

Методичка по практике разработки UI-компонентов на React

1. Цель методички

Данная методичка предназначена для студентов, изучающих основы React. Цель — закрепить ключевые концепции библиотеки через самостоятельную реализацию простых пользовательских интерфейсных компонентов.

В ходе выполнения заданий студенты научатся: - работать с JSX - использовать props и state - обрабатывать события - применять условный рендеринг - отображать списки данных - проектировать переиспользуемые компоненты

2. Общие требования к заданиям

Перед началом выполнения заданий студент должен: - уметь создавать React-проект (Create React App / Vite) - понимать основы JavaScript (ES6) - знать, что такое компонент и props

Общие правила:

- Использовать **функциональные компоненты**
 - Не использовать сторонние UI-библиотеки
 - Стилизация — минимальная или отсутствует
 - Каждый компонент — в отдельном файле
 - Имена компонентов — с заглавной буквы
-

3. Задания (по уровням сложности)

Уровень 1. JSX и props

Задание 1. Button

Описание: Реализуйте компонент кнопки.

Требования: - принимает `label` (текст кнопки) - принимает `onClick` - поддерживает `disabled`

Ожидаемые навыки: - работа с props - обработка событий

Задание 2. Title

Описание: Компонент заголовка страницы.

Требования: - принимает `text` - принимает `level` (1-3) - отображает соответствующий HTML-тег (`h1`, `h2`, `h3`)

Ожидаемые навыки: - динамический JSX

Задание 3. Card

Описание: Компонент-контейнер для контента.

Требования: - принимает `title` - отображает `children`

Ожидаемые навыки: - композиция компонентов - `props.children`

Уровень 2. Состояние и формы

Задание 4. Counter

Описание: Счётчик с кнопками увеличения и уменьшения.

Требования: - хранит значение в `state` - кнопки "+" и "-" - значение не может быть меньше 0 (опционально)

Ожидаемые навыки: - `useState` - обновление состояния

Задание 5. TextInput

Описание: Контролируемое поле ввода.

Требования: - принимает `value` - принимает `onChange` - отображает `placeholder`

Ожидаемые навыки: - controlled components

Задание 6. LoginForm

Описание: Форма авторизации.

Требования: - поля `email` и `password` - кнопка отправки - вывод данных в консоль при `submit`

Ожидаемые навыки: - формы - обработка `onSubmit`

Уровень 3. Условный рендеринг и списки

Задание 7. Toggle

Описание: Переключатель состояния.

Требования: - хранит boolean-состояние - отображает "ВКЛ" / "ВЫКЛ"

Ожидаемые навыки: - условный рендеринг

Задание 8. ItemList

Описание: Компонент отображения списка.

Требования: - принимает массив строк - использует `key`

Ожидаемые навыки: - рендеринг списков

Задание 9. TodoItem

Описание: Отдельный элемент todo-списка.

Требования: - текст задачи - статус выполнено / не выполнено - визуальное отличие выполненной задачи

Ожидаемые навыки: - props - conditional styles

Уровень 4. Композиция и логика

Задание 10. TodoList

Описание: Простейший todo-лист.

Требования: - добавление задач - удаление задач - отметка выполненных

Ожидаемые навыки: - работа с массивами в state - подъём состояния

Задание 11. Modal

Описание: Модальное окно.

Требования: - принимает `isOpen` - принимает `onClose` - отображает `children`

Ожидаемые навыки: - условный рендеринг - переиспользуемые компоненты

Задание 12. Alert

Описание: Компонент уведомлений.

Требования: - тип (`success` , `error` , `warning`) - текст сообщения

Ожидаемые навыки: - динамические классы

4. Критерии оценки

- компонент работает согласно описанию
 - код читаемый и аккуратный
 - корректно используется state и props
 - отсутствуют ошибки в консоли
-

5. Дополнительные задания (по желанию)

- Tabs
 - Accordion
 - Dropdown
 - Rating
 - Pagination
 - Search с фильтрацией списка
-

6. Рекомендации преподавателю

- поощряйте вопросы и обсуждение решений
 - проверяйте не только результат, но и подход
 - предлагайте студентам рефакторинг после проверки
-

Результат: После выполнения всех заданий студент должен уверенно понимать базовые принципы React и быть готов к изучению хуков, контекста и роутинга.