Паспорт курса **«Код Будущего: уверенный старт»**

Модуль «JavaScript»

| Продолжительность | 16 занятий по 90 минут |
| --- | --- |
| Возрастная категория | 12-15 лет |
| Рекомендуемая наполняемость групп | до 15 человек |
| Требования к базовому оснащению учебного класса | Рабочие станции по числу учеников   * устройства на Windows * устройства на MacOS * устройства на ChromeOS * устройство на Linux   Рабочая станция преподавателя   * устройства на Windows * устройства на MacOS * устройства на ChromeOS * устройство на Lynux   Любой современный браузер  Мультимедийный проектор  Выход в интернет (wi-fi)  Опционально: интерактивная доска  Канцелярские принадлежности |
| Программное обеспечение | Пакет офисных программ  Visual Studio Code - редактор кода  Операционная система Windows выше 10  Ссылки на все необходимые ресурсы указаны в поурочном планировании |
| Рекомендуемые требования к квалификации преподавателя | Педагогическое образование, либо опыт преподавательской работы с детьми (детский сад, школа, доп. образование).  Уверенное владение компьютером.  Понимание общих принципов программирования и логики создания компьютерных программ. Желательно владение навыками программирования на любом языке (предпочтительно: Python, C++, Java, JS) |
| Место в курсовой линейке | Четвертый модуль курса «Код Будущего: уверенный старт»  Возможно изучение модуля в любом порядке текущего курса.  Рекомендовано к изучению все 4 модуля. |

**Общие методические рекомендации к курсу**

| **Направление** | **Описание** |
| --- | --- |
| Возрастные особенности детей | 1. Когнитивное развитие  В этом возрасте подростки начинают мыслить более абстрактно и критически, могут анализировать информацию и делать выводы, что позволяет им усваивать сложные концепции программирования.  2. Социальное взаимодействие  Подростки тяжело воспринимают критику и могут стесняться ошибки.  Групповые занятия и работа в команде помогут развить навыки сотрудничества.  3. Мотивация и интересы  Важен высокий уровень мотивации; подростки склонны интересоваться теми темами, которые связаны с их увлечениями. Мотивируйте тем, что благодаря данному обучению, дети смогут попасть в проект «Код будущего».  - Используйте современные инструменты и примеры из популярных игр и технологий для создания интереса.  4. Эмоциональное развитие  Подростки могут быть чувствительными к своим успехам и неудачам. Позитивная обратная связь и поддержка важны для их самооценки и желания продолжать обучение.  5. Самостоятельность  Подростки стремятся к независимости и могут проявлять инициативу в выборе идеи для итогового проекта. |
| Особенности построения занятий | 1. Активное вовлечение  Занятия не должны быть скучными; используйте увлекательные методы обучения, активное обсуждение, чтобы поддерживать интерес учеников.  2. Самостоятельная работа  Значительная часть занятия должна быть выделена на самостоятельную работу, где ученики могут реализовывать свои идеи.  Преподаватель задает общее направление разработки и курирует процесс с помощью наводящих вопросов.  3. Допущение ошибок  Позволяйте ученикам делать ошибки, затем помогайте им находить и исправлять их. Учитывайте, что уровень подготовки у детей может быть разный.  4. Обсуждение результатов  Регулярно обсуждайте с группой результаты работ учеников. Привлекайте их к тестированию и оцениванию продуктов друг друга для обмена опытом.  5. Физическая активность и отдых  Включайте перерывы на физическую активность, а также проводите упражнения для борьбы с усталостью глаз.  6. Ограничение времени за экраном. Время непрерывной работы за компьютером или планшетом не должно превышать 30-35 минут, чтобы предотвратить усталость и повысить продуктивность. |
| Методические пособия | Для преподавателя будут полезны следующие дополнительные материалы:   * примеры готовых проектов * распечатанные или выполненные полиграфическим способом правила поведения в группе. * распечатанные сценарии уроков. |
| Адаптация курса | Курс может гибко адаптироваться с учетом индивидуальных особенностей детей в группе.  Временные рамки разделов занятия, указанные в планировании, могут быть изменены преподавателем в зависимости от индивидуальных особенностей группы. Главным показателем эффективности занятия является не выполнение проектов по образцу, а понимание всеми ученикам ключевых понятий занятия и умение применять полученные знания на практике для реализации собственных идей. |
| Мотивация | Для дополнительной мотивации учеников рекомендуется использовать систему поощрений.  В качестве поощрений могут выступать отметки в импровизированных зачетных книжках.  Также важным элементом мотивации является возможность получить по итогам обучения красочный диплом или сертификат, подтверждающий профессионализм юного программиста.  Также важно подчеркивать престиж профессии программиста, почаще называя отличившихся учеников программистами и подчеркивая значимость их результатов.  Напоминайте детям, что данный курс поможет им поступить на понравившуюся программу в федеральном проекте «Код будущего» и успешно ее пройти. После чего они получат важный сертификат для дальнейшего поступления в ВУЗы. |
| Работа с родителями | Родители должны иметь возможность посмотреть результаты работы своих детей.  Их можно приглашать после окончания занятий в класс, чтобы дети имели возможность продемонстрировать близким людям результаты своей работы и получить заслуженную похвалу. Проговаривайте родителю в чем успех его ребенка по итогу занятий. |

**Аннотация**

JavaScript - один из самых популярных языков программирования в мире. Он используется для добавления интерактивности на веб-страницы, серверной разработки, создания интерактивных игр, в разработке мобильных приложений, также находит применение в IoT и работе с данными.

В ходе курса дети:

* Получат новые знания о программировании на языке JavaScript, будут знать основы работы в программе Visual Studio Code.
* Научатся создавать простые игры и приложения.
* Будут работать над реальными проектами, что поможет им развить навыки планирования, организации и выполнения задач.

Программирование требует логического мышления и способности решать проблемы, что способствует развитию этих навыков у детей. Изучение JavaScript не только подготовит детей к будущей карьере в IT, но и поможет им стать более уверенными и креативными в решении различных задач в других областях.

Прохождение данного модуля поможет подготовиться к поступлению в федеральный проект «Код будущего», что является отличным стартом для карьеры. В связи с быстро растущими требованиями рынка труда к IT-специалистам, обучение программированию играет важную роль в подготовке детей к будущему, вооружая их необходимыми навыками для успешной жизни в цифровом обществе.

**Учебно тематический план**

|  | **Тема занятия** | **Часы** | **Краткое описание** | **Основные понятия** | **Инструменты программирования** |
| --- | --- | --- | --- | --- | --- |
| 1 | Знакомство с JavaScript | 2 | Создание проекта с использованием canvas  **Проект по теме:**  Создание нового проекта и написание программы, которая выводит фигуры прямоугольной формы на Canvas | JavaScript, canvas, getContext | JavaScript |
| 2 | Типы данных. Создание примитивных объектов на canvas | 2 | Изучение примитивных типов данных (строка, число, булева) и построение примитивных объектов на странице (линии, круги, квадраты)  **Проект по теме:**  Создание нового проекта и написание программы, которая позволяет выводить фигуры сложной формы на Canvas | Изменяемые (let) и неизменяемые (const) переменные, строки, числа и булева, fillRect, strokeRect, moveTo, lineTo, arc, clearRect | JavaScript |
| 3 | Условия. Перемещение объектов | 2 | Покадровая отрисовка объектов. Использование requestAnimationFrame для обновления canvas. Использование события keypress для перемещения объектов  **Проект по теме:**  Создание нового проекта и написание программы, которая управляет движением персонажа в разные стороны | if, else, else if, requestAnimationFrame, addEventListener, event, keypress, keyCode | JavaScript |
| 4 | Механика прыжка | 2 | Изучение и реализация механики прыжка на базе положения элемента внутри экрана  **Проект по теме:**  Создание нового проекта и написание программы, которая реализует механику прыжка | условия выполнения прыжка, логическое И, логическое ИЛИ, гравитация | JavaScript |
| 5 | Добавление изображений | 2 | Использование функции drawImage для загрузки изображений  **Проект по теме:**  Создание нового проекта и написание программы, в которой происходит подгрузка заднего фона и спрайтов, а также реализован алгоритм передвижения объектов для сбора | drawImage, загрузка изображений, паралакс | JavaScript |
| 6 | Массивы. Спрайты | 2 | Изучение принципов работы массива. Обращение к определенному элементу массива.  Добавление анимации для объекта на canvas с использованием массивов и спрайтов  **Проект по теме:**  Создание нового проекта и написание программы, в которой загружаются спрайты для персонажа и создается эффект параллакса | массив, индекс, длина массива, спрайт | JavaScript |
| 7 | Циклы. Генерация элементов. Коллизия объектов | 2 | Генерация элементов на странице с использованием циклов. Работа с коллизией объектов  **Проект по теме:**  Создание нового проекта и написание программы, в которой создается лабиринт с коллизией стен | Коллизия, for..of, for, break, continue | JavaScript |
| 8 | GUI | 2 | Понятие графического интерфейса пользователя. Создание графического интерфейса  **Проект по теме:**  Создание нового проекта и написание программы, в которой создается мини-игра “Собирание падающих монет” с использованием графического интерфейса пользователя (GUI) | GUI, fillText, strokeText, font, textAlign, direction | JavaScript |
| 9 | Реализация стрельбы с использованием функций | 2 | Разбор синтаксиса функций. Написание алгоритма выстрела и перемещения пули  **Проект по теме:**  Создание нового проекта и написание программы, в которой реализован полет космического корабля и механика стрельбы для космического корабля | function, return, передаваемые параметры | JavaScript |
| 10 | Объекты  Генерация уровня | 2 | Создание и использование типа данных объект для создания элементов на canvas  Использование массива объектов для генерации и отображения уровня  **Проект по теме:**  Создание нового проекта и написание программы, которая создает анимацию для объектов сбора | объект, цикл for..in, свойства объекта, обращение к свойству объекта  Генерация уровня, массив строк | JavaScript |
| 11 | Canvas-трансформации | 2 | Использование canvas-трансформаций для создания анимации вращения и перемещения  **Проект по теме:**  Создание нового проекта и написание программы, в которой используются матрицы масштабирования, вращения и перемещения | save, restore, rotate, translate | JavaScript |
| 12 | Работа с пикселями | 2 | Использование canvas для создания фильтров для изображений  **Проект по теме:**  Создание нового проекта и написание программы, в которой происходит наложение фильтров на игровые объекты и работа с пикселями | getImageData, putImageData | JavaScript |
| 13 | Подключение аудио | 2 | Использование Audio Api для подключения звуковых эффектов  **Проект по теме:**  Создание нового проекта и написание программы, в которой добавляются звуковые эффекты и создается аудиоплеер | аудиоэффекты, AudioContext, | JavaScript |
| 14 | Введение в ООП | 2 | Основные понятия объектно-ориентированного программирование. Создание классов и их экземпляров  **Проект по теме:**  Создание нового проекта и написание программы, в которой создается игровой персонаж с использованием парадигмы ООП | Класс, экземпляр, абстракция, инкапсуляция, полиморфизм, наследование, class, new | JavaScript |
| 15 | Использование ООП при генерации игры | 2 | Создание персонажа игры, объектов игрового уровня с использованием ООП  **Проект по теме:**  Создание нового проекта и написание программы, которая создает игровой уровень с использованием парадигмы ООП | Класс, экземпляр, абстракция, инкапсуляция, полиморфизм, наследование, class, new | JavaScript |
| 16 | Подготовка к итоговому проекту | 2 | Подготовка к созданию и защите проекта  **Проект по теме:**  Начало разработки проекта | Планирование проекта, реализация, тестирование | JavaScript |
|  |  | 4 | Выполнение итогового проекта  Защита итогового проекта | Презентация, демонстрация, обсуждение | JavaScript |

**Ожидаемые результаты обучения**

*По окончанию модуля «JavaScript» учащиеся должны освоить:*

* работу в программе Visual Studio Code
* основные навыки программирования на языке JavaScript
* принципы создания и использования различных игровых механик с использованием языка JavaScript
* механику создания графического интерфейса
* принципы построения объектов с использованием ООП

*По окончанию курса «Код будущего: уверенный старт»*

* использовать среду разработки IDLE и различные библиотеки для разработки программ, включая создание графических приложений, игр и веб-скрапов
* разрабатывать и реализовывать функциональность ботов в Telegram, применяя основные принципы программирования
* работать в среде разработки Visual Studio Code и применять основные навыки программирования, включая создание графических интерфейсов и игровых механик.
* использовать принципы объектно-ориентированного программирования для построения объектов и реализации различных функций в своих проектах.
* осуществлять взаимодействие с базами данных через выполнение SQL-запросов для создания таблиц, вставки и выборки данных, а также разрабатывать клиент-серверные приложения, обеспечивая эффективное взаимодействие между сервером и клиентом.
* применять основные навыки программирования для разработки приложений на платформе Arduino, включая создание и тестирование программ, которые решают поставленные задачи модуля.
* работать со средой Wonkwi, демонстрируя понимание принципов проектирования и выполнения программ, а также интеграции их с аппаратным обеспечением Arduino.

**Бескомпьютерные активности:**

* В качестве бескомпьютерных активностей в курсе выступает формат обсуждений, вопросов для размышления, деление по группам для выполнения различных заданий.
* После завершения работы в группах, каждая группа презентует свои идеи, обсуждает с учителем. Учитель добавляет своими примерами.