaScript, muy similar al HTML. Utiliza etiquetas para luego ser utilizadas.

¿Por qué usamos {} dentro de JSX? Pon un ejemplo.

Esto nos permite añadirlo al código de JavaScript

¿Para qué sirve la prop key en listas?.

Se utiliza para dar una identidad única y estable a una lista de elementos en React, esto le permite a React a saber cuál se ha añadido, eliminado o modificado.

¿Por qué usamos className y no class en React?

Se utiliza className y no class para evitar conflictos, ya que en JavaScript tiene la palabra reservada de class, por eso se utiliza className.