

# Хуки

## Цель работы

Познакомить с понятием Хуков и работой с ними в React.

## Задания для выполнения

1. Реализуйте функцию App со следующим синтаксисом:

```
import './styles.css';
import React, { useState } from 'react';

export default function App() {
  // Объявляем новую переменную состояния "count"
  const [count, setCount] = useState(0);

  return (
    <div>
      <p>Вы нажали {count} раз</p>
      <button onClick={() => setCount(count + 1)}>
        Нажми на меня
      </button>
    </div>
  );
}
```

В этом примере, `useState` — это хук. Мы вызываем его, чтобы наделить наш функциональный компонент внутренним состоянием. React будет хранить это состояние между рендерами. Вызов `useState` возвращает массив с двумя элементами, который содержит: *текущее* значение состояния и функцию для его обновления. Эту функцию можно использовать где угодно, например, в обработчике событий. Она схожа с `this.setState` в классах, но не сливает новое и старое состояние вместе.

2. Хук состояния можно использовать в компоненте более одного раза.

```
// Объявляем несколько переменных состояния!
const [age, setAge] = useState(42);
const [fruit, setFruit] = useState('банан');
const [todos, setTodos] = useState([
  { text: 'Изучить хуки' }
]);
```

Реализуйте для этих состояний функционал (кнопку), а также измените состояния.

3. Загрузить созданную страницу на GitHub в репозиторий `tera`, используя формат в названии `Фамилия (латинскими буквами)_10`.

## Методические указания

Хуки — нововведение в React 16.8, которое позволяет использовать состояние и другие возможности React без написания классов.

Хуки не меняют ваши знания о концепциях в React. Вместо этого, хуки предоставляют более прямой доступ к API уже знакомых вам понятий: пропсов, состояния, контекста, рефов, и жизненного цикла. Мы также рассмотрим мощный способ компоновать эти понятия с помощью хуков.

**Хуки позволяют вам повторно использовать логику состояния, не затрагивая дерево компонентов. Также они позволяют разбить один компонент на маленькие функции по их назначению (например, подписке или загрузке данных).**

Хуки **НЕ** работают внутри классов, а используются *вместо* них.

## Полезные ссылки

<https://habr.com/ru/post/529950/>

<https://ru.reactjs.org/docs/hooks-overview.html>