

PRACTICE REPORT

Subject: Web application

Lab 5: Build React App

Instructor: Trần Vĩnh Khiêm

Report date: 30/12/2023

1. GENERAL INFORMATION:

Class: MSIS207.012.CTTT.1

No	Name	Student ID	Email
1	Nguyễn Văn Trường Khoa	20521472	20521472@gm.uit.edu.vn
2			

DETAILED REPORT

Link Github contain code:

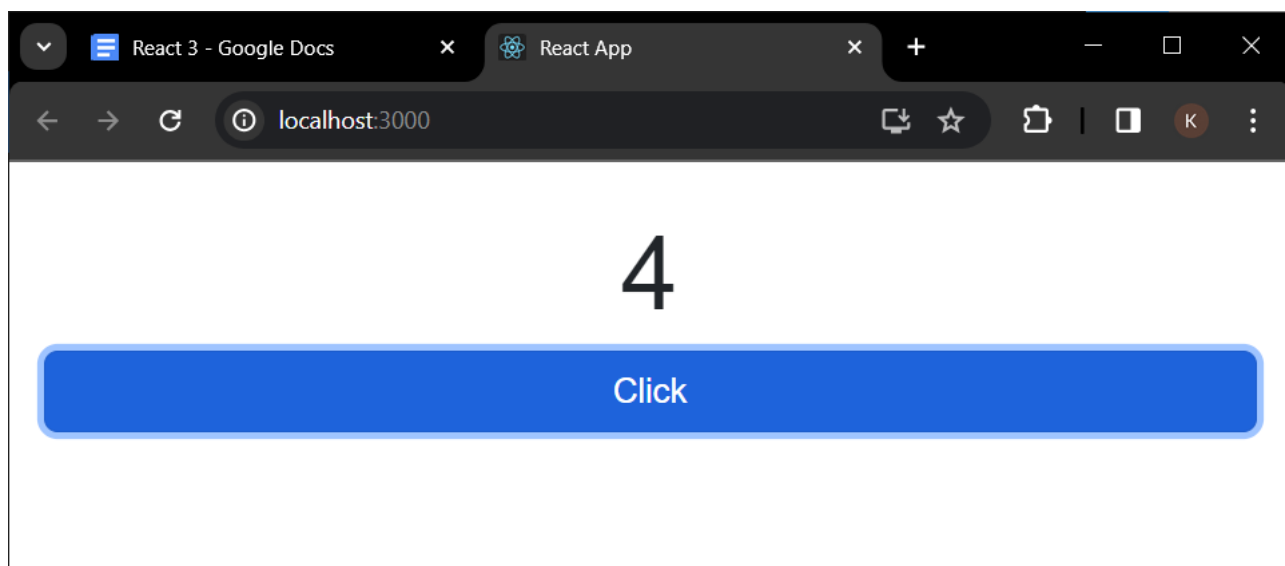
<https://github.com/KRocker3512/Web-app-practice-class>

Note : If you can't access link above, please contact to me via email :20521472@gm.uit.edu.vn .

1. Exercise 0

React 3:

Demo:



Code:

App.js

```
import Button from "react-bootstrap/Button";
import React, { useState } from "react";
import "./style.scss";

const React3 = () => {
  const [counter, setCounter] = useState(0);

  const increaseCounter = () => {
    setCounter(counter + 1);
  };

  return (
    <div class="react3">
      <div className="result">{counter}</div>
      <Button onClick={increaseCounter} size="lg">
```

```
    Click
  </Button>
</div>
);
};

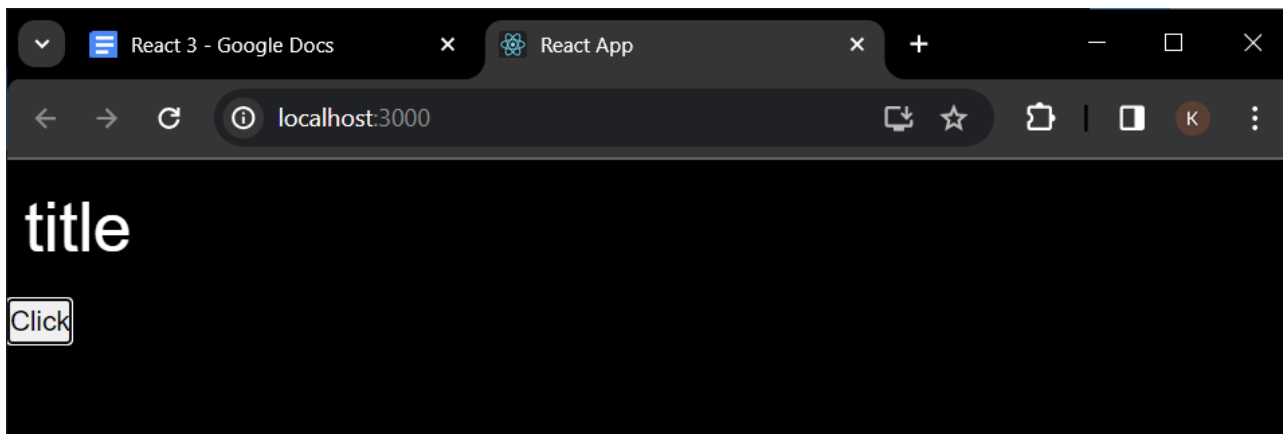
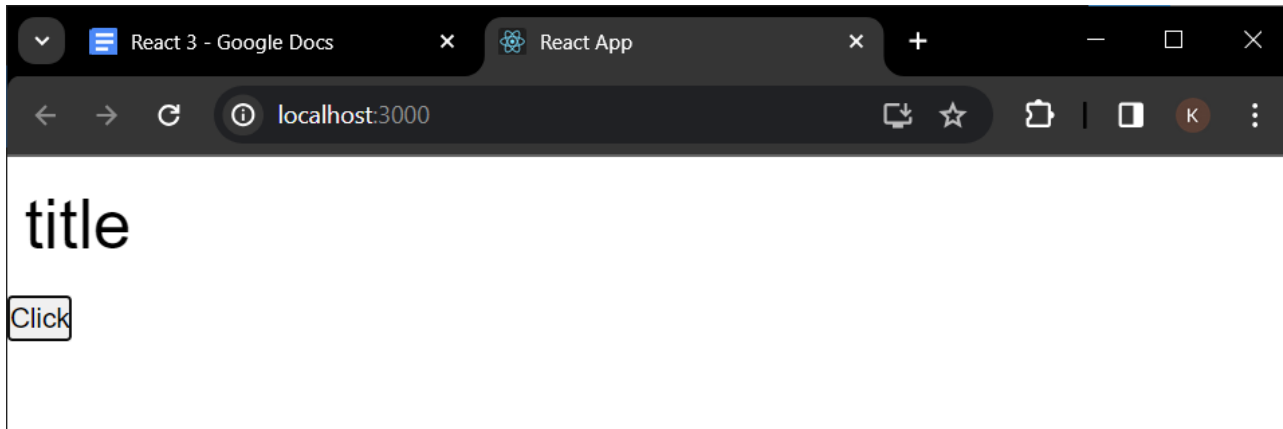
export default React3;
```

SCSS:

```
.react3 {
  display: flex;
  justify-content: center;
  padding: 20px;
  flex-direction: column;
  .result {
    font-size: 60px;
    text-align: center;
  }
}
```

React 6:

Demo:



Code:

Index.js

```
import MyComponent from "../components/myComponent";
import { ThemeProvider } from "../ThemeContext";

const React6 = () => {
  return (
    <ThemeProvider>
      <MyComponent />
    </ThemeProvider>
  );
};

export default React6;
```

ThemeContext.js

```
import React, { createContext, useEffect, useState } from "react";

const ThemeContext = createContext();

const ThemeProvider = (props) => {
  const { children } = props;

  const [theme, setTheme] = useState(false);

  const toggleTheme = () => {
    if (theme) document.documentElement.dataset.theme = "light";
    else document.documentElement.dataset.theme = "dark";
    setTheme((prev) => !prev);
  };

  useEffect(() => {
    document.documentElement.dataset.theme = "light";
  }, []);

  const value = { theme, toggleTheme };
  return (
    <ThemeContext.Provider value={value}>{children}</ThemeContext.Provider>
  );
};

export { ThemeContext, ThemeProvider };
```

Item component:

JS

```
import React, { useContext } from "react";
import "../style.scss";
```

```
import { ThemeContext } from "react6/ThemeContext";

const Item = (props) => {
  const context = useContext(ThemeContext);
  console.log(context);

  const { title, content } = props;
  return (
    <div className="component-item">
      <div className="title">{title}</div>
      <div className="content">{content}</div>
    </div>
  );
};

export default Item;
```

Scss

```
.component-item {
  display: flex;
  margin: 10px;
  .title,
  .content {
    padding: 10px;
    border: 5px solid;
  }
  .title {
    width: 200px;
    border-color: red;
  }
  .content {
    width: calc(100% - 200px);
    border-color: blue;
  }
}
```

myComponent component:

JS

```
import React, { useContext } from "react";
import { ThemeContext } from "../../ThemeContext";
import "./style.scss";

const MyComponent = () => {
  const context = useContext(ThemeContext);
  return (
    <div className="my-component">
      <div className="title">title</div>
      <button onClick={context.toggleTheme}>Click</button>
    </div>
  );
};
```

```
    </div>
  );
};

export default MyComponent;
```

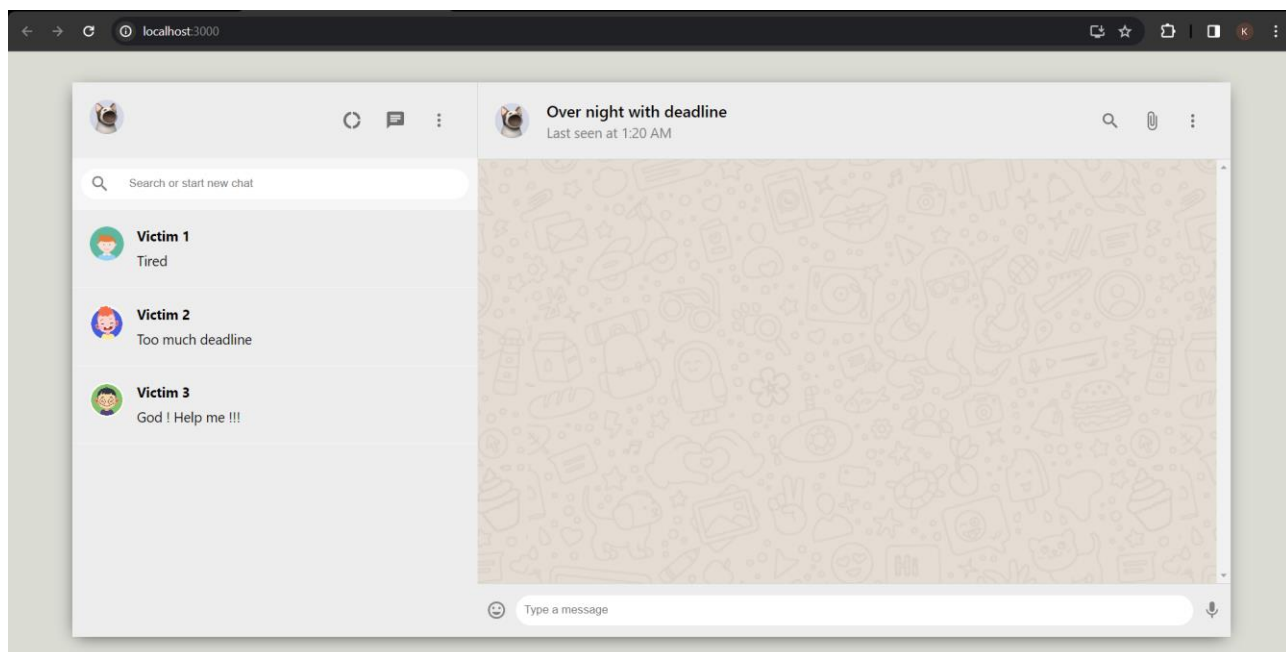
SCSS

```
.my-component {
  height: 100vh;
  background-color: var(--bg-main);
  .title {
    padding: 10px;
    font-size: 40px;
    color: var(--txt1);
  }
}
```

2.Exercise 1

3.Exercise 2

Demo :



Code : Because the code is too much and too long, I don't show it in the doc file but put it on github