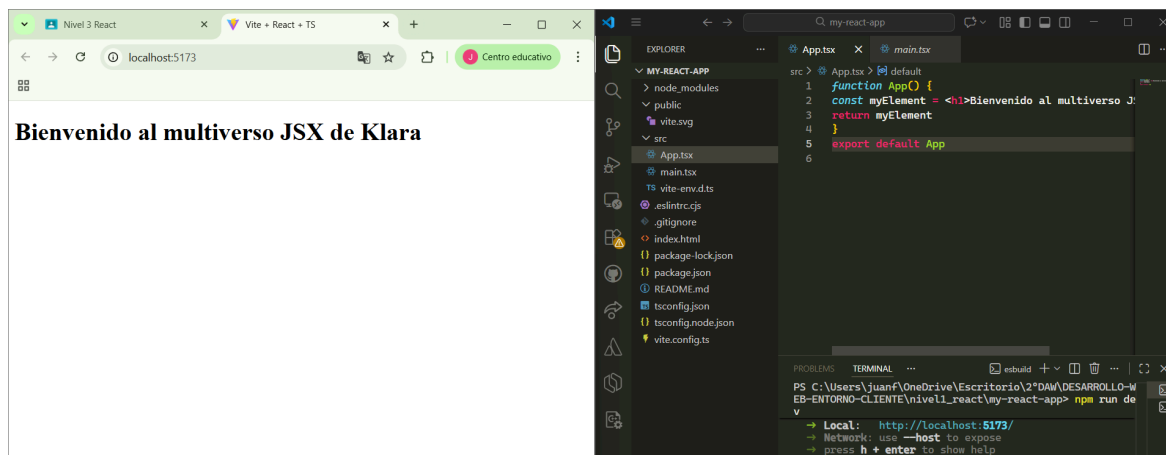


REACT NIVEL 3



PARTE A:.....	2
PARTE B:.....	2
PARTE C:.....	3
PARTE D:.....	4
PREGUNTAS EXTRA PARA EL PDF.....	6

PARTE A:

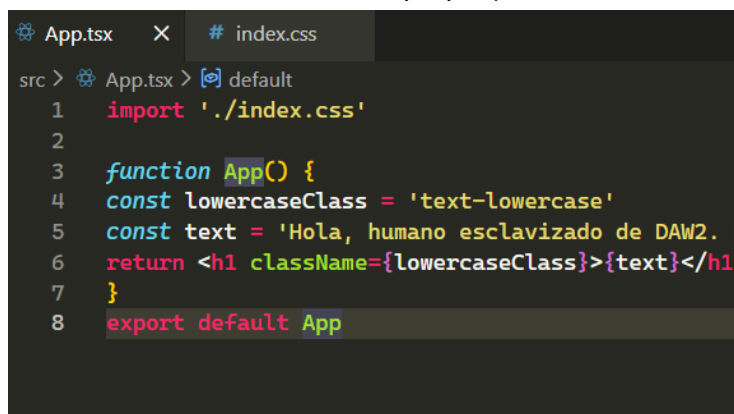


(Ilustración 1. Código mostrando la página web corriendo con el resultado, y el código dónde se han realizado los cambios para crear un elemento JSX y devolverlo. Elaboración propia.)

PARTE B:

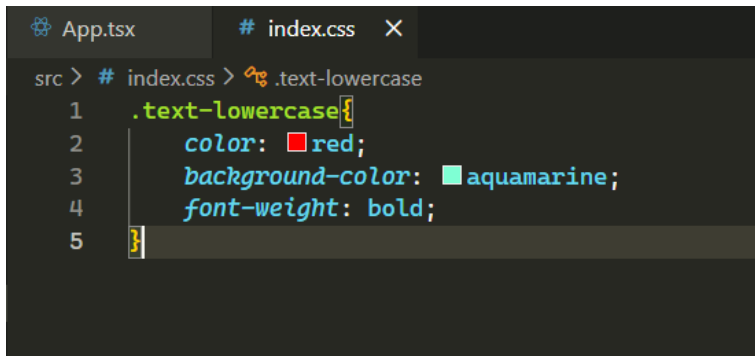


(Ilustración 2. Captura de pantalla con los cambios realizados, creando un `index.css` y añadiendo estilos. Elaboración propia.)



(Ilustración 3. Captura de pantalla del código `App.tsx` añadiendo una `className`. Elaboración propia.)

PARTE C:



```
App.tsx  # index.css X
src > # index.css > .text-lowercase
1  .text-lowercase{
2      color: red;
3      background-color: aquamarine;
4      font-weight: bold;
5  }
```

(Ilustración 4. Captura de pantalla del código de index.css que le da color a nuestro App.tsx. Elaboración propia)

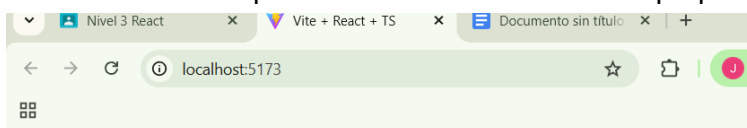


```
App.tsx > default
import './index.css'

function App() {
  const users = [
    { id: 1, name: 'Jose', role: 'Web Developer' },
    { id: 2, name: 'Estefanía', role: 'Web Designer' },
    { id: 3, name: 'Rubén', role: 'Team Leader' },
    { id: 4, name: 'Antoñico', role: 'Profesor DAW2' },
    { id: 5, name: 'Juanaco', role: 'Alumno de Antoñico' },
    { id: 6, name: 'Juanky', role: 'Alumno de Antoñico' },
  ]
  return (
    <>
    <p>Lista de usuarios activos:</p>
    <ul>
      {users.map(function (user) {
        return (
          <li key={user.id}>
            {user.name} — {user.role}
          </li>
        )
      })}
    </ul>
    </>
  )
}

export default App
```

(Ilustración 4. Captura de pantalla del código de App.tsx con más registros haciendo uso del nombre de compañeros de clase. Elaboración propia.)



Lista de usuarios activos:

- Jose — Web Developer
- Estefanía — Web Designer
- Rubén — Team Leader
- Antoñico — Profesor DAW2
- Juanaco — Alumno de Antoñico
- Juanky — Alumno de Antoñico

(Ilustración 5. Captura de pantalla de nuestra página web de React dónde se aprecian los cambios que hemos realizado previamente en App.tsx. Elaboración propia)

PARTE D:

```
# index.css  App.tsx  X
src > App.tsx > App > users > role
1  import './index.css'
2
3  function App() {
4    const users = [
5      { id: 1, name: 'Jose', role: 'WebDeveloper' },
6      { id: 2, name: 'Estefania', role: 'WebDesigner' },
7      { id: 3, name: 'Rubén', role: 'TeamLeader' },
8      { id: 4, name: 'Antoñico', role: 'ProfesorDAW2' },
9      { id: 5, name: 'Juanaco', role: 'AlumnadoAntoñico' },
10     { id: 6, name: 'Juanky', role: 'AlumnadoAntoñico' },
11   ]
12   return (
13     <>
14     <p>Lista de usuarios activos:</p>
15     <ul>
16       {users.map(function (user) {
17         return (
18           <li key={user.id} className={user.role}>
19             {user.name} - {user.role}
20           </li>
21         )
22       })}
23     </ul>
24   )
25 }
26
27 export default App
```

(Ilustración 6. Captura de pantalla del código de `App.tsx` con las últimas modificaciones para que funcione correctamente como nos pide el ejercicio. Elaboración propia.)

```
# index.css  X  App.tsx
src > # index.css > .WebDesigner
1  .text-lowercase{
2    color: red;
3    background-color: aquamarine;
4    font-weight: bold;
5  }
6
7  .AlumnadoAntoñico{
8    color: red;
9    background-color: aquamarine;
10   font-weight: bold;
11 }
12
13 .ProfesorDAW2{
14   color: aquamarine;
15   background-color: red;
16   font-weight: lighter;
17 }
18
19 .WebDeveloper {
20   color: pink;
21   background-color: antiquewhite;
22 }
23
24 .TeamLeader {
25   color: pink;
26   background-color: antiquewhite;
27 }
28
29 .WebDesigner{
30   color: pink;
31   background-color: antiquewhite;
32 }
```

(Ilustración 7. Captura de pantalla de nuestro `index.css` dándole color a nuestros compañeros. Esto es posible gracias a `className = {user.role}`, que hace que podamos referenciar cómodamente en nuestro `css` en función del rol de nuestros compañeros. Elaboración propia.)

Lista de usuarios activos:

- Jose — WebDeveloper
- Estefanía — WebDesigner
- Rubén — TeamLeader
- Antón — ProfesorDAW2
- Juanaco — AlumnodeAntón
- Juanky — AlumnodeAntón

(Ilustración 8. Captura de pantalla realizada de nuestro proyecto React con los cambios ya realizados en nuestro código `App.tsx` y `index.css` Elaboración propia.)

PREGUNTAS EXTRA PARA EL PDF

PREGUNTA 1: ¿Qué es JSX con tus palabras?

JSX es una forma de escribir componentes en React usando una sintaxis parecida a HTML dentro de JavaScript para que el código sea más fácil de leer.

PREGUNTA 2: ¿Por qué usamos `{}` dentro de JSX? Pon un ejemplo.

Se usan para meter código JavaScript dentro de JSX. Ejemplo: `<p>nombre</p>`

PREGUNTA 3: ¿Para qué sirve la prop `key` en listas?

Sirve para que React identifique cada elemento de una lista y pueda actualizarla mejor.

PREGUNTA 4: ¿Por qué usamos `className` y no `class` en React?

Porque `class` es una palabra reservada en JavaScript y React usa `className` para evitar conflictos.