//App.jsx

import React from 'react';

import './App.css';

// import { Greetings, Counter } from './Components/Counter';

import PasswordStrengthChecker from './Components/passwordCheck'

function App() {

  return (

    <div>

      {/\* <Counter />

      <Greetings name="World" /> \*/}

      <h1>hey pookie!! U can check ur password STRENGTH</h1>

      <PasswordStrengthChecker />

    </div>

  );

}

export default App;

// Counter.jsx

//components

import React, { useState } from 'react';

// Functional component for the counter

export function Counter() {

  const [count, setCount] = useState(0);

  return (

    <div>

      <p>You have clicked {count} times</p>

      <button onClick={() => setCount(count + 1)}>Click me</button>

    </div>

  );

}

// Functional component for greetings

export function Greetings(props) {

  return <h1>Hello, {props.name}</h1>;

}

// PasswordStrengthChecker.jsx

//components

import React, { useState } from 'react';

const PasswordStrengthChecker = () => {

  const [password,setPassword]=useState('')

  const [strength,setStrength]=useState('')

  const checkstr=(password) =>{

    let strength="weak"

    if (password.length >= 8) {

        if (/[A-Z]/.test(password) && /[a-z]/.test(password) && /\d/.test(password)) {

          strength = 'Strong';

        } else if (/[A-Z]/.test(password) || /[a-z]/.test(password)) {

          strength = 'Medium';

        }

    }

    setStrength(strength)

  }

  const take =(e)=>{

    const pass=e.target.value;

    setPassword(pass)

    checkstr(pass)

  }

  return (

    <div className="password-checker">

      <h2>Password Strength Checker</h2>

      <input

        type="password"

        value={password}

        onChange={take}

        placeholder="Enter your password"

      />

      <p className={`strength ${strength.toLowerCase()}`}>{strength}</p>

    </div>

  );

};

export default PasswordStrengthChecker;

// TIMER.JSX

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

const Timer = () => {

  const [time, setTime] = useState(0);

  const [isActive, setIsActive] = useState(false);

  useEffect(() => {

    let interval = null;

    if (isActive) {

      interval = setInterval(() => {

        setTime(prevTime => prevTime + 1);

      }, 1000);

    } else if (!isActive && time !== 0) {

      clearInterval(interval);

    }

    return () => clearInterval(interval);

  }, [isActive, time]);

  const startTimer = () => setIsActive(true);

  const stopTimer = () => setIsActive(false);

  const resetTimer = () => {

    setIsActive(false);

    setTime(0);

  };

  const formatTime = (seconds) => {

    const mins = Math.floor(seconds / 60);

    const secs = seconds % 60;

    return `${mins}:${secs < 10 ? `0${secs}` : secs}`;

  };

  return (

    <div>

      <h1>Timer</h1>

      <p>{formatTime(time)}</p>

      <button onClick={startTimer}>Start</button>

      <button onClick={stopTimer}>Stop</button>

      <button onClick={resetTimer}>Reset</button>

    </div>

  );

};

export default Timer;