

Trabajo de App Móviles

```
`Nombre : Juan Carlos Jara`  
`Curso : 3 Nivel`  
`Materia: Aplicaciones Móviles`
```

Ejercicio 4

Importación y Declaración de Estado:

```
import React, { useState } from 'react';
```

```
const COLORS = [/* Lista de colores */];
```

```
const ResistorColorCalculator = () => {  
  const [selectedColors, setSelectedColors] = useState(['black', 'black']);
```

Aquí, se importa `React` y `useState` de `React` para gestionar el estado del componente funcional. Se define una constante `COLORS` que contiene una lista de colores que representan los valores de las bandas de una resistencia. `selectedColors` es un estado que almacena los colores seleccionados por el usuario.

Funciones de Manejo de Cambios de Color:

```
const handleColorChange = (index, color) => {  
  const newColors = [...selectedColors];  
  newColors[index] = color;  
  setSelectedColors(newColors);  
};
```

Esta función se encarga de actualizar el estado `selectedColors` cuando el usuario selecciona un color de una de las bandas de la resistencia. Recibe el índice de la banda y el color seleccionado, crea una nueva matriz de colores copiando la matriz existente, actualiza el color en la posición dada y luego actualiza el estado con la nueva matriz de colores.

Función de Decodificación del Valor:

```
const colorCode = (color) => COLORS.indexOf(color);  
  
const decodedValue = () => colorCode(selectedColors[0]) * 10 +  
  colorCode(selectedColors[1]); ``
```

`colorCode` es una función que devuelve el índice del color en la lista `COLORS`. `decodedValue`

calcula el valor de la resistencia decodificando los dos primeros colores seleccionados por el usuario, multiplicando el valor de la primera banda por 10 y sumándole el valor de la segunda banda.

Renderizado del Componente:

```
return (  
  <div>  
    {/* JSX para la interfaz del calculador de resistencias */}  
  </div>  
);``
```

Devuelve el JSX que representa la interfaz del calculador de resistencias. Incluye etiquetas `select` para que el usuario pueda seleccionar los colores de las dos bandas y muestra el valor decodificado de la resistencia.

Exportación del Componente:

```
`export default ResistorColorCalculator;`
```

Exporta el componente `ResistorColorCalculator` para que pueda ser utilizado en otros archivos de React.

Ejercicio 2

Este código implementa un componente de React llamado `ResistorColorCalculator`, que permite al usuario seleccionar los colores de las bandas de un resistor y calcula su valor. Aquí está una explicación detallada del código:

1. **useState Hook:** Utiliza el hook `useState` de React para manejar el estado de los colores seleccionados. Inicializa el estado con tres colores negros (`'black'`) por defecto.
2. **Array de Colores:** Define un array llamado `COLORS` que contiene los nombres de los colores posibles para las bandas del resistor.
3. **Funciones `decodedValue` y `decodedResistorValue`:** Estas funciones son responsables de calcular el valor del resistor basado en los colores seleccionados. `decodedValue` devuelve el valor numérico correspondiente a un color dado, mientras que `decodedResistorValue` calcula el valor del resistor en base a los colores seleccionados. La lógica de `decodedResistorValue` se basa en las reglas comunes para leer el valor del resistor a partir de los colores de sus bandas.
4. **Función `addZeros`:** Esta función auxiliar se utiliza para agregar ceros al valor del resistor según sea necesario, para asegurar que el valor tenga el formato correcto (por ejemplo, agregando ceros para representar valores en kiloohms, megaohms, etc.).

5. **Función `handleColorChange`** : Esta función maneja el cambio de color cuando el usuario selecciona un color diferente para una banda del resistor. Actualiza el estado `selectedColors` con el nuevo color seleccionado.
6. **Interfaz de Usuario**: Renderiza la interfaz de usuario, que consiste en un título, selectores de colores para cada banda del resistor y un área para mostrar el valor del resistor calculado. Utiliza el estado `selectedColors` para controlar los valores de los selectores de colores y llama a la función `decodedResistorValue` para calcular y mostrar el valor del resistor.
7. **Exportación del Componente**: Exporta el componente `ResistorColorCalculator` para que pueda ser importado y utilizado en otros archivos de la aplicación.

ENLACE DE YOUTUBE

<https://youtu.be/VzBVzSFbx8s>