

Итоговая аттестация
по программе
«Программирование на JavaScript — полный курс»

Итоговая аттестация проходит в форме зачета – выполнения итогового задания на компьютере.

Таблица 1

Формы и методы контроля и оценки результатов освоения модулей

№ п/п	Наименование процедуры	Основные показатели оценки	Формы и методы контроля и оценки
1.	Текущий контроль. Введение в JavaScript. Переменные и типы данных	Владеет основными понятиями программирования	Создать прм проект,. установить и обновить необходимые пакеты. Разбить js-код на модули
2.	Текущий контроль. GitHub. Типы данных. Преобразование типов. Операторы сравнения	Владеет характеристиками React, TypeScript, инициализация приложения	Создать react приложение. Провести базовую настройку инструментов разработчика.
3.	Текущий контроль. Преобразование типов. Операторы сравнения. Массивы	Владеет основными понятиями и характеристиками webpack	Провести настройку webpack сборки в prod и dev режимы. Конфигурация babel
4.	Текущий контроль. Условные и логические операторы, циклы	Знает характеристики props, state	Создать react-компоненты. Передать необходимые props, добавить типизацию props
5.	Текущий контроль. Введение в функции	Владеет функцией Хуки и обработки событий	Добавить состояние для компонента с помощью хука useState. Рассказать про условия применения хуков. Добавить базовые обработчики события (onClick, onChange)
6.	Текущий контроль. Примитивы. Объекты	Владеет параметрами архитектуры приложения, key - проп, стилизации компонентов	Рассказать про container-component подход. Стилизовать компонент с помощью scss модулей и styled-components. На основе статического списка объяснить принцип работы key-props
7.	Текущий контроль. Методы массивов и объектов	Понимает особенности жизненного цикла компонентов	С помощью хука useEffect наделить компонент логикой работы с методами жизненного цикла. Затронуть этапы монтирования, обновления и размонтирования компонента.
8.	Текущий контроль. Продвинутая работа с функциями	Знает функции и возможности Роутинг в реакт приложении	Настроить роутинг в приложении. Рассказать базовые принципы навигации в SPA
9.	Текущий контроль. Прототипы и классы	Знает функции и возможности Redux toolkit, документацию API	Подключить и настроить redux-toolkit. Рассказать про основные составляющие redux. С помощью API положить данные в redux store и отрендерить их на странице

10.	Текущий контроль. DOM	Применяет методы продвинутой работы с хуками	С помощью хука useRef получить доступ до DOM и провести различные манипуляции с элементом (добавление/удаление атрибутов). Рассказать про мемоизацию компонентов. Применить хуки useMemo и useCallback для оптимизации рендера компонента
11.	Текущий контроль. Асинхронность в JavaScript. EventLoop	Владеет основами Node.js	Добавить API к текущему проекту
12.	Текущий контроль . Обработка ошибок. Fetch	Владеет основными принципами обработки ошибок	Реализовать перехват и обработку ошибки при передаче неправильной ссылки
13.	Текущий контроль Введение в продвинутый курс	Владеет основными понятиями программирования	Проверка усвоения материала
14.	Текущий контроль Знакомство с React и TypeScript. Инициализация приложения	Владеет характеристиками React, TypeScript, и инициализация приложения	Проверка настройки рабочей среды
15.	Текущий контроль Знакомство с webpack	Владеет основными понятиями и характеристиками webpack	Провести базовую настройку babel
16.	Текущий контроль Компоненты, props, state	Знает характеристики props, state	Выбрать макет с любого ресурса. Создать страницу, с разбиением на логические компоненты (header, content (mainInfo + description + slider + gallery), footer).
17.	Текущий контроль Хуки и обработка событий	Владеет функцией Хуки и обработки событий	Проверка реализации задачи «Карусель»
18.	Текущий контроль Архитектура приложения, key - prop, стилизация компонентов	Владеет параметрами архитектуры приложения, key - prop, стилизации компонентов	Добавить в приложение несколько страниц, следуя заложенному подходу (контейнер-компоненты).
19.	Текущий контроль Жизненный цикл компонентов	Понимает особенности жизненного цикла компонентов	1. Создать страницу + форму регистрации. Реализовать валидацию по аналогии с классной работой. Добавить проверку на равенство введенных пользователем паролей. 2. Опираясь на компонент Input реализовать компонент PasswordInput, добавить в компонент логику просмотра пароля. 3. Модифицировать компоненты Input и PasswordInput, что бы они меняли свои стили в зависимости

			от корректности ввода, валидация поля должна происходить при снятии фокуса с элемента
20.	Текущий контроль Роутинг в реакт приложении	Знает функции и возможности Роутинг в реакт приложении	Настроить роутинг для страниц и навигацию
21	Текущий контроль Redux toolkit и работа с API	Знает функции и возможности Redux toolkit, документацию API	Настроить подключение redux и реализовать кейс сохранения информации с регистрации пользователя (email, password) в отдельные поля внутри стора. И после нажатия на кнопку регистрации, если все данные валидны, редиректить пользователя на страницу авторизации (заполнять поля информацией из стора email и password)
22.	Текущий контроль Продвинутая работа с хуками	Применяет методы продвинутой работы с хуками	Реализовать приложение со списком дел используя hook
23.	Текущий контроль Основы Node.js	Владеет основами Node.js	Реализовать сбор статистики и создание отчетов на Node.js
	Итоговая аттестация	Реализация способностей в сфере разработки, отладки, проверки работоспособности, модификации программного обеспечения	Зачет/Проверка выполненного итогового задания на компьютере

Критерии оценки итоговой аттестации (текущего контроля)

Результат итоговой оценки практического задания	Критерии оценки
Не зачтено	Практическое задание выполнено с большим количеством недочетов
Зачтено	Практические задания выполнены без недочетов