Web Dev Assignment

Lab - 08

Name - Shikhar
Roll No - 22IT3046

Ques 1:

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

const CurrencyConverter = () => {
  const [amount, setAmount] = useState('');
  const [fromCurrency, setFromCurrency] = useState('USD');
  const [toCurrency, setToCurrency] = useState('EUR');
  const [convertedAmount, setConvertedAmount] = useState(null);

  const exchangeRate = 0.84; // Example exchange rate from USD to EUR

  const handleAmountChange = (e) => {
    setAmount(e.target.value);
  };

  const handleFromCurrencyChange = (e) => {
    setFromCurrency(e.target.value);
  };

  const handleToCurrencyChange = (e) => {
    setToCurrency(e.target.value);
  };

  const handleConvert = () => {
    const converted = parseFloat(amount) *
exchangeRate;
    setConvertedAmount(converted.toFixed(2));
  };

  return (
    <div>
      <h2>Currency Converter</h2>
      <div>
        <label htmlFor="amount">Amount:</label>
        <input type="number" id="amount"
value={amount} onChange={handleAmountChange} />
      </div>
      <div>
```

```
        <label htmlFor="fromCurrency">From:</label>
        <select id="fromCurrency" value={fromCurrency}
onChange={handleFromCurrencyChange}>
          <option value="USD">USD</option>
          <option value="EUR">EUR</option>
          {/* Add more options for other currencies */}
        </select>
      </div>
      <div>
        <label htmlFor="toCurrency">To:</label>
        <select id="toCurrency" value={toCurrency}
onChange={handleToCurrencyChange}>
          <option value="USD">USD</option>
          <option value="EUR">EUR</option>
          {/* Add more options for other currencies */}
        </select>
      </div>
      <button
onClick={handleConvert}>Convert</button>
      {convertedAmount && (
        <p>
          Converted Amount: {convertedAmount}
{toCurrency}
        </p>
      )}
```

```
    </div>
  );
};

export default CurrencyConverter;
```

Ques - 2 :

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

const Stopwatch = () => {
  const [time, setTime] = useState(0);
  const [isRunning, setIsRunning] = useState(false);

  useEffect(() => {
    let intervalId;

    if (isRunning) {
      intervalId = setInterval(() => {
        setTime((prevTime) => prevTime + 1);
      }, 1000);
    } else {
      clearInterval(intervalId);
    }

    return () => clearInterval(intervalId);
  }, [isRunning]);

  const startTimer = () => {
    setIsRunning(true);
  };

  const pauseTimer = () => {
    setIsRunning(false);
  };

  const resetTimer = () => {
    setTime(0);
    setIsRunning(false);
  };

  const formatTime = (time) => {
    const minutes = Math.floor(time / 60);
    const seconds = time % 60;
    return ${minutes < 10 ? '0' : ''}${minutes}:${seconds
< 10 ? '0' : ''}${seconds};
  };

  return (
    <div>
      <h2>Stopwatch</h2>
      <div>
        <p>{formatTime(time)}</p>
        <button onClick={startTimer} disabled={isRunning}>Start</button>
        <button onClick={pauseTimer} disabled={!isRunning}>Pause</button>
        <button onClick={resetTimer}>Reset</button>
      </div>
    </div>
  );
};

export default Stopwatch;
```