Mini Project Report - 09

Master of Computer Application – Generative AI
Semester – I

Sub: Front-End Frameworks and Technologies

Topic: Simple Interest Calculator using React

By
Name: MADAN KK
Reg no.: PROV/ASAC/MCA/7/25/078

Faculty Name: VEERA RAGHAV K

Faculty Signature: _______________

Department of Computer Application
Alliance University
Chandapura - Anekal Main Road, Anekal
Bengaluru - 562 106

September 2025

INDEX

INTRODUCTION

Introduction to HTML

HTML (HyperText Markup Language) is the backbone of all websites. It is used to structure and organize content on the web so that browsers can display text, images, videos, and links in a readable way. Instead of running calculations like a programming language, HTML works by marking up content with different tags to tell the browser *what each part of the page means*.

For example, you can use headings to make titles stand out, paragraphs for text, images for visuals, and links to connect one page to another. HTML works together with **CSS** (for styling and design) and **JavaScript** (for interactivity) to create complete and dynamic websites.

Some of the most important HTML tags are:

- <html> → The container that wraps the entire webpage.
- <head> → Holds meta information, links to CSS/JS, and the page title.
- <title> → Sets the title shown in the browser tab.
- <body> → Holds the visible content like headings, text, images, and links.

Introduction to CSS

CSS (Cascading Style Sheets) is the language that makes websites visually attractive and user-friendly. While HTML provides the structure of a webpage, CSS is responsible for its design—controlling colors, fonts, spacing, layouts, and even animations. With CSS, the same HTML content can be presented in completely different styles, giving web designers full creative control over how a site looks and feels.

In CSS:

- **Selector** → Targets the HTML element you want to style (e.g., h1).
- **Property** → Specifies the aspect to change (e.g., color, font-size).
- **Value** → Defines the exact style setting (e.g., blue, 20px).

Introduction to JS

JavaScript is a scripting language used to make web pages interactive and dynamic. It works with HTML and CSS to enhance user experience by responding to user actions like clicks or inputs. JS allows developers to update content, control multimedia, and animate elements without reloading the page. It can also handle form validation and interactive menus. Overall, JavaScript adds life and functionality to modern websites.

Introduction to React JS:

ReactJS, often referred to as React, is an open-source JavaScript library for building user interfaces (UIs). Developed by Facebook and maintained by Meta and a large community, it is widely used for creating dynamic and interactive single-page applications (SPAs) and complex web applications.

INPUT CODE :

React code:

```
//simple interest calculator using react
import React ,{useState} from 'react';
import "../src/counting.css"
```

```
function App(){
  const [price,setPrice]=useState();
  const [time,setTime]=useState();
  const [rate,setRate]=useState();
  const [interest,setInterest]=useState();
  const simple_interest =()=>{
    const interest = (price*time*rate)/100;
    setInterest(interest);


  };
  return(
    <div>
      <h1>Simple interest calculator using React</h1>
      <label>Price Amount:    </label>
      <input type = "number"
      placeHolder = "Principle Amount"
      value = {price}
      onChange = {(e) =>setPrice(e.target.value)}
      />
      <br></br>
      <label>Time rate: </label>
      <input class type = "number"
      placeHolder = "Rate of time"
      value = {time}
```

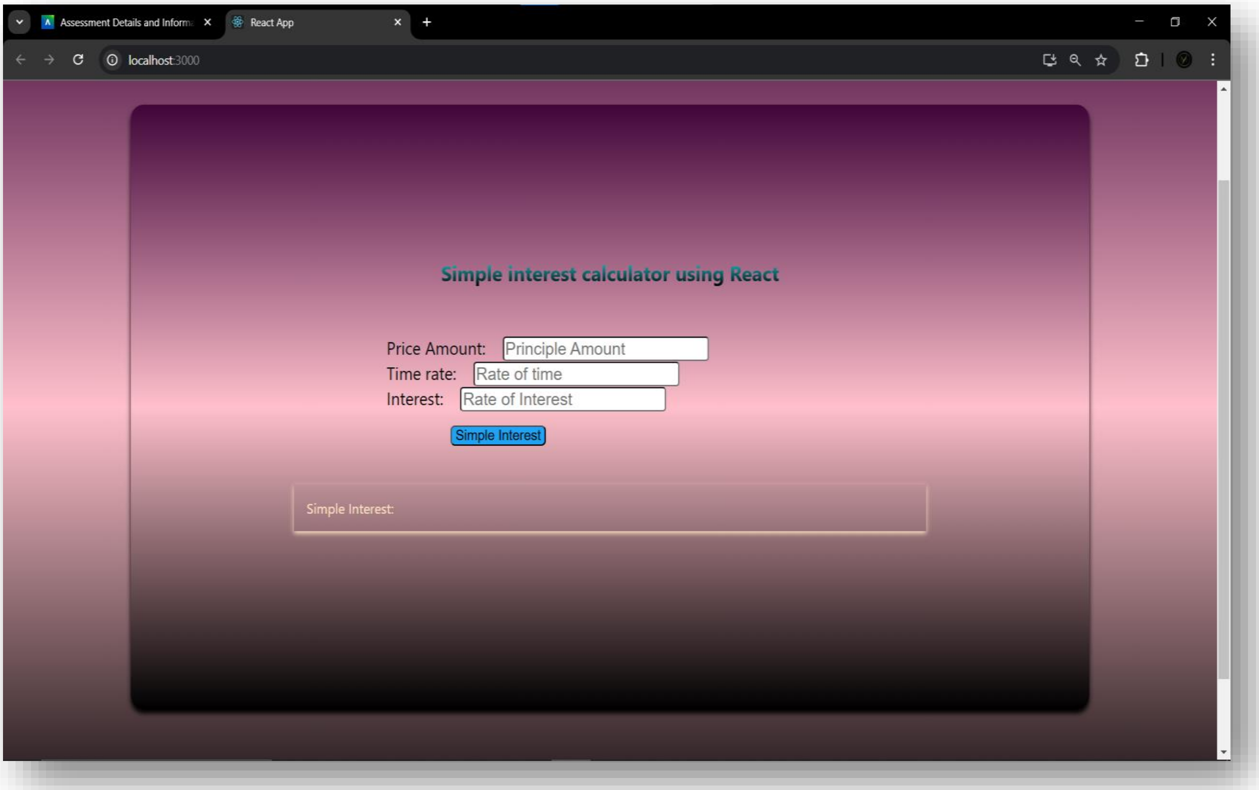
```
     onChange = {(e)=>setTime(e.target.value)}
     ></input>
     <br></br>
     <label>Interest: </label>
     <input type = "number"
     placeHolder = "Rate of Interest"
     value = {rate}
     onChange = {(e)=>setRate(e.target.value)}/>
     <br></br>
     <button onClick ={simple_interest}>Simple Interest</button>
     <p>Simple Interest: {interest}</p>
    </div>


  );
}
export default App;
```

CSS Code:

```
  h1{
    background: linear-gradient(rgb(28, 210, 216),black);
    color: transparent;
    -webkit-background-clip: text;

  }
  /* .heading{
    text-align: center;
    color: transparent;
    background: linear-gradient(black,yellow);
    -webkit-background-clip: text;
} */
h1{
    font-size: 30px;
    text-align: center;
    margin-bottom: 70px;
}
label{
    color:rgb(10, 13, 13);
    margin-left: 200px;
    font-size: 25px;

}
button{
    margin-top: 20px;
    text-align: center;
    margin-left: 300px;
    font-size: 20px;
    background-color: rgba(0, 162, 255, 0.866);
    border-radius: 7px;

}
p{
   color: bisque;
   box-shadow: 0px 3px 7px;
   margin-right: 100px;
   margin: 55px;
   text-align:  left;
   padding: 20px;
   font-size: 20px;
}
div{
    background: linear-gradient(rgb(65, 5, 58),pink,black);
    padding: 200px;
```

```
        border-radius: 20px;
        box-shadow: 0px 5px 7px;
    }
    input{
        margin-left: 20px;
        font-size: 25px;
        border-radius: 5px;
}
```

OUTPUT:



Simple interest calculator using React

Price Amount: Principle Amount
Time rate: Rate of time
Interest: Rate of Interest

Simple Interest:

CONCLUSION:

This mini project highlights how combining REACTcan transform plain content into a well-structured and visually appealing Calculator. Through this project, I gained hands-on experience in organizing content with REACT and enhancing its design with CSS. While the design is simple, it demonstrates the practical use of web technologies in creating professional and presentable documents, laying a solid foundation for building more advanced web projects in the future.