React - Hooks

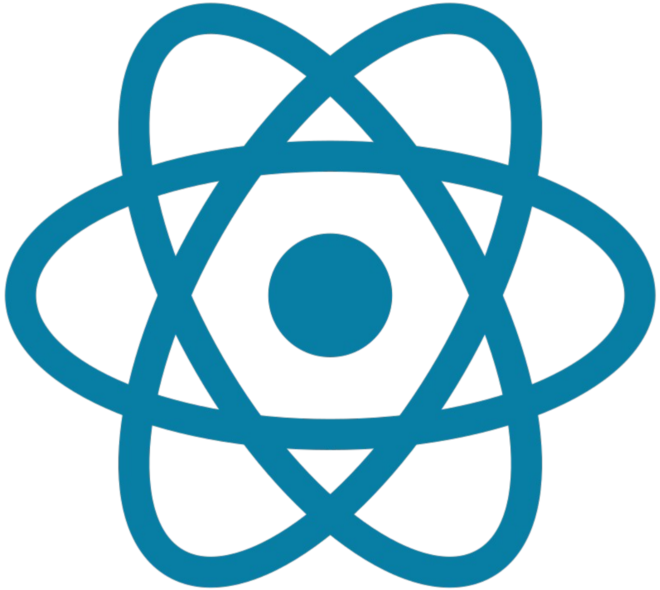
Justina Balsė

Julius Zabulėnas

TECHIN

Content

- Hooks
- useState
- useEffect

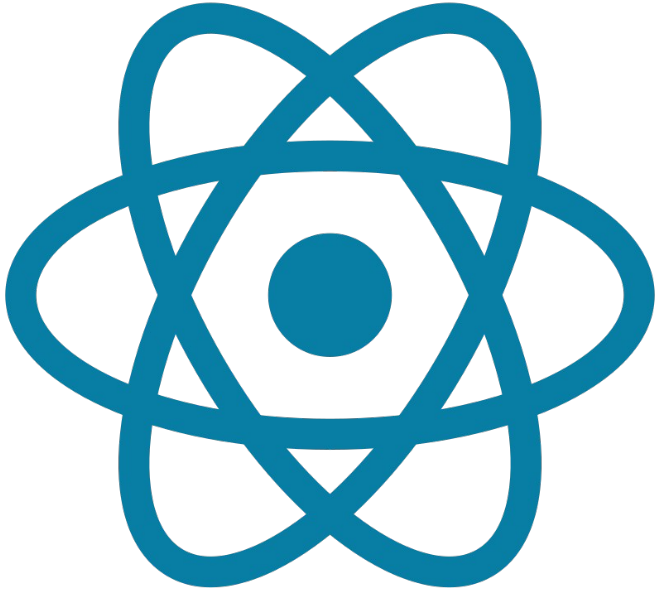


Kas tai - React Hooks?

- Hooks are a new addition in React 16.8
- They let you use state and other React features without writing a class

```
const [state, setState] = useState(initialState);
```

What Do Square Brackets Mean?



```
const [fruit, setFruit] = useState("banana");

// This JavaScript syntax is called "array destructuring".
// It means that we’re making two new variables fruit and setFruit,
// where fruit is set to the first value returned by useState,
// and setFruit is the second. It is equivalent to this code:

// Returns a pair
const fruitStateVariable = useState("banana");

// First item in a pair
const fruit = fruitStateVariable[0];

// First item in a pair
const setFruit = fruitStateVariable[1];
```

Destructuring assignment

```
// Oh no!
const person = {
  firstname: "John",
  lastname: "Snow",
  age: 59,
};

const firstname = person.firstname;
const lastname = person.lastname;
const age = person.age;
```

```
// Wow!
const person = {
  firstName: "John",
  lastName: "Snow",
  age: 59,
};

const { firstName, lastName, age } =
person;

const { firstName: fn } = person;
console.log(fn); // John
```

Spread syntax

```
// ES5
const arr1 = [23, 59, 61];
const arr2 = [71, 54, 96, 77];
const mergeArrays = arr1.concat(arr2);
```

```
// ES6
const arr1 = [23, 59, 61];
const arr2 = [71, 54, 96, 77];
const mergeArrays1 = [...arr1, ...arr2];
const mergeArrays2 = ["Hi", ...arr1, "Wow", ...arr2];
```

useState

- const [color, setColor] = useState("red");
- color - būsenos kintamasis
- setColor - funkcija
- "red" - pradinė būsenos kintamojo color reikšmė

```
import { useState } from "react";

export default function Example() {
  const [color, setColor] = useState("red");

  return <h1>My favorite color is {color}!</h1>;
}
```

useState + onClick()

- setColor() - keičiame būseną

```
import { useState } from "react";

export default function Example() {
  const [color, setColor] = useState("red");

  return (
    <div>
      <h1>My favorite color is {color}!</h1>
      <button
        onClick={() => setColor("green")}
        type="button"
        className="btn btn-light"
      >
        Change Color
      </button>
    </div>
  );
}
```

useState + onClick() + changeColor()

- changeColor() - funkcija, kuri kviečiama kai paspaudžiamas mygtukas *Change color*, norint pakeisti komponento būseną

```
import { useState } from "react";

export default function Example() {
  const [color, setColor] = useState("red");
  function changeColor() {
    setColor("green");
  }

  // Taip kurti priimtiniau
  //const changeColor = () => {
  //  setColor("green");
  //};

  return (
    <div>
      <h1>My favorite color is {color}!</h1>
      <button
        onClick={changeColor}
        type="button"
        className="btn btn-light"
      >
        Change Color
      </button>
    </div>
  );
}
```

useState + onClick() + changeColor()

- <h1 className={color}> keisti galima ir CSS klases

```
import { useState } from "react";

export default function Example() {
  const [color, setColor] = useState("text-danger");
  const changeColor = () => {
    setColor("text-success");
  }

  return (
    <div>
      <h1 className={color}>My favorite color is {color}!</h1>
      <button
        onClick={changeColor}
        type="button"
        className="btn btn-light"
      >
        Change Color
      </button>
    </div>
  );
}
```

useState + onClick() + changeColor()

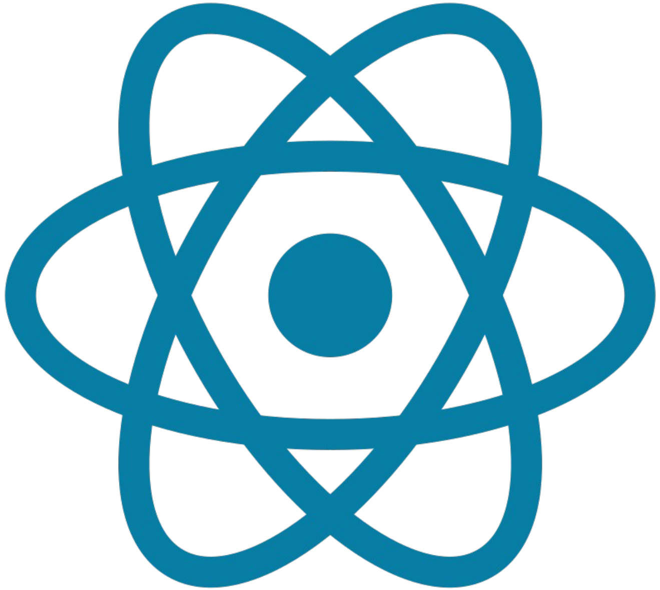
- className={lightTheme ? "text-danger" : "text-success"}
  Ternary sakinio naudojimas

```
import { useState } from "react";

export default function Example() {
  const [lightTheme, setLightTheme] = useState(false);
  function changeTheme() {
    setLightTheme(!lightTheme);
  }

  return (
    <div>
      <h1 className={lightTheme ? "text-danger" : "text-success"}>
        My Theme is {lightTheme ? "RED" : "GREEN"}!
      </h1>
      <button
        onClick={changeTheme}
        type="button"
        className="btn btn-light"
      >
        Change Theme
      </button>
    </div>
  );
}
```

TECHIN

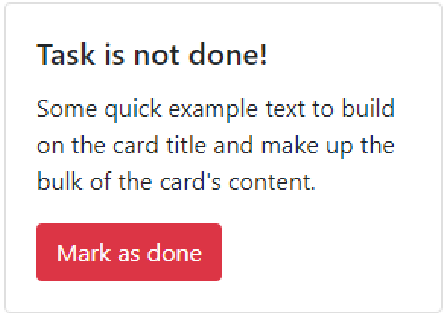
Using Multiple State Variables

```
export default function
ExampleWithManyStates() {
  // How to declare multiple state variables
  const [age, setAge] = useState(42);
  const [fruit, setFruit] =
useState("banana");
  const [todos, setTodos] = useState([{ text:
"Learn Hooks" }]);
}
```

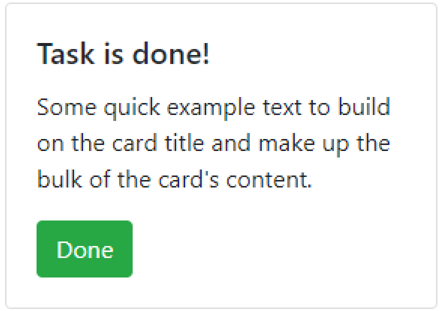


Praktika (1)

- Komponento mygtukas yra raudonas, kai užduotis yra neatlikta
- Komponento mygtukas yra žalias, kai užduotis yra atlikta
- Keičiasi ir komponento antraštė


Task is not done!

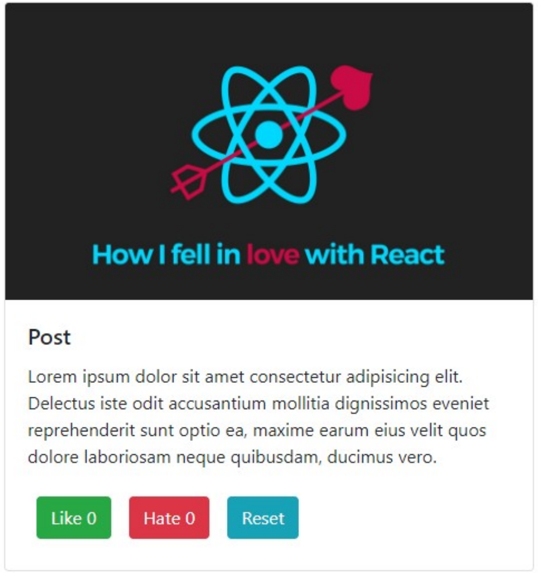
Some quick example text to build on the card title and make up the bulk of the card's content.


Task is done!

Some quick example text to build on the card title and make up the bulk of the card's content.

Praktika (2)

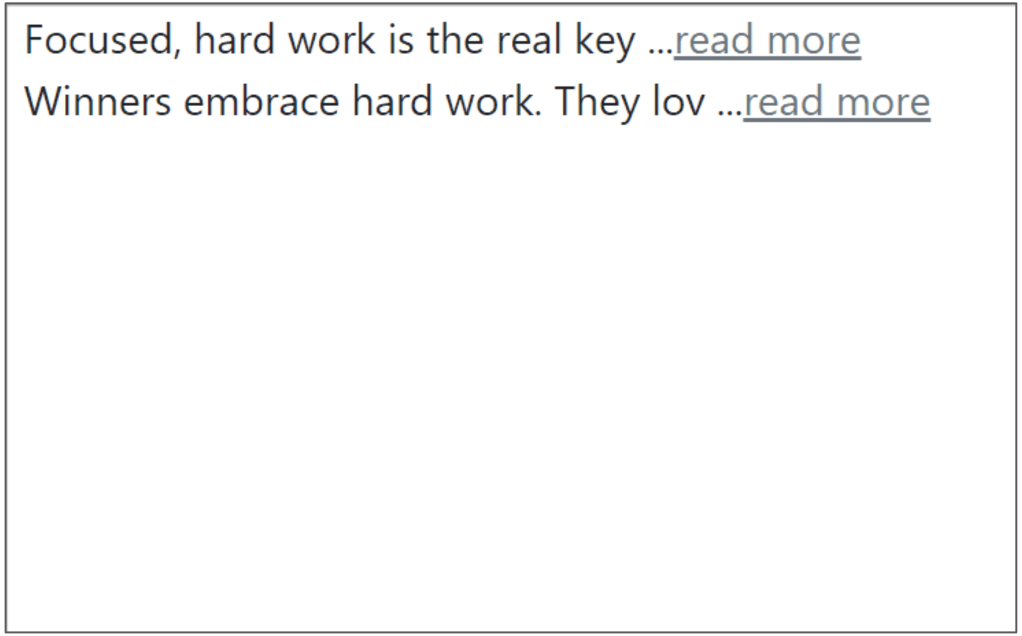
- Sukurti komponentą, kuris pakeistų savo būseną. Skaičiuotų kiek kartų buvo paspausti mygtukai *Like* ir *Hate*. Mygtukas *Reset* atstatytų komponento būseną į pradinę



Post

Lorem ipsum dolor sit amet consectetur adipisicing elit. Delectus iste odit accusantium mollitia dignissimos eveniet reprehenderit sunt optio ea, maxime earum eius velit quos dolore laboriosam neque quibusdam, ducimus vero.

Praktika (3)


Focused, hard work is the real key ...read more

Winners embrace hard work. They lov ...read more

Praktika (3) | <LessText />

```
<LessText
  text={"Focused, hard work is the real key
          to success. Keep your eyes on the goal,
          and just keep taking the next step
          towards completing it."}
  maxLength={35}
/>;
```

Pagalba:

substring()

trim()

TECHIN

Daugiau gyvybės

Bootstrap Icons:

```
npm install react-icons
```

Updating properties in multiple objects

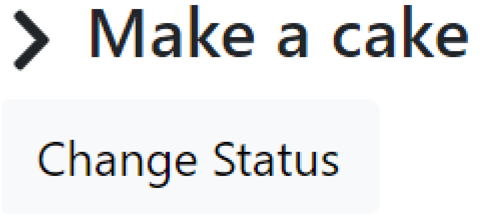
```
const arr1 = [
  { id: 1, name: "Alice", city: "London" },
  { id: 2, name: "Tom", city: "Paris" },
  { id: 3, name: "Charlie", city: "Berlin" },
];

const newArr = arr1.map((obj) => {
  if (obj.id === 1) {
    return { ...obj, name: "Bob" };
  }

  return obj;
});

console.log(newArr);
```

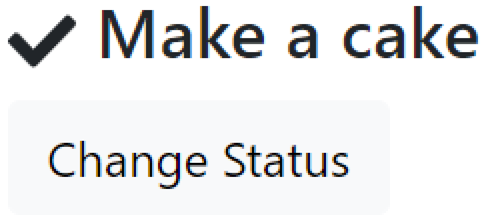
The functional or *updater* form of setCount()

```
import { useState } from "react";

export default function StepTracker() {
  const [steps, setSteps] = useState(0);

  function increment() {
    setSteps((prevState) => prevState + 1);
  }

  // Arba su arrow funkciją
  //const increment = () => {
  //  setSteps((prevState) => prevState + 1);
  //}

  return (
    <div>
      Today you've taken {steps} steps!
      <br />
      <button onClick={increment}>I took another
step</button>
    </div>
  );
}
```

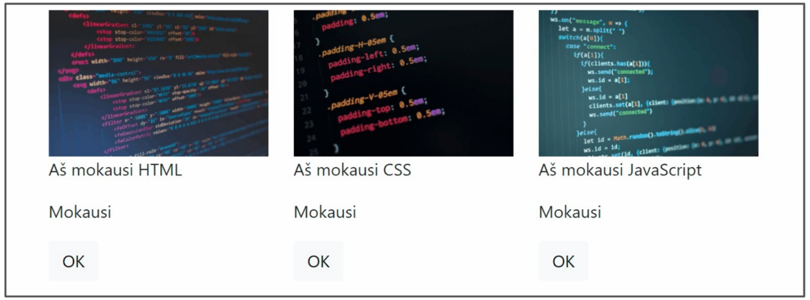
useState

```
import { useState } from "react";
import { FaCheck, FaChevronRight } from "react-icons/fa";

export default function Example() {
  const [task, setTask] = useState({
    taskTitle: "Make a cake",
    status: false,
  });

  function changeStatus() {
    setTask((previousState) => {
      return { ...previousState, status: true };
    });
  }

  return (
    <div>
      <h3>
        <span>{task.status ? <FaCheck /> : <FaChevronRight />}</span>
        {task.taskTitle}
      </h3>
      <button
        onClick={changeStatus}
        type="button"
        className="btn btn-light"
      >
        Change Status
      </button>
    </div>
  );
}
```





Praktika (4)


Aš mokausi HTML

Mokausi

OK

Aš mokausi CSS

Mokausi

OK

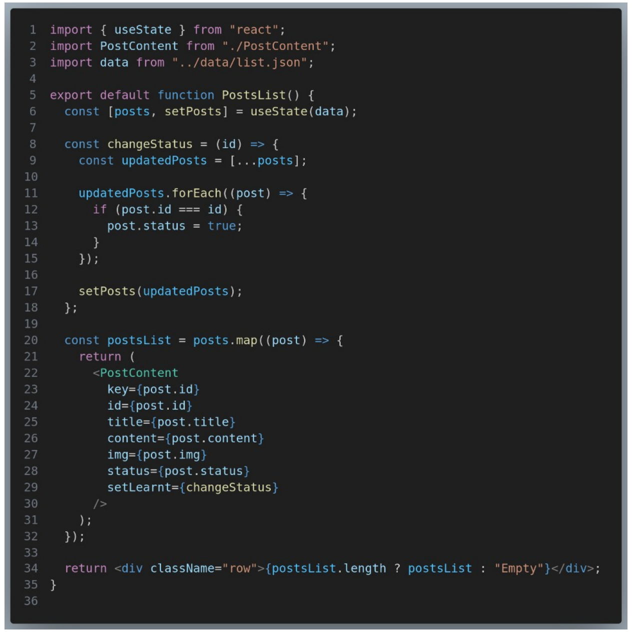
Aš mokausi JavaScript

Mokausi

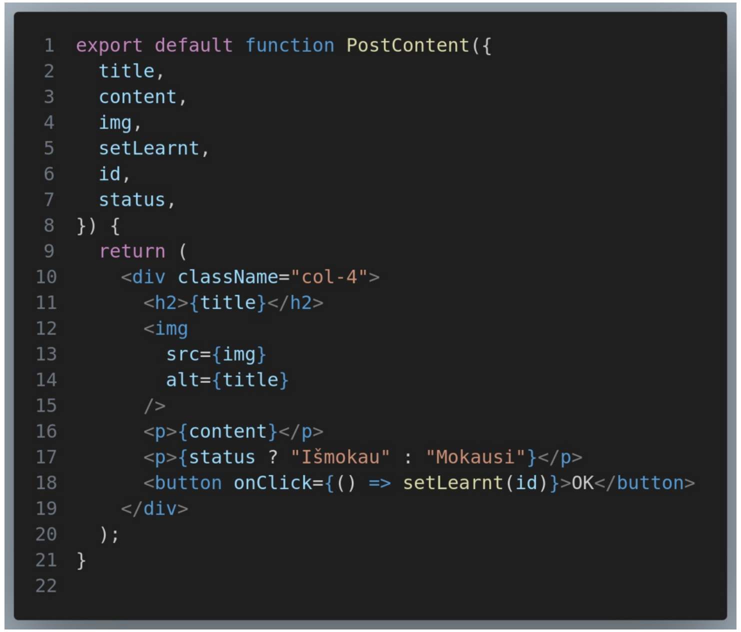
OK

- Paspaudus mygtuką OK iš Mokausi pasiverčia į Išmokau

Kaip perduoti funkciją?

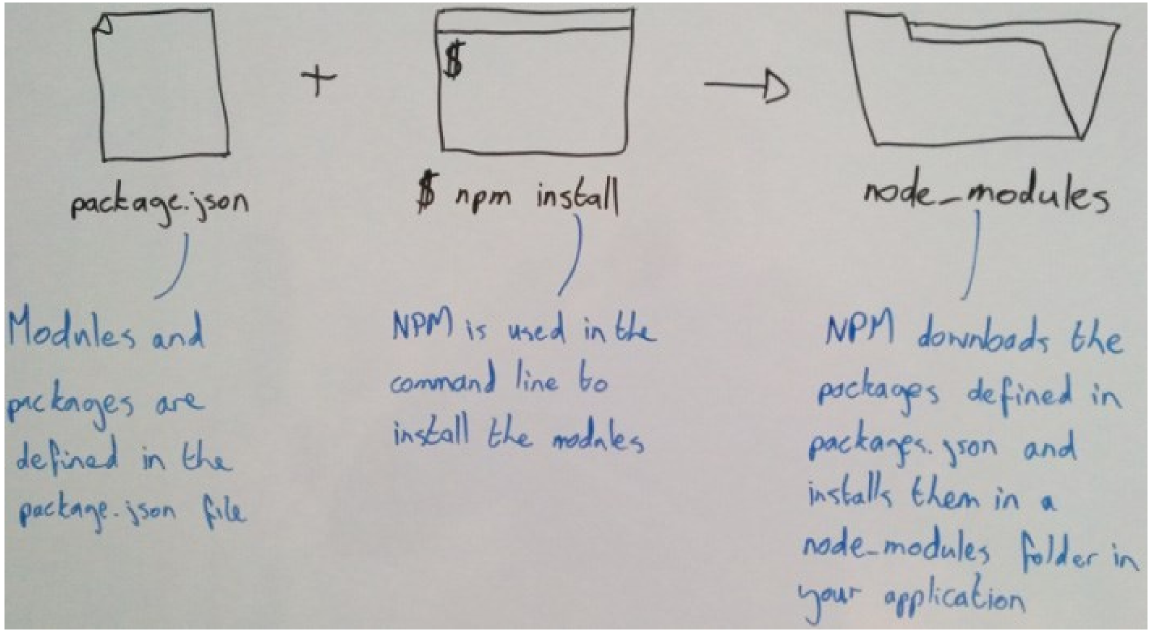


```jsx
import { useState } from "react";
import PostContent from "./PostContent";
import data from "../data/list.json";

export default function PostsList() {
  const [posts, setPosts] = useState(data);

  const changeStatus = (id) => {
    const updatedPosts = [...posts];

    updatedPosts.forEach((post) => {
      if (post.id === id) {
        post.status = true;
      }
    });

    setPosts(updatedPosts);
  };

  const postsList = posts.map((post) => {
    return (
      <PostContent
        key={post.id}
        id={post.id}
        title={post.title}
        content={post.content}
        img={post.img}
        status={post.status}
        setLearnt={changeStatus}
      />
    );
  });

  return <div className="row">{postsList.length ? postsList : "Empty"}</div>;
}

```



```jsx
export default function PostContent({
  title,
  content,
  img,
  setLearnt,
  id,
  status,
}) {
  return (
    <div className="col-4">
      <h2>{title}</h2>
      <img
        src={img}
        alt={title}
      />
      <p>{content}</p>
      <p>{status ? "Išmokau" : "Mokausi"}</p>
      <button onClick={() => setLearnt(id)}>OK</button>
    </div>
  );
}

```

package.json → npm install → node_modules


package.json
$
$ npm install
node_modules
Modules and packages are defined in the package.json file
NPM is used in the command line to install the modules
NPM downloads the packages defined in packages.json and installs them in a node_modules folder in your application

Effect Hook

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

export default function Example() {
  const [count, setCount] = useState(0);

  useEffect(() => {
   // Update the document title using the browser API
   document.title = `You clicked ${count} times`;
  });

  return (
   <div>
     <p>You clicked {count} times</p>
     <button onClick={() => setCount(count + 1)}>Click me</button>
   </div>
  );
}
```

- The *Effect Hook* lets you perform side effects in function components.
- **What does useEffect do?**
  - By using this Hook, you tell React that your component needs **to do something after render**
- React will remember the function you passed (we’ll refer to it as our “effect”), and call it later after performing the DOM updates. In this effect, we set the document title, but we could also perform data fetching or call some other imperative API

Dependencies argument | 1

```
import { useEffect } from "react";

export default function MyComponent() {
  useEffect(() => {
    // Runs after EVERY rendering
  });
}
```

- Not provided: the side-effect runs after *every* rendering.

TECHIN

Dependencies argument | 2

```
import { useEffect } from "react";

export default function MyComponent() {
  useEffect(() => {
    // Runs ONCE after initial rendering
  }, []);
}
```

- An empty array `[]`: the side-effect runs *once* after the initial rendering.

TECHIN

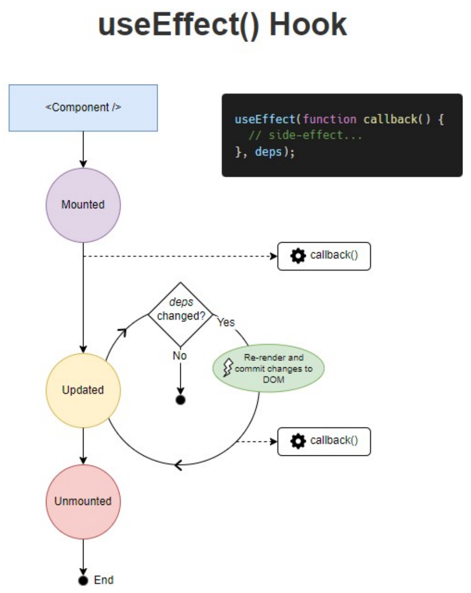
Dependencies argument | 3

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

export default function MyComponent({ prop }) {
  const [state, setState] = useState("");

  useEffect(() => {
    // Runs ONCE after initial rendering
    // and after every rendering ONLY IF `prop` or
`state` changes
  }, [prop, state]);
}
```

- Has props or state values `[prop1, prop2, ..., state1, state2]`: the side-effect runs *only when any dependendecy value changes*.

Effect Hook | Example 1

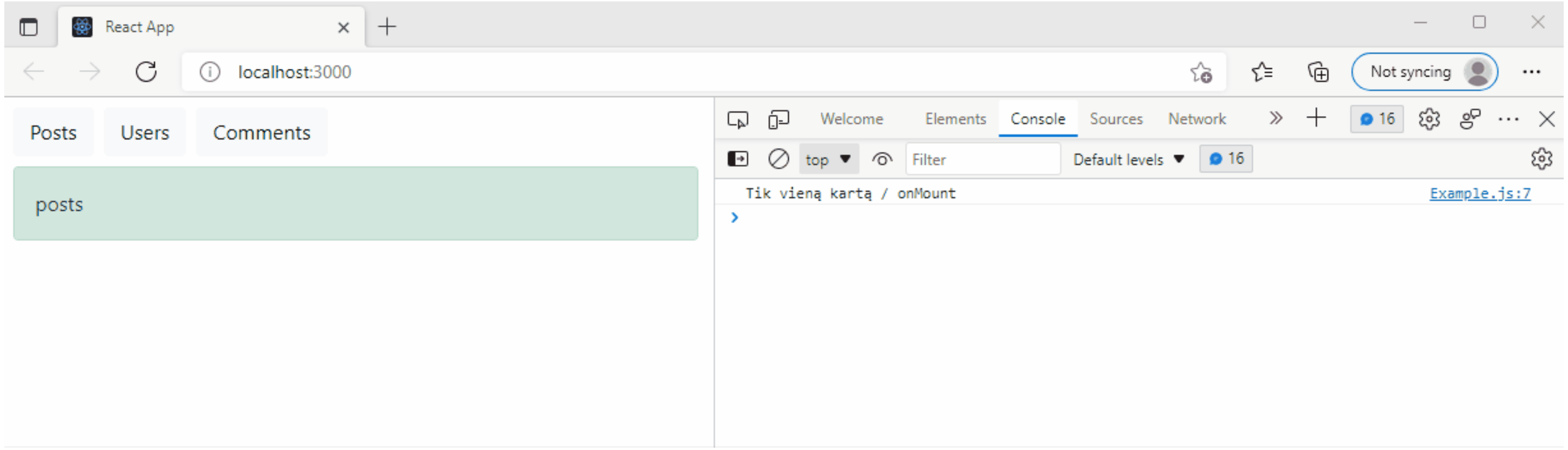
```
import { useState, useEffect } from "react";

export default function Example() {
  const [type, setType] = useState("posts");
  useEffect(() => {
    console.log("Tik vieną kartą / onMount");
  }, []);

  return (
    <div>
      <div className="my-2">
        <button
          onClick={() => setType("posts")}
          className="btn btn-light me-2"
        >
          Posts
        </button>
        <button
          onClick={() => setType("users")}
          className="btn btn-light me-2"
        >
          Users
        </button>
        <button
          onClick={() => setType("comments")}
          className="btn btn-light me-2"
        >
          Comments
        </button>
      </div>
      <div
        className="alert alert-success"
        role="alert"
      >
        {type}
      </div>
    </div>
  );
}
```



```
useEffect(function callback() {
  // side-effect...
}, deps);
```

Justina Balsė, Julius Zabulėnas

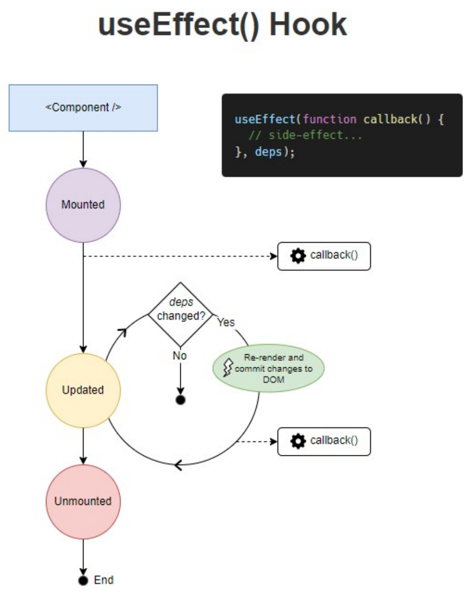
Effect Hook | Example 1



```
Tik vieną kartą / onMount                                        Example.js:7
```

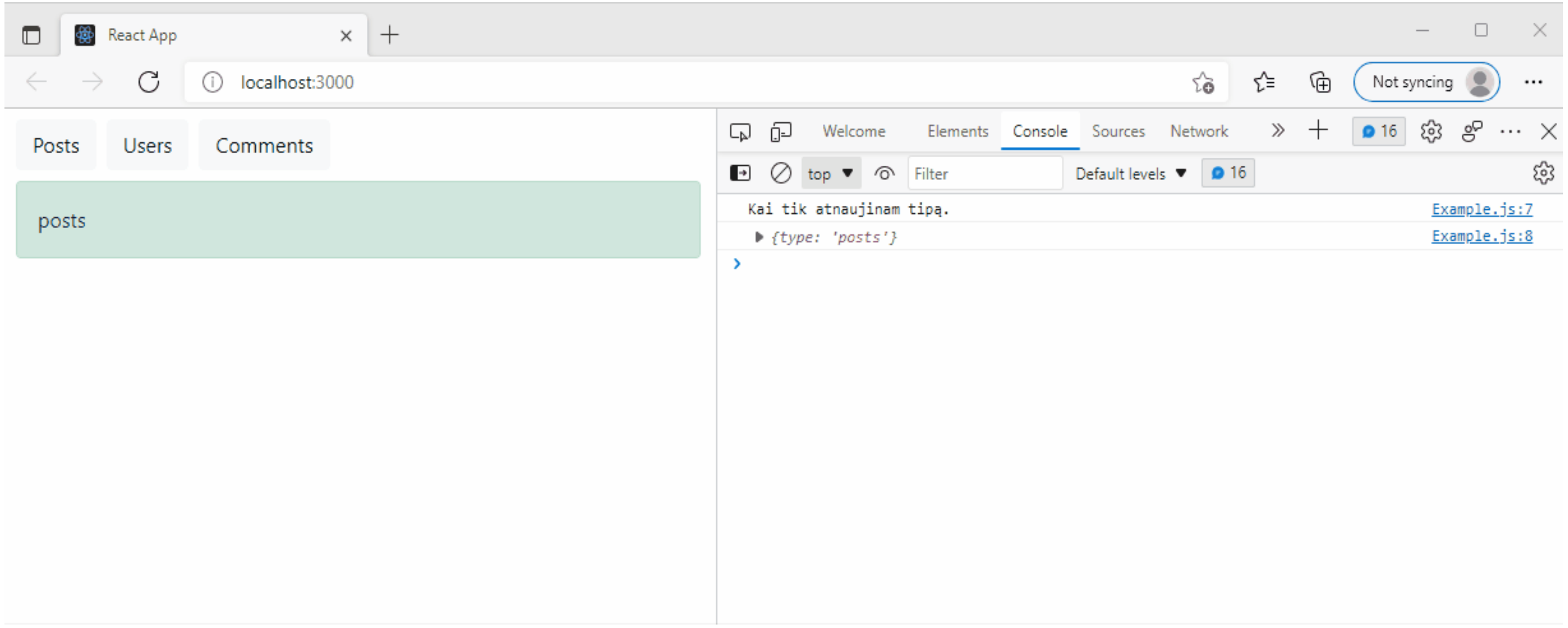
Effect Hook | Example 2

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

export default function Example() {
  const [type, setType] = useState("posts");
  useEffect(() => {
    console.log("Kai tik atnaujinam tipą.");
    console.log({ type });
  }, [type]);

  return (
    <div>
      <div className="my-2">
        <button
          onClick={() => setType("posts")}
          className="btn btn-light me-2"
        >
          Posts
        </button>
        <button
          onClick={() => setType("users")}
          className="btn btn-light me-2"
        >
          Users
        </button>
        <button
          onClick={() => setType("comments")}
          className="btn btn-light me-2"
        >
          Comments
        </button>
      </div>
      <div
        className="alert alert-success"
        role="alert"
      >
        {type}
      </div>
    </div>
  );
}
```



```
useEffect(function callback() {
  // side-effect...
}, deps);
```

Effect Hook | Example 2

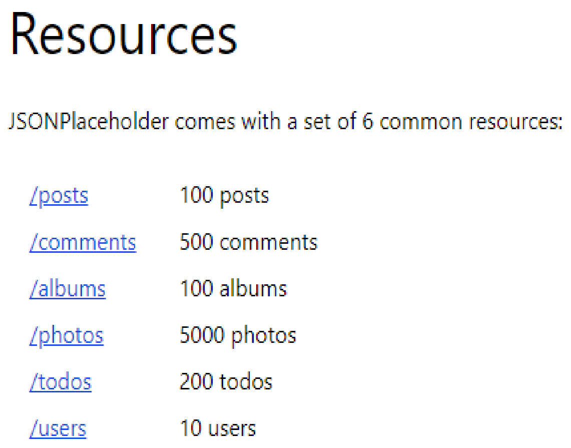


posts

```
Kai tik atnaujinam tipą.                Example.js:7
▶ {type: 'posts'}                       Example.js:8
>
```

Praktika (6) – tęsinys kitoje skaidrėje

1. Duomenys: https://jsonplaceholder.typicode.com/
2. Priklausomai nuo pasirinkimo, sugeneruoti duomenis


Resources

JSONPlaceholder comes with a set of 6 common resources:

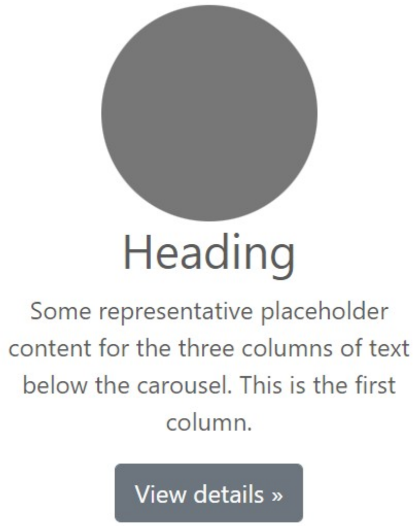
/posts	100 posts
/comments	500 comments
/albums	100 albums
/photos	5000 photos
/todos	200 todos
/users	10 users

Pavyzdys | getUsers() -> async, await

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

export default function UseEffectFetchData() {
  const url = "https://api.github.com/users";
  const [users, setUsers] = useState([]);

  const getUsers = async () => {
    const response = await fetch(url);
    const users = await response.json();
    setUsers(users);
    // console.log(users);
  };

  useEffect(() => {
    getUsers();
  }, []);

  return <>// Pavaizduoti ekrane users</>;
};
```

Praktika (7)

1. Doumenys: https://api.github.com/users
2. Github vartotojų duomenis atvaizduoti šiame Bootstrap šablone: Carousel Template


Heading

Some representative placeholder content for the three columns of text below the carousel. This is the first column.