

## React - Ejercicio Cards - Parte 2

Dando continuidad con nuestro ejercicio de las `Cards` vamos a darle la funcionalidad para que cuando le demos clic, nos redirija a otra página o ruta de nuestro proyecto y sea allí donde podamos apreciar su contenido de manera independiente.

### Instalando `react-router-dom`

Para que poder implementar esa navegación que deseamos para nuestras Cards debemos instalar una biblioteca que nos permita dicha integración es por eso que debemos instalar `react-router-dom` ya que ésta es una herramienta fundamental para construir aplicaciones de página única (SPA) en React, proporcionando una forma elegante y eficiente de manejar la navegación dentro de la aplicación.

```
Terminal ...vite-proyecto/
```

[Copiar código](#)

```
npm i react-router-dom
```

### Re-configurando nuestra `src/Card.jsx`

En esta nueva configuración, importamos la biblioteca de `react-router-dom`, y re-configuramos los `props`, en este caso los pasamos como un parámetro y como objeto.

```
src/Card.jsx
```

[Copiar código](#)

```
import "../Card.css"
import { Link } from "react-router-dom"

function Card ({titulo = "titulo", descripcion = "descripcion"}){
  return(
    <div className="card">
      <Link to = {titulo}>
        <h2>{titulo}</h2>
      </Link>
      <p>{descripcion}</p>
    </div>
  )
}


export default Card;
```

## NOTA

Este código define un componente de tarjeta en React llamado Card, que muestra un título y una descripción. El componente utiliza estilos CSS importados y el componente Link de react-router-dom para crear un enlace que redirige a una ruta basada en el título proporcionado. Cuando se hace clic en el título, el usuario será dirigido a la ruta correspondiente, lo que permite la navegación dentro de la aplicación sin necesidad de recargar la página.

Muy bien! Ahora configuremos `src/vistas` las cuales se encargaran de mostrar el contenido de las `Cards` en sus respectivas rutas:

src/vistas/VistasLenguajes.jsx


 Copiar código

```
import React from "react"
import "../VistasLenguajes.css"

function VistasLenguajes({len}) {
  return(
    <div className="len-vista">
      <h1>{len.ntitulo}</h1>
      <p>{len.ndescripcion}</p>
      <p><img src={len.imagen} alt={len.imagen + "imagen"} /></p>
    </div>
  )
}

export default VistasLenguajes
```


Definimos el componente `VistasLenguajes`, sirve para mostrar `src/datos/lenguajes.js`. Utiliza propiedades para recibir datos del lenguaje, como su título, descripción e imagen, y los muestra dentro de un contenedor HTML con estilos definidos en un archivo CSS externo. El componente renderiza el título del lenguaje como un encabezado principal, seguido de su descripción y una imagen. La imagen se muestra utilizando la etiqueta `img`, donde la ruta de la imagen y un texto alternativo se obtienen de las propiedades del lenguaje. Una vez renderizado, este componente puede ser utilizado en otras partes de la aplicación para mostrar información detallada sobre diferentes lenguajes de programación.

`src/vistas/VistasLenguajes.css` Copiar código

```
.len-vista{
  max-width: 70%;
  margin: auto;
  text-align: center;
}

.len-vistas img{
  max-width: 50%;
}
```

Bien teniendo listas nuestras `Cards` y listas nuestras vistas para `Lenguajes.js` y sus estilos, solo nos haría falta re-configurar todo en nuestro archivo principal llamado `src/main.jsx`

`src/main.jsx` Copiar código

```
import React from 'react'
import ReactDOM from 'react-dom/client'
import App from './App.jsx'
import './index.css'
import lenguajes from './datos/lenguajes.js'
// Importamos router
import { BrowserRouter, RouterProvider } from 'react-router-dom'
// Importamos VistasLenguajes
import VistasLenguajes from './vistas/VistasLenguajes.jsx'

const router = [
  {
    path: "/",
    element: <App />,
  },
];

lenguajes.forEach((len) => {
  router.push({
```

```
path: len.titulo,  
element: <VistasLenguajes len = {len} />  
});  
});
```

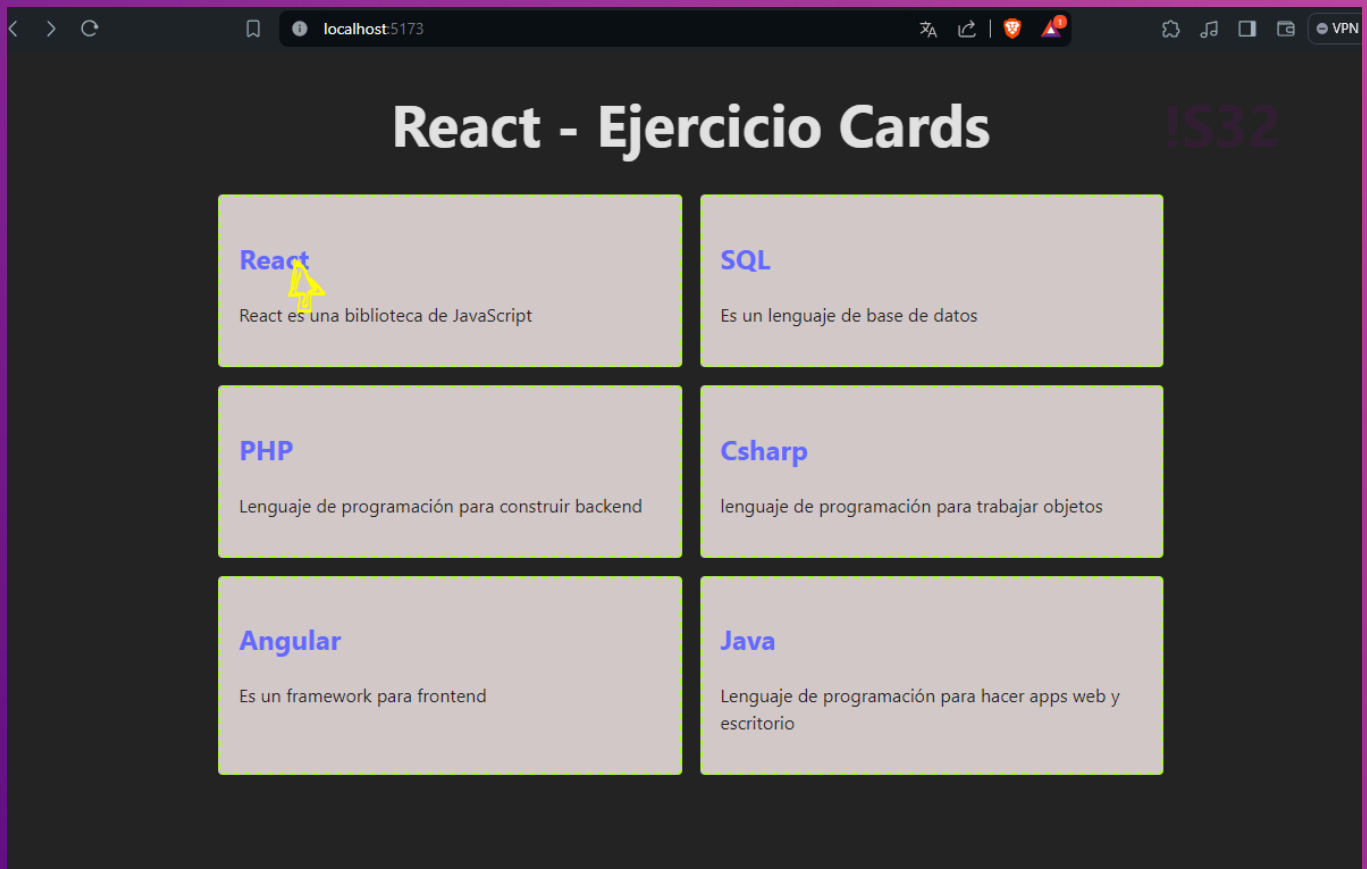
24.04.2024 S32-M2

```
const routes = createBrowserRouter(router)
```

```
ReactDOM.createRoot(document.getElementById('root')).render(  
  <React.StrictMode>  
    <RouterProvider router={routes} />  
  </React.StrictMode>,  
)
```

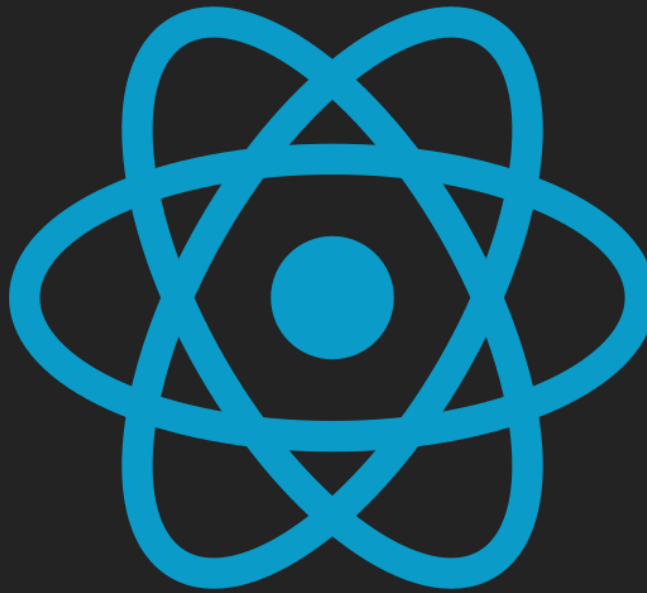
## Comprobando funcionalidad Cards

Si todo esta configurado de manera correcta, deberíamos poder hacer lo siguiente, cada contenido de nuestras Cards debería aparecer en su nueva ruta si damos clic en ellas asi:



# React

React es una biblioteca de JavaScript



## React - Ejercicio Cards

IS32

### React

React es una biblioteca de JavaScript

### SQL

Es un lenguaje de base de datos

### PHP

Lenguaje de programación para construir backend

### Csharp

lenguaje de programación para trabajar objetos

### Angular

Es un framework para frontend

### Java

Lenguaje de programación para hacer apps web y escritorio



localhost:5173/Csharp



24.04.2024 5:32-M2

# Csharp

lenguaje de programación para trabajar objetos




## React - Actividad

En esta nueva actividad vamos a crear unas **Cards** con datos sobre cursos referentes a los lenguajes de programación que ya hemos visto anteriormente. La idea es mostrar el tipo de lenguaje, nombre del curso, instructor, y el curso en formato de video.

Para esto, primero vamos a definir nuestra estructura de carpetas con los archivos nuevos que vamos a tener que crear y modificar:

Estructura de carpetas

 Copiar código

↓ src

↓ Componentes  
Card.css  
Card.jsx  
CardCurso.css// New  
CardCurso.jsx// New


↓ datos  
lenguajes.js  
cursos.jsx// New

↓ vistas  
VistasLenguajes.css  
VistasLenguajes.jsx  
VistasCursos.css// New  
VistasCursos.jsx// New  
App.css// Modified  
App.jsx// Modified  
index.css  
main.jsx// Modified

24.04.2024 S32-M2

## Definiendo nuestro modelo de datos

src/datos/cursos.js

 Copiar código

```
const cursos = [
  {
    nTipoLenguaje: "React Curso",
    nNombreCurso: "CURSO REACT 2024 - Aprende desde cero",
    nInstructor: "@midudev",
    nCurso: "https://www.youtube.com/embed/7iobxzd_2wY"
  },
  {
    nTipoLenguaje: "SQL Curso",
    nNombreCurso: "Curso de SQL y BASES DE DATOS Desde Cero para PRINCIPIANTES",
    nInstructor: "@mouredev",
    nCurso: "https://www.youtube.com/embed/OuJerKzV5T0"
  },
  {
    nTipoLenguaje: "Php Curso",
    nNombreCurso: "CURSO de php desde cero",
    nInstructor: "@develoteca",
    nCurso: "https://www.youtube.com/embed/nCB1gEkRZ1g"
  },
  {
    nTipoLenguaje: "Csharp Curso",
    nNombreCurso: "CURSO DE C# .NET GRATIS (Intensivo)",
    nInstructor: "@hdeleonnet",
    nCurso: "https://www.youtube.com/embed/j8sxDnr7nPY"
  },
  {
    nTipoLenguaje: "Angular Curso",
    nNombreCurso: "ANGULAR desde cero | Curso tutorial completo gratis por Sergie Code",
    nInstructor: "@SergieCode",
    nCurso: "https://www.youtube.com/embed/soInCF7nbDw"
  },
]
```


```
{
  nTipoLenguaje: "Java Curso",
  nNombreCurso: "Curso Completo de Java para Principiantes - 15 horas (2024)",
  nInstructor: "@atl.academy",
  nCurso: "https://www.youtube.com/embed/SpR_jWINBzg"
}

]

export default cursos;
```

## Configuramos nuestra Card para los cursos

src/Componentes/CardCursos.jsx


 Copiar código

```
import "../Card.css"
import "../CardCurso.css"
import { Link } from "react-router-dom"

function CardCursos ({TipoLenguaje = "TipoLenguaje", NombreCurso = "NombreCurso", Instructor = "Instructor"}) {
  return(
    <div className="card card-cursos">
      <Link to = {TipoLenguaje}>
        <h2>{TipoLenguaje}</h2>
      </Link>
      <p>{NombreCurso}</p>
      <p>{Instructor}</p>
    </div>
  )
}


export default CardCursos;
```

src/Componentes/CardCursos.css

 Copiar código

```
.card-cursos{
  background-color: #ceae1e;
}
```

src/Componentes/Card.css

 Copiar código



```
.card{
  border: dashed 2px greenyellow;
  background-color: #d6c8c8;
  padding: 1em;
  border-radius: 4px;
  width: 30%;
  color: #242424;
  transition: transform 0.3s ease;
}
```


```
.card:hover {
  filter: sepia(33%);
  transform: rotateY(10deg) rotateX(10deg);
}
```

## NOTA

Hemos añadido un estilo `hover` a la la clase `.card`. Debes tener en cuenta que no se lo aplicamos directamente a `card-curso` ya que como es un estilo que queremos definir de manera global para todas las `cards` lo hemos dejado así, para dejar `card-cursor.css` únicamente para estilos personalizados que sean del tipo cursos.

## Definimos la vista de cursos

src/vistas/VistasCursos.jsx

 Copiar código


```
import React from "react"
import "../VistasLenguajes.css"
import "../VistasCursos.css"

function VistasCursos({len}) {
  return(
    <div className="len-vista vista-cursos">
      <h1>{len.nTipoLenguaje}</h1>
      <h2>{len.nNombreCurso}</h2>
      <h3>{len.nInstructor}</h3>
      <iframe width="430" height="280" src={len.nCurso} title={len.nNombreCurso} frameborder="0" />
    </div>
  )
}

export default VistasCursos
```

## Configurando App.jsx

src/App.jsx

 Copiar código

```
import './App.css'
import Card from './Componentes/Card'
import CardCursos from './Componentes/CardCurso'
// Importamos nuestro Json con nuestros datos
import lenguajes from './datos/lenguajes.js'
import cursos from './datos/cursos.js'

function App() {

  const ListasLenguajes = lenguajes.map((l, index) =>{
    return <Card key = {index} titulo = {l.ntitulo} descripcion = {l.ndescripcion}/>
  })


  const ListasCursos = cursos.map((l, index) =>{
    return <CardCursos key = {index} TipoLenguaje = {l.nTipoLenguaje} NombreCurso = {l.nNombreCurso}/>
  })

  return (
    <div className="app">
      <h1>React - Ejercicio Cards</h1>
      <div className="container">
        {ListasLenguajes}
      </div>
      <h1>React - Actividad 🧑🏻</h1>
      <div className="container">
        {ListasCursos}
      </div>
    </div>
  )
}

export default App
```

## Terminamos configurando src/main.jsx

js

 Copiar código

```
import React from 'react'
import ReactDOM from 'react-dom/client'
import App from './App.jsx'
import './index.css'
import lenguajes from './datos/lenguajes.js'
```

```
import cursos from './datos/cursos.js'
// Importamos router
import { createBrowserRouter, RouterProvider } from 'react-router-dom'
// Importamos VistasLenguajes
import VistasLenguajes from './vistas/VistasLenguajes.jsx'
import VistasCursos from './vistas/VistasCursos.jsx'

const router = [
  {
    path: "/",
    element: <App />,
  },
];

lenguajes.forEach((len) => {
  router.push({
    path: len.ntitulo,
    element: <VistasLenguajes len = {len} />
  });
});

cursos.forEach((len) => {
  router.push({
    path: len.nTipoLenguaje,
    element: <VistasCursos len = {len} />
  });
});

const routes = createBrowserRouter(router)

ReactDOM.createRoot(document.getElementById('root')).render(
  <React.StrictMode>
    <RouterProvider router={routes} />
  </React.StrictMode>,
)
```

## Comprobación de nuestro ejercicio 🕵️

Con nuestro ejercicio completo de manera correcta tenemos la siguientes vistas en acción:



localhost:5173



24.04.2024 S32-M2



# React - Actividad



# !S32

## React Curso

CURSO REACT 2024 - Aprende desde cero

@midudev

## SQL Curso

Curso de SQL y BASES DE DATOS Desde Cero para PRINCIPIANTES

@mouredev

## Php Curso

CURSO de php desde cero

@develoteca

## Csharp Curso

CURSO DE C# .NET GRATIS (Intensivo)

@hdeleonnet

## Angular Curso

ANGULAR desde cero | Curso tutorial completo gratis por Sergie Code

@SergieCode

## Java Curso

Curso Completo de Java para Principiantes - 15 horas (2024)

@atl.academy



localhost:5173/Csharp%20Curso

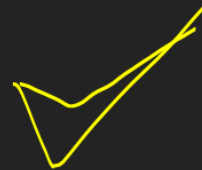


24.04.2024 5:32-M2

# Csharp Curso

CURSO DE C# .NET GRATIS (Intensivo)

@hdeleonnet





localhost:5173



24.04.2024



2-M2

# React - Actividad

# IS32

## React Curso

CURSO REACT 2024 - Aprende desde cero

@midudev

## SQL Curso

Curso de SQL y BASES DE DATOS Desde Cero para PRINCIPIANTES

@mouredev

## Php Curso

CURSO de php desde cero

@develoteca

## Csharp Curso

CURSO DE C# .NET GRATIS (Intensivo)

@hdeleonnnet

## Angular Curso

ANGULAR desde cero | Curso tutorial completo gratis por Sergie Code

@SergieCode

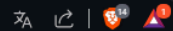
## Java Curso

Curso Completo de Java para Principiantes - 15 horas (2024)

@atl.academy



localhost:5173/React%20Curso



24.04.2024 S52-M2

# React Curso

CURSO REACT 2024 - Aprende desde cero

@midudev



Mira el repositorio en Github

Bryan Hernández | Telento Tech DWFSV2-42 | 2024

<https://4shc.online/tech/>