**2.2-2.3\*: Información del curso paso 7-8**

//COMPONENTE PRINCIPAL

const App = () => {

  const course = "Half Stack application development"

  const parts = [

    {

      name: "Fundamentals of React",

      exercises: 10

    },

    {

      name: "Using props to pass data",

      exercises: 7

    },

    {

      name: "State of a component",

      exercises: 14

    },

    {

      name: "Redux",

      exercises: 11

    }

  ]

  return <Course course={course} parts={parts}/>

}

// COMPONENTE QUE FORMATEA UN CURSO A LA VEZ

const Course = ({ course, parts }) => { //Course recibe dos props: course (nombre del curso) y parts (array de partes del curso).\*

  return ( //Dentro del componente Course, se renderizan dos componentes:\*\*

  /\*

  Header: se le pasa el prop course para mostrar el nombre del curso.

  Content: se le pasa el prop parts, que contiene las partes del curso, y este componente será responsable de mostrar los detalles de cada parte.

  \*/

    <>

      <Header course={course} />

      <Content parts={parts} />

    </>

  )

}

// COMPONENTE HEADER (ENCABEZADO)

 //Componente que muestra el título del curso

const Header = ({ course }) => <h1>{course}</h1>

/\*

Header recibe el prop course que contiene el nombre del curso.

Muestra el nombre del curso dentro de una etiqueta HTML <h1>. Es simplemente el encabezado.

\*/

// COMPONENTE QUE DEFINE UNA PARTE

//Este componente se encarga de renderizar una parte específica del curso (nombre de la parte y cuántos ejercicios tiene).

const Part = ({ part }) => (

  <p>

    {part.name} {part.exercises}

  </p>

  /\*

Part recibe un objeto part como prop, que contiene dos propiedades: name (nombre de la parte) y exercises (número de ejercicios).

Luego, dentro de un párrafo (<p>), muestra el nombre de la parte y el número de ejercicios.

\*/

)

// COMPONENTE CONTENT (CONTENIDO)

//Este componente es responsable de renderizar todas las partes del curso, usando el componente Part para cada una

const Content = ({ parts }) => {

// sumar todos los ejercicios usando el metodo reduce

const total = parts.reduce((sum, part) => sum+part.exercises,0)

return (

  <>

    {parts.map((part, index) => (

      <Part key={index} part={part} />

    ))}

    <p><strong>Total of {total} exercises </strong></p>

  </>

  )

  }

  /\*

Content recibe un prop parts, que es un array de objetos (cada objeto representa una parte del curso).

Usa el método map() para iterar sobre cada parte en el array parts. Por cada parte, se crea un componente Part pasándole el objeto part como prop.

key={index}: key es una propiedad especial en React que ayuda a identificar cada elemento cuando se renderizan listas de elementos. Usamos index como clave única para cada parte en el array.

\*/

/\*

const totalExercises = parts.reduce((sum, part) => sum + part.exercises, 0): Aquí estamos usando reduce para sumar todos los ejercicios:

    sum es el acumulador que comienza en 0 (el segundo argumento del reduce).

    part.exercises es el número de ejercicios de la parte actual.

    En cada iteración, se va sumando el número de ejercicios de la parte al valor acumulado.

<p><strong>Total of {totalExercises} exercises</strong></p>: Después de mostrar todas las partes, agregamos una línea que muestra el total de ejercicios calculado.

\*/

export default App

// COMPONENTE CONTENT (CONTENIDO)

const Content = ({ parts }) => {

// Cálculo de la suma total de ejercicios usando reduce

const totalExercises = parts.reduce((sum, part) => sum + part.exercises, 0)

return (

<>

{parts.map((part, index) => (

<Part key={index} part={part} />

))}

<p><strong>Total of {totalExercises} exercises</strong></p>

</>

)

}

**Explicación del cambio:**

* **const totalExercises = parts.reduce((sum, part) => sum + part.exercises, 0)**: Aquí estamos usando reduce para sumar todos los ejercicios:
  + sum es el acumulador que comienza en 0 (el segundo argumento del reduce).
  + part.exercises es el número de ejercicios de la parte actual.
  + En cada iteración, se va sumando el número de ejercicios de la parte al valor acumulado.
* **<p><strong>Total of {totalExercises} exercises</strong></p>**: Después de mostrar todas las partes, agregamos una línea que muestra el total de ejercicios calculado.

**Resultado:**

Ahora, al final del listado de partes, también verás el total de ejercicios sumados.