

El objeto de esta práctica es afianzar, reforzar y consolidar los conocimientos teóricos presentados en la lección.

Al finalizarla, el estudiante:

- Habrá creado un componente que define un contexto.
- Habrá usado el contexto en un subcomponente.

Objetivos

El objetivo de la práctica es mostrar cómo trabajar con el contexto. Por una parte, crearemos un componente que define su propio contexto. Y por otra parte, un subcomponente que lo use. Definiremos una aplicación que muestre bandas de música en una tabla. La tabla será el componente que define el contexto y la fila el que lo usa.

Preparación del entorno

Para comenzar, crearemos un proyecto de aplicación **React** mediante el generador de **Justo**:

1. Abrir una la consola.
2. Crear el directorio de la práctica e ir a él.
3. Invocar el generador de **Justo**:

```
> justo -g react
```

Responder a las preguntas realizadas por el generador. A la hora de responder:

- Indique **N** cuando le pregunte si va a usar **React Router**.

4. Instalar las dependencias del proyecto:

```
> npm install
```

5. Invocar la tarea **build** del catálogo del proyecto para construir la aplicación:

```
> justo build
```

6. Abrir el archivo **dist/index.html** con el navegador.

Debe aparecer el mensaje Hello World!

Creación del componente padre

Ahora, vamos a crear un componente que representa la tabla de datos y, en el cual, definiremos un contexto al que podrán acceder sus subcomponentes:

1. Ir a la consola.
2. Crear un componente mediante el generador de **Justo**:

```
> justo -g react component
```

Tenga en cuenta lo siguiente a la hora de responder:

- Definir el componente como **BandTable**.
- Definirlo como inmutable. Pero podría ser mutable si fuera necesario.
- Definirlo como compuesto. Pero podría ser simple si fuera necesario.
- Definirlo mediante una clase.
- Indicar que se desea definir el contexto, para que así se añadan el método

`getChildContext()` y la propiedad estática `childContextTypes`.

- Indicar que no se desea usar el contexto en este componente.

3. Editar el archivo `app/components/BandTable.jsx`. Y echarle un vistazo detenidamente.
4. Añadir la propiedad `tags` al contexto, de tipo texto. Su valor será el valor de la propiedad homónima del componente:

```
getChildContext() {  
  return {  
    tags: this.props.tags  
  };  
}  
  
static get childContextTypes() {  
  return {  
    tags: React.PropTypes.string  
  };  
}
```

5. Definir su método `render()` como sigue:

```
render() {  
  return (  
    <table>  
      <thead>  
        <tr>  
          <th>Banda</th>  
          <th>Año</th>  
          <th>Tags</th>  
        </tr>  
      </thead>  
  
      <tbody>  
        <BandRow name="Love and Rockets" year="1985" />  
        <BandRow name="Lloyd Cole and the Commotions" year="1982" />  
      </tbody>  
    </table>  
  );  
}
```

Observe que la tabla muestra los *tags*, pero no se pasa mediante ninguna propiedad a las filas.

6. Importar el componente `BandRow`, que definiremos en breve:

```
import BandRow from "../BandRow";
```

7. Guardar cambios.

8. Editar el archivo `app/views/App.jsx`.

9. Importar el componente `BandTable`:

```
import BandTable from "../components/BandTable";
```

10. Ir al método `render()` y hacer que reproduzca un componente `BandTable`:

```
render() {  
  return (  
    <BandTable tags="indie" />  
  );  
}
```

11. Guardar cambios.

Creación del subcomponente

Ahora, vamos a definir el componente fila, el cual usa el contexto definido en la tabla:

1. Ir a la consola.
2. Definir un nuevo componente mediante el generador:

```
> justo -g react component
```

Tenga en cuenta lo siguiente en las respuestas:

- Nombre del componente: `BandRow`.
- Tipo de componente: `Immutable`.
- Tipo de estructura: `Simple`.
- Tipo de implementación: `Class`.
- Definir contexto: `N`.
- Usar contexto: `Y`.

3. Editar el archivo `app/components/BandRow.jsx`.

4. Ir a la propiedad estática `contextTypes` e indicar que usaremos la propiedad `tags` del contexto:

```
static get contextTypes() {
  return {
    tags: React.PropTypes.string
  };
}
```

5. Ir al método `render()` y definirlo como sigue:

```
render() {
  return (
    <tr>
      <td>{this.props.name}</td>
      <td>{this.props.year}</td>
      <td>{this.context.tags}</td>
    </tr>
  );
}
```

Observe que las dos primeras columnas de la fila muestran el valor de las propiedades `name` y `year`, mientras que la tercera muestra la propiedad `tags` del contexto.

6. Guardar cambios.

7. Ir a la consola.

8. Construir la aplicación y abrirla en el navegador:

```
> just build chrome
```

Debe aparecer algo como lo siguiente:

Banda	Año Tags
Love and Rockets	1985 indie
Lloyd Cole and the Commotions	1982 indie