**CLASES DE PROGRAMACION**

**PRESUPUESTO**

cd desktop/proyecto\_node/control-gastos

npm run dev

Todos los estilos están en “index.css”

Para llamarlos se pondrá className=“Nombre del estilo y especificaciones como columna o tamaño”

VALIDAR QUE SOLO VALGAN NUMEROS Y NO TEXTO(aunque puedas escribirlo)

Creamos un nuevo jsx llamado Mensaje

import { useState } from "react";

import Mensaje from "./Mensaje";

const NuevoPresupuesto = ({ presupuesto, setPresupuesto }) => {

const [mensaje, setMensaje] = useState("");

const handlePresupuesto = (e) => {

e.preventDefault();

if (!Number(presupuesto) || Number(presupuesto) < 0) {

setMensaje("No es un presupuesto válido");

return;

}

setMensaje(""); /// Resetear el mensaje

};

return (

<div className="contenedor-presupuesto contenedor sombra">

<form onSubmit={handlePresupuesto} className="formulario">

<div className="campo">

<label>Definir Presupuesto</label>

<input

className="nuevo-presupuesto"

type="text"

placeholder="Añade tu Presupuesto"

value={presupuesto}

onChange={(e) => setPresupuesto(e.target.value)}

/>

</div>

<input type="submit" value="Añadir" />

{mensaje && <Mensaje tipo="error">{mensaje}</Mensaje>}

</form>

</div>

);

};

export default NuevoPresupuesto;

import React from "react";

const Mensaje = ({ children, tipo }) => {

return <div className={`alerta ${tipo}`}>{children}</div>;

};

export default Mensaje;

PARA CONVERTIR UN VALOR NUMERICO EN “DINERO”

const ControlPresupuesto = ({ presupuesto }) => {

const formatearCantidad = (cantidad) => {

return cantidad.toLocaleString("es-SP", {

style: "currency",

currency: "EUR",

});

};

return (

<div className="contenedor-presupuesto contenedor sombra dos-columnas">

<div>

<p>Grafica aquí</p>

</div>

<div className="contenido-presupuesto">

<p>

<span>Presupuesto: </span> {formatearCantidad(presupuesto)}

</p>

<p>

<span>Disponible: </span> {formatearCantidad(0)}

</p>

<p>

<span>Gastado: </span> {formatearCantidad(0)}

</p>

</div>

</div>

);

};

AGREGAR BOTON PARA UN NUEVO GASTO

<https://www.udemy.com/course/react-de-principiante-a-experto-creando-mas-de-10-aplicaciones/learn/lecture/29688878#content>

Lo colocamos en “App.jsx”

import IconoNuevoGasto from "./img/nuevo-gasto.svg"; //IMPORTAR IMAGEN

{isValidPresupuesto && ( // ES UNA CONDICIÓN PARA QUE NO APAREZCA EN LA PRIMERA PAGINA SI NO YA EN LA PAGINA TRAS DECIR NUETROS PRESUPUESTO)

<div className="nuevo-gasto">

<img src={IconoNuevoGasto} alt="icono nuevo gasto" />

</div>

)}

APARICION DE VENTANA MODAL

( <https://www.udemy.com/course/react-de-principiante-a-experto-creando-mas-de-10-aplicaciones/learn/lecture/29688880?start=15#content> )

Cuando le demos al boton + (símbolo de nuevo gasto) aparecerá un módulo translucido donde habrá que poner el gasto, categoría y demás datos.

Dentro de “App.jsx” deberemos añadir lo siguiente:

import { useState } from "react";

import Header from "./components/Header";

import Modal from "./components/Modal"; //Para no alargar el codigo

import IconoNuevoGasto from "./img/nuevo-gasto.svg";

function App() {

const [presupuesto, setPresupuesto] = useState(0);

const [isValidPresupuesto, setIsValidPresupuesto] = useState(false);

const [modal, setModal] = useState(false);

const [animarModal, setAnimarModal] = useState(false); //Efecto de apertura

const handleNuevoGasto = () => {

setModal(true);

setTimeout(() => { // Agregar tiempo en aparición del módulo

setAnimarModal(true);

}, 500);

};

return (

<div>

<Header

presupuesto={presupuesto}

setPresupuesto={setPresupuesto}

isValidPresupuesto={isValidPresupuesto}

setIsValidPresupuesto={setIsValidPresupuesto}

/>

{isValidPresupuesto && (

<div className="nuevo-gasto">

<img

src={IconoNuevoGasto} //Imagen símbolo +

alt="icono nuevo gasto"

onClick={handleNuevoGasto} //Lo que debe ocurrir/aparecer al clicar en el símbolo +

/>

</div>

)}

{modal && <Modal

setModal={setModal}

animarModal={animarModal} //Llamar a esa función

setAnimarModal={setAnimarModal}

/>}

</div>

);

}

export default App;

**CREAREMOS “MODAL.JSX”** para no hacer tan largo el código en “App.jsx”

import { useState } from "react"; //Permite almacenar/seleccionar el valor que pongamos

import CerrarBtn from "../img/cerrar.svg"; //para importar una imange

const Modal = ({ setModal, animarModal, setAnimarModal }) => { //Fondo translucido y módulos que aparecerán

const [nombre, setNombre] = useState("");

const [cantidad, setCantidad] = useState(""); // Datos que vamos almacenar

const [categoria, setCategoria] = useState("");

const ocultarModal = () => {

setAnimarModal (false);

setTimeout(() => { //Tiempo hasta que desaparezca el fondo negro

setModal(false);

}, 500);

};

return (

<div className="modal">

<div className="cerrar-modal">

<img src={CerrarBtn} alt="cerrar modal" onClick={ocultarModal} /> //Imagen en este caso de una X para cerrar

</div>

// Al ser varias imágenes y estilos superpuestos, en index.css se puede ver “.modal .formulario.animar” y el de cerrar “.modal .formulario.cerrar”. Debemos crear una condición de que si está activo “animarModal” se ponga el fondo translucido y si no es asi se ponga el fondo “transparente”. (El nivel de opacidad que se muestra en el index.css)

<form className={`formulario ${animarModal ? "animar" : "cerrar"}`}>

<legend>Nuevo Gasto</legend>

<div className="campo">

<label htmlFor="nombre">Nombre Gasto</label>

<input

id="nombre"

type="text"

placeholder="Añadir el Nombre del Gasto"

value={nombre}

onChange={(e) => setNombre(e.target.value)} //Tipo de valor que almacenamos

/>

</div>

<div className="campo">

<label htmlFor="cantidad">Cantidad</label>

<input

id="cantidad"

type="number"

placeholder="Añadir la Cantidad del Gasto ej: 300"

value={cantidad}

onChange={(e) => setCantidad(Number(e.target.value))} //valor numerico

/>

</div>

<div className="campo">

<label htmlFor="categoria">Categoría</label>

<select

id="categoria"

value={categoria}

onChange={(e) => setCategoria(e.target.value)}

> // Diferentes tipos de categorías que podemos selecionar

<option value="">-- Seleccione --</option>

<option value="ahorro">Ahorro</option>

<option value="comida">Comida</option>

<option value="casa">Casa</option>

<option value="gastos">Gastos Varios</option>

<option value="ocio">Ocio</option>

<option value="salud">Salud</option>

<option value="suscripciones">Suscripciones</option>

</select>

</div>

<input type="submit" value="Añadir Gasto" /> //Botón para añadir el gasto

</form>

</div>

);

};

export default Modal;