

LABORATORIO

Trabajando con componentes en una aplicación React js

En este laboratorio vamos a implementar la página index.html utilizando componentes y manejando estado de éstos.

1. Trabajando con el componente Menu.js

Agregando un selector a index.css

Primero agregamos en el archivo index.css, el selector de class ítem y definimos sus propiedades. Consideramos que agregamos 6 hipervínculos.

```
# index.css X
src > # index.css > h1
19
20 .item{
21   width: 16.6%;
22   height: 30px;
23   padding-top: 10px;
24   text-align: center;
25   display: inline-block;
26   color: white;
27   background-color: gray;
28 }
29
30 .item:hover{
31   color: gray;
32   background-color: white;
33 }
34
```

Definiendo el selector de class: item

Figura 01: index.css
Fuente: Diseño propio

Trabajando con Menu.js

En la carpeta src, agregamos un nuevo Archivo (new File) y lo llamaremos Menu.js.

- En el archivo importar React y {Component}
- Defina un Array llamado ítems, el cual almacena la lista de elementos (tipo cadena)
- A continuación defina la clase Menu de tipo Component

```
# index.css JS Menu.js X JS index.js
src > JS Menu.js > ...
1 import React,{Component} from 'react';
2
3 const items=new Array("Index","Tradiciones","Costumbres","Museos",
4   "Mas Visitados","Contactenos");
5
6 class Menu extends Component{
7
8 }
9
10 export default Menu;
11
12
```

defina un Array de elementos: items

Defina la clase Menu de tipo Component

Figura 02: Menu.js
Fuente: Diseño propio

En la clase Menu:

- Defina el componente lista de tipo Array, state
- En el constructor, inicialice el valor de lista con un Array de hipervínculos <a>
- Al renderizar, envíe el valor del componente lista: {this.state.lista}



Figura 03: Menu.js
Fuente: Diseño propio

Renderizar index.js

A continuación, abrir el archivo index.js y definir el siguiente código:

- Importar el archivo Menu.js y lo llamaremos Menu
- Al renderizar la página importamos el contenido de Menu



Figura 04: Index.js
Fuente: Diseño propio

En la ventana de consola, ejecutar el proyecto React JS: **npm start sesion02react** y dar Enter, donde se visualiza el contenido de Menu.js

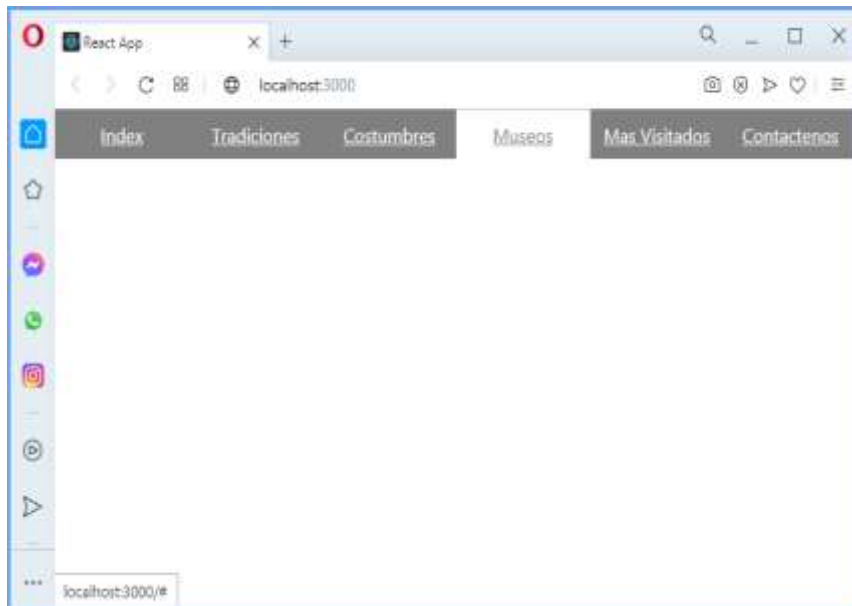


Figura 05: Index.html
Fuente: Diseño propio

2. Trabajando con el componente Header.js

Agregando selectores a index.css

Primero agregamos en el archivo index.css, los selectores de class container, h1cab, imgcab y definimos sus propiedades.

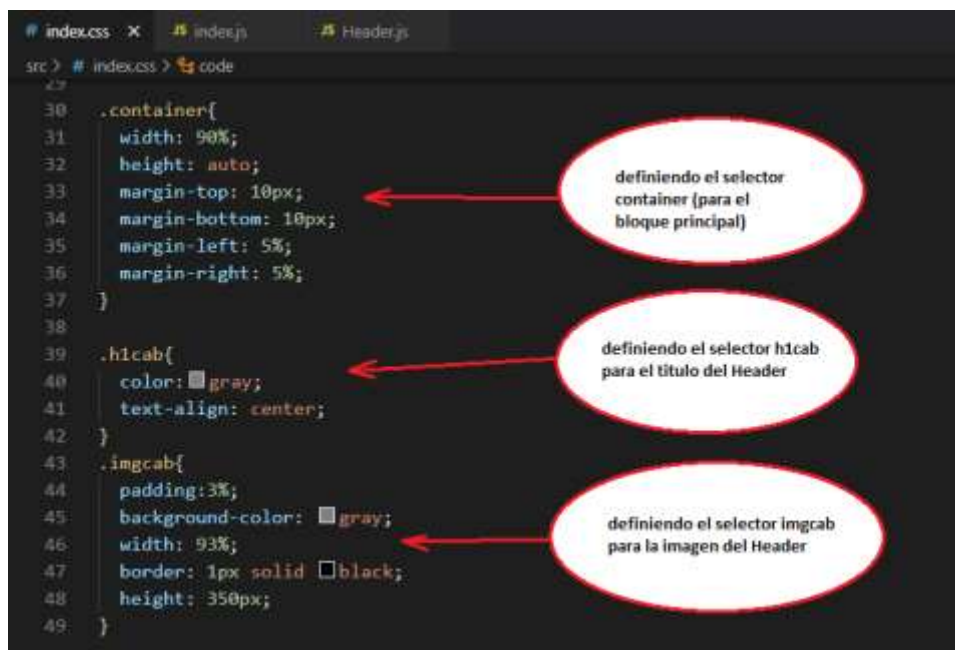


Figura 06: Index.css
Fuente: Diseño propio

Trabajando con Header.js

En la carpeta src, agregamos un nuevo Archivo (new File) y lo llamaremos Header.js.

- En el archivo importar React y {Component}
- Defina un Array llamado imagenes, el cual almacena la lista de direcciones de las imágenes las cuales se ubican en la carpeta img direccionada en **public**.
- A continuación defina la clase Header de tipo Component

```

1  import React,{Component} from 'react';
2
3  const imagenes=["/img/img1.jpg","/img/img2.jpg","/img/img3.jpg",
4  "/img/img4.jpg","/img/img5.jpg","/img/img6.jpg","/img/img7.jpg"];
5
6  class Header extends Component{
7
8  }
9
10 export default Header;
11

```

defina el Array de imagenes, contiene las ubicaciones de las imagenes del proceso

defina la clase Header de tipo Component

Figura 07: Header.js
Fuente: Diseño propio

En la clase Header:

- En el constructor, inicialice el valor del estado "c" a cero
- Al renderizar, ejecuta un setTimeout() donde, por cada segundo, actualizo el valor de "c" con un valor aleatorio entre 0 hasta 7 (longitud del array imágenes).

```

6  class Header extends Component{
7    constructor(props){
8      super(props);
9      this.state={
10       |  c:0,
11       |  };
12    };
13    state={
14      c:0
15    };
16    render(){
17      setTimeout(() => this.setState({
18        |  c: Math.floor(Math.random()*imagenes.length)
19        |  }),1000);
20    return(
21      <div>
22        <h1 className="h1cab">Un Viaje a las tradiciones del Perú</h1>
23        <img className="imgcab" src={imagenes[this.state.c]} />
24      </div>
25    );
26  }
27
28  export default Header;
29

```

por cada de 1 segundo "c", se le asigna (setState) un valor aleatorio entre 0 hasta tamaño de imagenes

Figura 08: Header.js
Fuente: Diseño propio

Renderizar index.js

A continuación, abrir el archivo index.js y definir el siguiente código:

- Importar los archivo Menu.js y Header.js
- Al renderizar la página importamos el contenido de Menu y Header, tal como se muestra

```
JS index.js x
src > JS index.js
1 import React from 'react';
2 import ReactDOM from 'react-dom';
3 import './index.css';
4
5 import Menu from './Menu';
6 import Header from './Header';
7
8 ReactDOM.render(
9   <div className="container">
10     <Menu/>
11     <Header/>
12   </div>,
13   document.getElementById("root")
14 );
15
```

Figura 09: index.js
Fuente: Diseño propio

Actualizar la página para visualizar el contenido de la class Header.

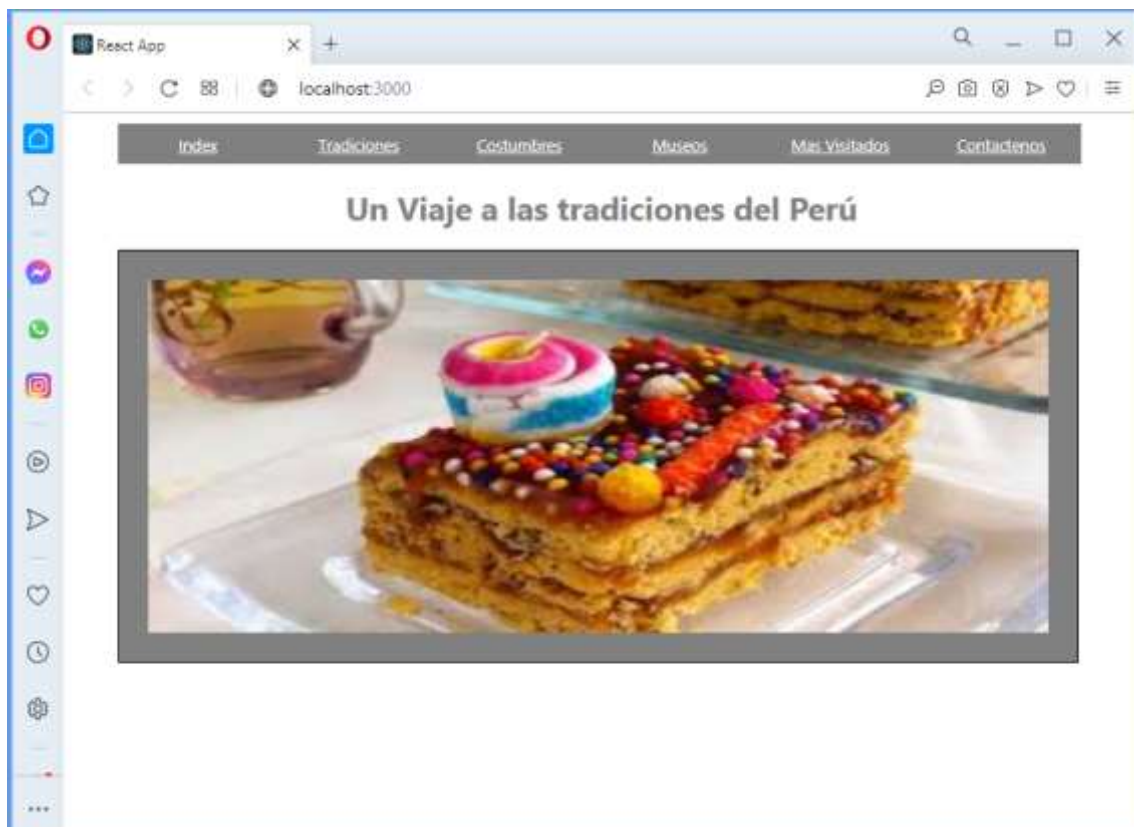


Figura 10: Index.html
Fuente: Diseño propio

3. Trabajando con el componente RegionFecha.js

Agregando selectores a index.css

Primero agregamos en el archivo index.css, el selector de class div-fecha, defina sus propiedades

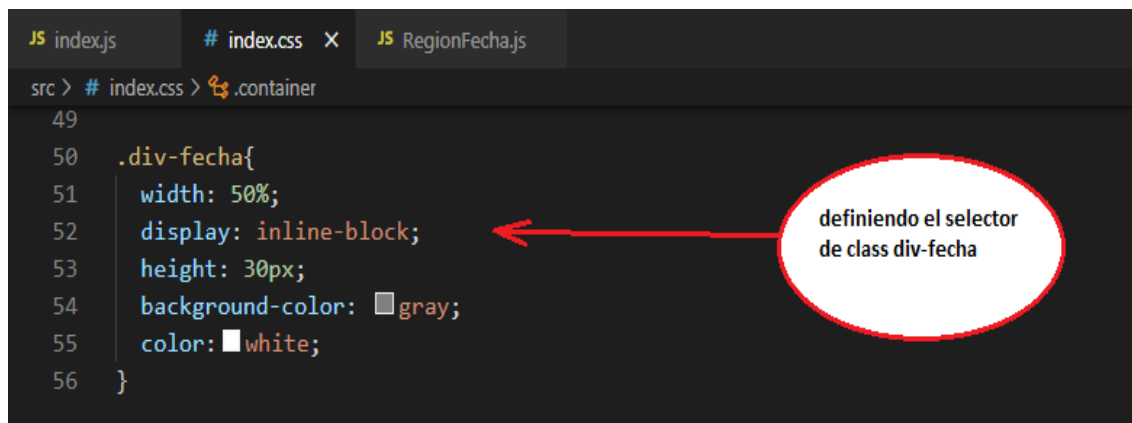


Figura 11: index.css
Fuente: Diseño propio

Trabajando con RegionFecha.js

En la carpeta src, agregamos un nuevo Archivo (new File) y lo llamaremos RegionFecha.js.

- En el archivo importar React y {Component}
- Defina una variable f que es la instancia de la clase Date()
- A continuación defina la clase RegionFecha de tipo Component

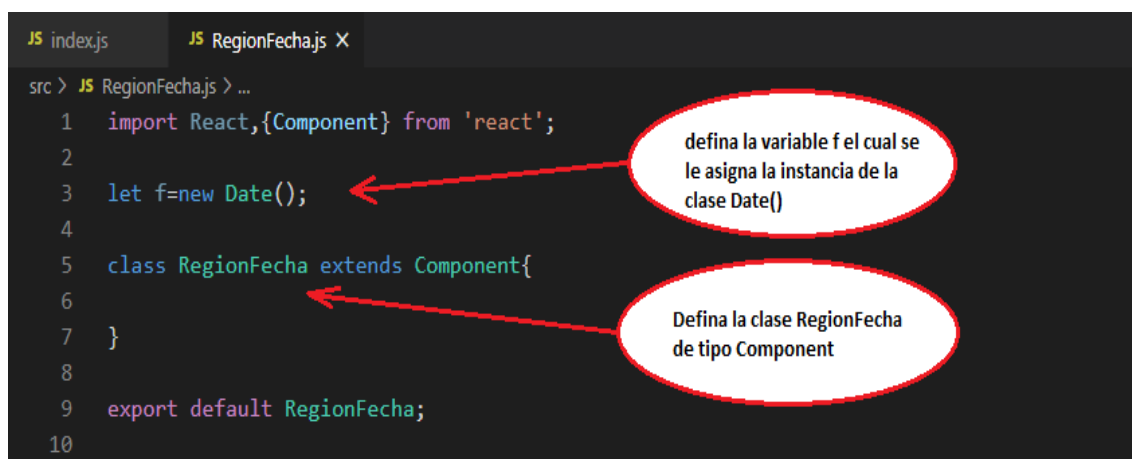


Figura 12: RegionFecha.js
Fuente: Diseño propio

En la clase RegionFecha:

- Defina en el state, los estados fecha y hora
- En el constructor, inicialice el valor del estado "fecha" a la fecha del sistema
- Al renderizar, ejecuta un setTimeout() donde, por cada segundo, actualizo el valor de "hora" con los valores de hora (h), minutos (m) y segundos (s) concatenados.

```

5  class RegionFecha extends Component{
6    constructor(props){
7      super(props);
8      this.state={
9        fecha:f.getDate()+"/"+(f.getMonth()+1)+"/"+f.getFullYear()
10     };
11   };
12   state={
13     fecha:null,
14     hora:null
15   };
16   render(){
17     f=new Date();
18     let h=f.getHours();
19     let m=(f.getMinutes())<10 ? "0"+f.getMinutes():f.getMinutes();
20     let s=(f.getSeconds())<10 ? "0"+f.getSeconds():f.getSeconds();
21     setTimeout(() => this.setState({
22       hora:h+": "+m+": "+s
23     }),1000);
24     return(
25       <div>
26         <div className="div-fecha">Hoy es {this.state.fecha}</div>
27         <div className="div-fecha">Son las {this.state.hora}</div>
28       </div>
29     );
30   }
31 }

```

al renderizar la clase,
actualizar el valor de
hora por cada segundo

Figura 13: RegionFecha.js

Fuente: Diseño propio

Renderizar index.js

A continuación, abrir el archivo index.js y definir el siguiente código:

- Importar los archivo Menu.js, Header.js y RegionFecha.js
- Al renderizar la página importamos el contenido de Menu, Header y RegionF, tal como se muestra

```

1  import React from 'react';
2  import ReactDOM from 'react-dom';
3  import './index.css';
4
5  import Menu from './Menu';
6  import Header from './Header';
7  import RegionF from './RegionFecha';
8
9  ReactDOM.render(
10    <div className="container">
11      <Menu/>
12      <Header/>
13      <RegionF/>
14    </div>,
15    document.getElementById("root")
16  );
17

```

Figura 14: index.js

Fuente: Diseño propio

Actualizar la página para visualizar el contenido de la class RegionFecha.



Figura 15: Index.html
Fuente: Diseño propio

4. Trabajando con el componente Tradiciones.js

Agregando selectores a index.css

Primero agregamos en el archivo index.css, el selector de class div-fecha, defina sus propiedades

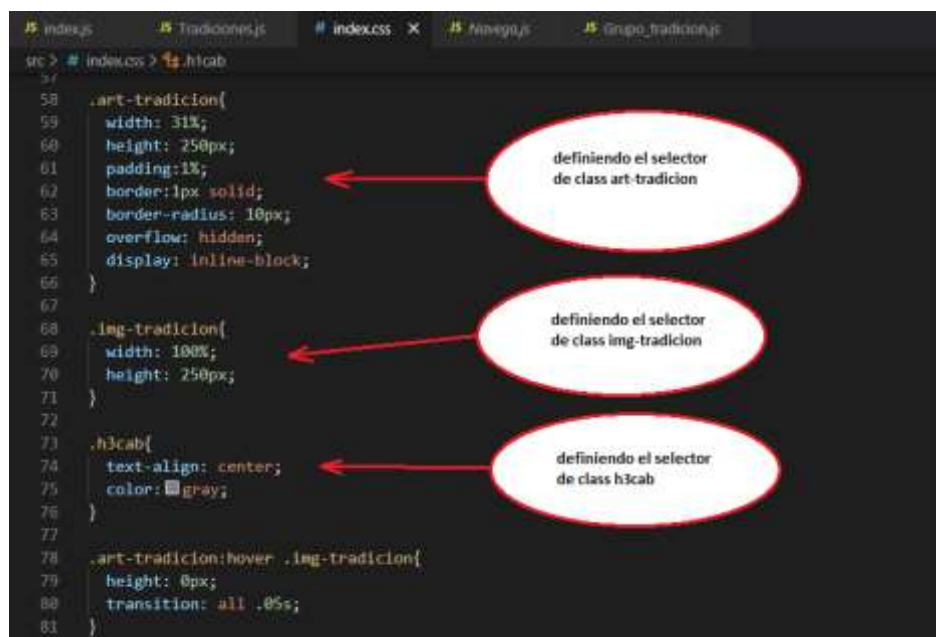
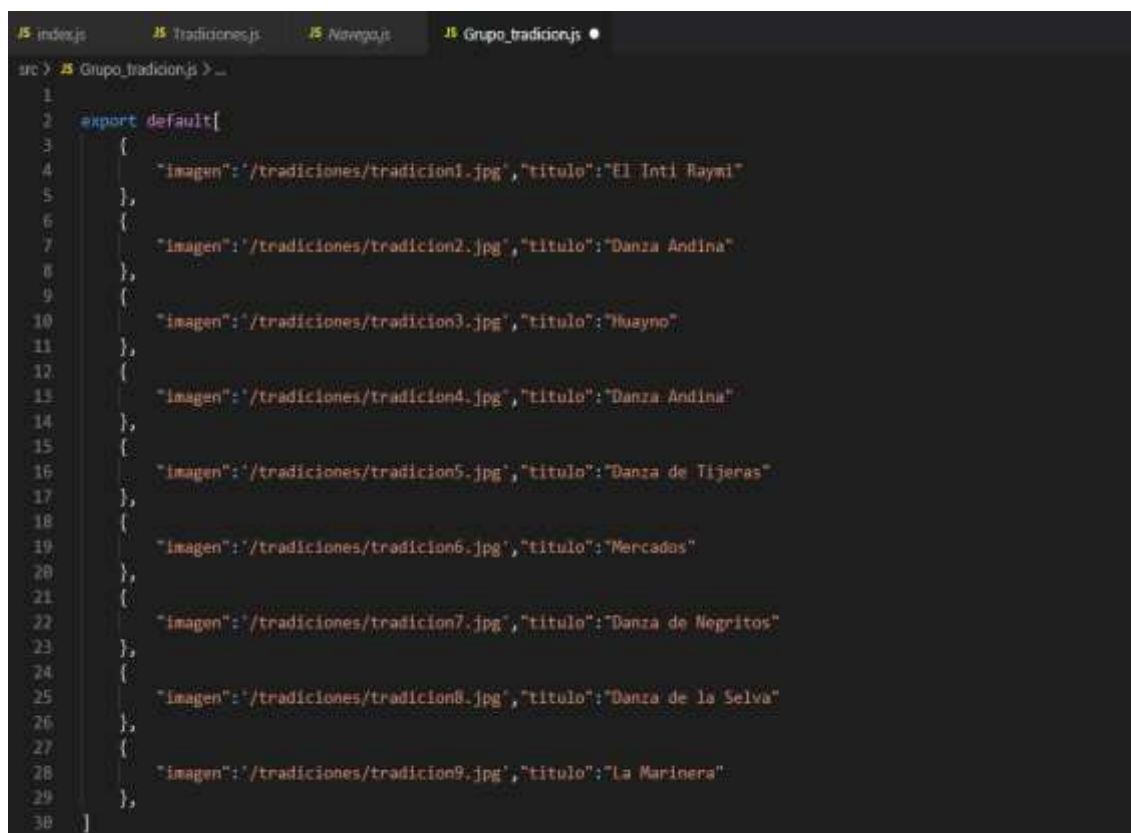


Figura 16: index.css
Fuente: Diseño propio

Definiendo un Array con Grupo_tradicion.js

En el archivo exportamos un Array de elementos, el cual esta conformado (para cada ítem), imagen y titulo, cada uno con su respectivo valor



```

1
2 export default [
3   {
4     "imagen": "/tradiciones/tradicion1.jpg", "titulo": "El Inti Raymi"
5   },
6   {
7     "imagen": "/tradiciones/tradicion2.jpg", "titulo": "Danza Andina"
8   },
9   {
10    "imagen": "/tradiciones/tradicion3.jpg", "titulo": "Huayno"
11  },
12  {
13    "imagen": "/tradiciones/tradicion4.jpg", "titulo": "Danza Andina"
14  },
15  {
16    "imagen": "/tradiciones/tradicion5.jpg", "titulo": "Danza de Tijeras"
17  },
18  {
19    "imagen": "/tradiciones/tradicion6.jpg", "titulo": "Mercados"
20  },
21  {
22    "imagen": "/tradiciones/tradicion7.jpg", "titulo": "Danza de Negritos"
23  },
24  {
25    "imagen": "/tradiciones/tradicion8.jpg", "titulo": "Danza de la Selva"
26  },
27  {
28    "imagen": "/tradiciones/tradicion9.jpg", "titulo": "La Marinera"
29  },
30 ]
  
```

Figura 17: Grupo_tradicion
Fuente: Diseño propio

Trabajando con Tradiciones.js

En la carpeta src, agregamos un nuevo Archivo (new File) y lo llamaremos Tradiciones.js.

- En el archivo importar React y {Component}, importar el archivo Grupo.
- A continuación defina la clase RegionFecha de tipo Component



```

1 import React, {Component} from 'react';
2 import Grupo from './Grupo_tradicion';
3
4 class Tradiciones extends Component{
5
6 }
7
8 export default Tradiciones;
9
  
```

importar el archivo Grupo_tradicion

Definir la clase Component Tradiciones

Figura 18: Tradiciones.js
Fuente: Diseño propio

En la clase Tradiciones:

- Defina en el state la lista
- En el constructor, inicialice la lista almacenando una colección de elementos leyendo el contenido de Grupo
- Al renderizar, visualizamos el contenido de lista

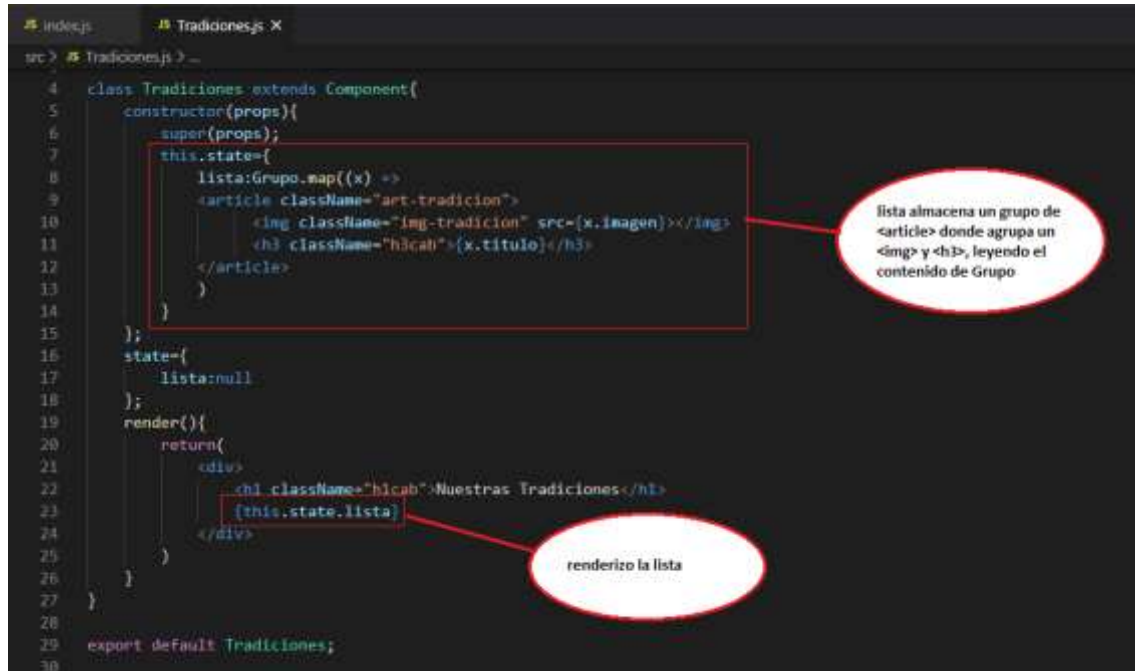


Figura 19: Tradiciones.js

Fuente: Diseño propio

Renderizar index.js

A continuación, abrir el archivo index.js y definir el siguiente código:

- Importar los archivo Menu.js, Header.js, RegionFecha.js y Tradiciones.js
- Al renderizar la página importamos el contenido de Menu, Header, RegionF y Tradiciones, tal como se muestra



Figura 20: index.js

Fuente: Diseño propio

Actualizar la página para visualizar el contenido de la class Tradicion.

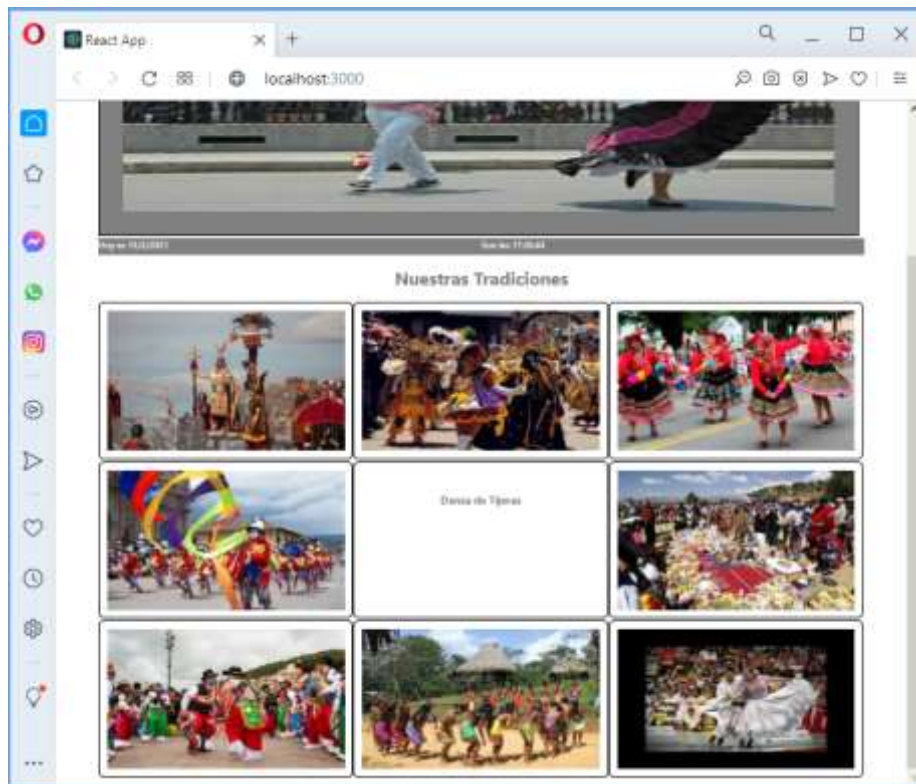


Figura 21: Index.html
Fuente: Diseño propio

5. Trabajando con el componente Footer.js

Agregando selectores a index.css

Primero agregamos en el archivo index.css, el selector de class `div-footer`, `img-rs` y `li-footer`, defina sus propiedades



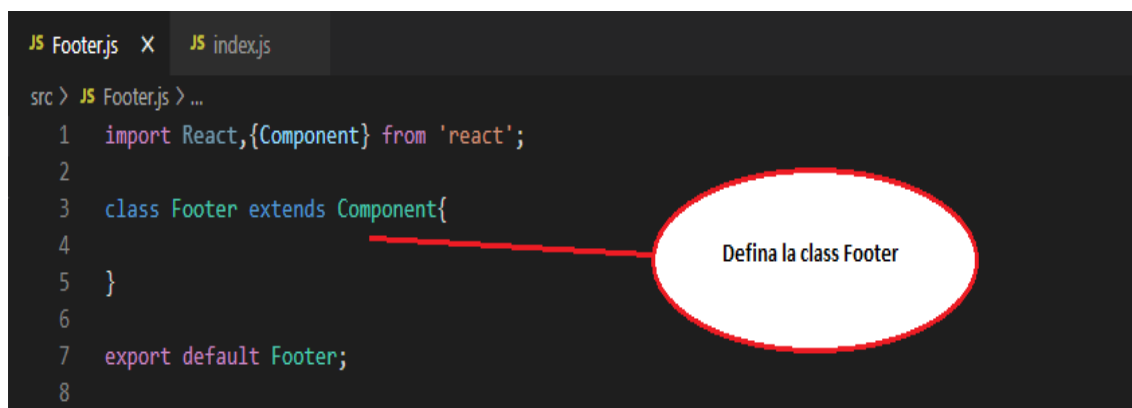
Figura 22: Index.css
Fuente: Diseño propio

Trabajando con Footer.js

En la carpeta src, agregamos un nuevo Archivo (new File) y lo llamaremos Footer.js.

En el archivo importar React y {Component}

A continuación defina la clase Footer de tipo Component



```
JS Footer.js X JS index.js
src > JS Footer.js > ...
1  import React,{Component} from 'react';
2
3  class Footer extends Component{
4
5  }
6
7  export default Footer;
8
```

Figura 23: Tradiciones.js
Fuente: Diseño propio

Defina los Arrays para listar las imágenes de las redes sociales (imgrs) y un segundo Array el cual almacena por cada ítem, un texto o descripción y route (dirección url)



```
JS Footer.js X JS index.js
src > JS Footer.js > Footer
1  import React,{Component} from 'react';
2
3  const imgrs=new Array("/img/face.jpg","/img/twitter.jpg","/img/instagram.jpg");
4
5  const listaOpciones = [
6    { text: "Index", route: "#" },
7    { text: "Tradiciones", route: "#" },
8    { text: "Costumbres", route: "#" },
9    { text: "Platos Típicos", route: "#" },
10   { text: "Restaurantes", route: "#" },
11   { text: "Museos y Parques", route: "#" },
12   { text: "Calendario", route: "#" },
13   { text: "Contactenos", route: "#" }
14 ];
15
16 class Footer extends Component{
17
18 }
19
20 export default Footer;
```

Figura 24: Tradiciones.js
Fuente: Diseño propio

A continuación, defina:

- El objeto `rs`, el cual almacena un Array de `img`, definidos mediante el método `createElement` de React
- El objeto `listaOpciones` el cual almacena un Array de `` y `<a>` mediante el método `createElement` de React

```

1  import React, {Component} from 'react';
2
3  const ingrs=new Array("/img/face.jpg","/img/twitter.jpg","/img/instagram.jpg");
4
5  const listaOpciones = [ ...
14 ];
15
16 const rs=ingrs.map((origen) =>React.createElement('img',{className:"img-rs", src:origen} ));
17
18 const lista=listaOpciones.map((it) => React.createElement('li',{className:'li-footer',
19   React.createElement('a',{href:it.route},it.text)}));
20
21 class Footer extends Component{
22
23 }
24
25 export default Footer;
26

```

definir el objeto `rs`, que almacena un Array de `img` leyendo cada elemento de `ingrs`

definir el objeto `lista` que almacena un Array de `` y `<a>` leyendo cada elemento de `listaOpciones`

Figura 25: Tradiciones.js

Fuente: Diseño propio

En la clase, al enviar los elementos definidos los agrupo a cada uno a través de un bloque `<div>`

```

20
21 class Footer extends Component{
22   render(){
23     return(
24       <div>
25         <div className="div-footer">
26           {lista}
27         </div>
28         <div className="div-footer">
29           <h3>Siguenos en...</h3>
30           {rs}
31         </div>
32       </div>
33     );
34   }
35 }
36
37
38 export default Footer;

```

Al renderizar la clase, envío el objeto `lista` y el objeto `rs`, cada uno agrupados en un `div`

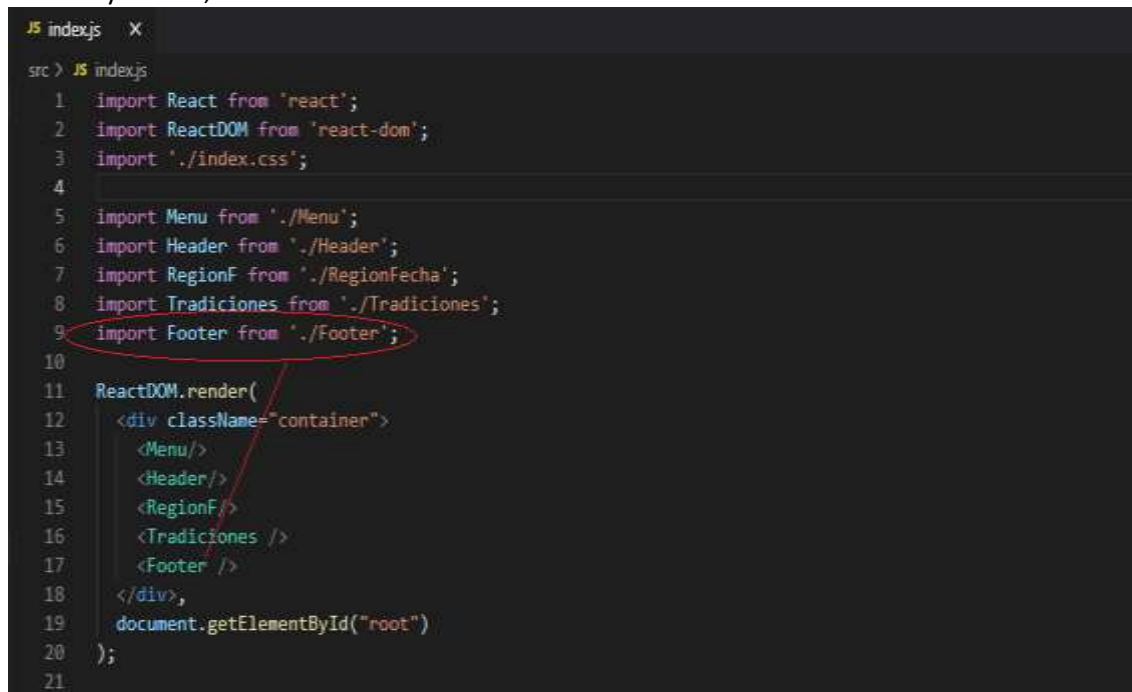
Figura 26: Tradiciones.js

Fuente: Diseño propio

Renderizar index.js

A continuación, abrir el archivo index.js y definir el siguiente código:

- Importar los archivos Menu.js, Header.js, RegionFecha.js, Tradiciones.js y Footer.js
- Al renderizar la página importamos el contenido de Menu, Header, RegionF, Tradiciones y Footer, tal como se muestra



```

1  import React from 'react';
2  import ReactDOM from 'react-dom';
3  import './index.css';
4
5  import Menu from './Menu';
6  import Header from './Header';
7  import RegionF from './RegionFecha';
8  import Tradiciones from './Tradiciones';
9  import Footer from './Footer';
10
11 ReactDOM.render(
12   <div className="container">
13     <Menu/>
14     <Header/>
15     <RegionF/>
16     <Tradiciones />
17     <Footer />
18   </div>,
19   document.getElementById("root")
20 );
21

```

Figura 27: index.js
Fuente: Diseño propio

Actualizar la página para visualizar el contenido de la class Tradicion.

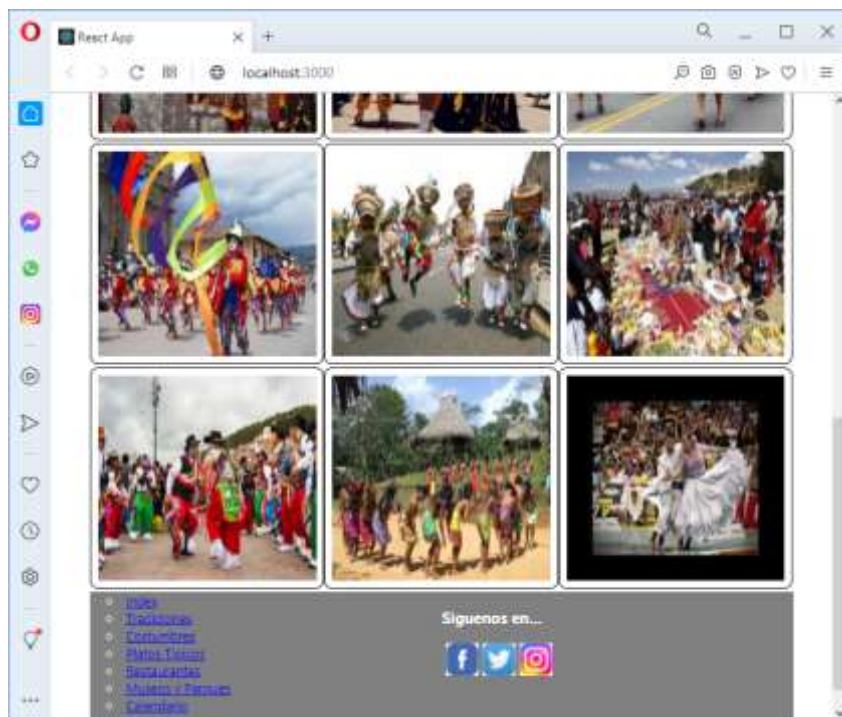


Figura 28: index.html
Fuente: Diseño propio