

**Индивидуальный предприниматель
Селендеева Оксана Николаевна**

УТВЕРЖДАЮ

Индивидуальный предприниматель

_____/Селендеева О.Н./

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА
ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«ПРОГРАММИРОВАНИЕ НА JAVASCRIPT ДЛЯ ДЕТЕЙ»**

Москва, 2023

Оглавление

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	2
1.1. Общая характеристика программы	2
1.2. Цели и задачи программы	3
1.3. Планируемые результаты обучения	4
2. УЧЕБНЫЙ ПЛАН ПРОГРАММЫ	6
3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК	8
4. РАБОЧАЯ ПРОГРАММА	9
5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ	12
5.1. Контроль знаний, умений и навыков	12
5.2. Критерии оценивания освоения программы при проведении различных форм контроля:	13
6. УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ (УЧЕБНИКИ, РАЗДАТОЧНЫЕ МАТЕРИАЛЫ, МЕТОДИЧЕСКИЕ УКАЗАНИЯ, ПЛАКАТЫ, СЛАЙДЫ, ИНТЕРНЕТ- РЕСУРСЫ)	14
6.1. Информационные и учебно-методические условия реализации программы	14
6.2. Рекомендованная литература для освоения программы:	14
6.3. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее- сеть "Интернет"), рекомендованных для освоения программы:	15
7. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ..	15
7.1. Материально-техническая и ресурсная база	15
7.2. Кадровое обеспечение программы	15
Приложения. Примерные вопросы и задачи для проведения контроля знаний	17
Приложение 1. Примерные вопросы для промежуточного тестирования	17
Приложение 2. Примерные задания для оценки качества освоения учебного материала	21

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1.1. Общая характеристика программы

Данный документ описывает комплекс основных характеристик образования (объем, содержание, планируемые результаты) и организационно-педагогических условий, который представлен в виде учебного плана, календарного учебного графика, рабочих программ учебных предметов, модулей, иных компонентов, а также оценочных и методических материалов общеобразовательной общеразвивающей программы "Программирование на JavaScript для детей".

В ходе дополнительной общеобразовательной общеразвивающей программы "Программирование на JavaScript для детей" слушатели познакомятся с основными концепциями программирования, освоят язык программирования JavaScript. В ходе курса ученики получают активный навык самостоятельного создания программ различного назначения и степени сложности на языке программирования JavaScript.

Выдача обучающимся документов о дополнительном образовании (сертификат установленного образца) осуществляется при условии успешного прохождения итоговой аттестации.

Программа разработана на основе следующих **нормативных документов**:

- Федеральный закон от 29 декабря 2012 г. №273-ФЗ «Об образовании в Российской Федерации»;
- Приказ Министерства просвещения РФ от 9 ноября 2018 г. N 196 "Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам"
- Постановление Главного государственного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи" (вместе с "СП 2.4.3648-20. Санитарные правила...")» (Зарегистрировано в Минюсте России 18.12.2020 № 61573).

Направленность (профиль) программы: техническая

Актуальность программы, соответствие государственному, социальному заказу/запросам.

В современном мире умение программировать и работать с технологиями для веб-сайтов является **ценным и востребованным навыком**, который все больше и больше ценится на рынке, особенно сегодня, в условиях ограниченных ресурсов и повышенных расходов. В результате прохождения обучения слушатель сформирует знания в области основ программирования, разовьет логическое и алгоритмическое мышление, активизирует навыки использования полученных знаний и умений в практической деятельности. Обучающиеся получают востребованные знания для будущих специалистов по веб-разработке. Обучающимся будут созданы оптимальные условия для всестороннего удовлетворения потребностей и развития их индивидуальных склонностей и способностей, появится мотивация личности к познанию и творчеству. Обучение по данной программе будет полезно

для тех, кто интересуется развитием в области научно-технической направленности, будущих программистов, аналитиков, инженеров.

Отличительные особенности программы: В процессе освоения программы обучающиеся смогут в раннем возрасте получить профессиональную ориентацию по направлению веб-разработки.

Срок обучения: программа реализуется в объеме 32 академических часов, 12 недель (3 месяцев).

Режим занятий: 3-4 академических часа в неделю

Продолжительность академического часа – 45 минут.

Занятия начинаются не ранее 9.00 часов утра и заканчиваются не позднее 20.00 часов. Для обучающихся в возрасте 16-18 лет допускается окончание занятий в 21.00 часов.

Продолжительность занятий в учебные дни - не более 3-х академических часов в день, в выходные и каникулярные дни - не более 4 академических часов в день. После 30-45 минут теоретических занятий организуется перерыв длительностью не менее 10 мин.

Адресат программы и примерный портрет слушателя курсов: программа разработана для учащихся от 11 до 16 лет, которых интересует программирование на языке JavaScript. К освоению дополнительной общеобразовательной программы – дополнительной общеразвивающей программы допускаются: лица без предъявления требований к уровню образования.

По завершении реализации программы, как правило, проводится анкетирование обучающихся с целью изучения мнения по вопросу эффективности и информативности проведенного обучения, уровню организации учебного процесса, удовлетворенности учебно-методическим материалом, работниками образовательной организации проводится анализ высказанных предложений и пожеланий.

1.2. Цели и задачи программы

Цель программы – получение обучающимися знаний по основным концепциям программирования, а также навыков по написанию кода для взаимодействия с tml-страницами на языке программирования JavaScript.

Задачи программы:

1. Сформировать системные знания в области компьютерных технологий и программирования.
2. Сформировать навыки программирования на языке JavaScript.
3. Предоставить специализированные практические навыки написания программ на JavaScript.
4. Дать представление о профессии веб-разработчика.
5. Сформировать интерес к увлечению программированием и раскрытию своих способностей в сфере IT-технологий.

1.3. Планируемые результаты обучения

По итогам освоения дополнительной общеобразовательной программы - дополнительной общеразвивающей программы "Программирование на JavaScript для детей" обучающиеся должны будут овладеть следующими знаниями, умениями и навыками:

Знать:

- основы языка JavaScript
- основные концепции и понятия программирования
- типы данных в JavaScript
- правила создания и комментирования кода на языке JavaScript
- виды условных конструкций в JavaScript
- виды циклов в JavaScript
- логические переменные и логические операторы в JavaScript
- способы работы с разными типами данных в JavaScript
- понятие события в JavaScript
- понятие функции
- свойства и методы массивов в JavaScript
- рекомендации и технологии по созданию дизайна страниц и приложений
- технологию создания и настройки форм на html-страницах
- способы взаимодействия с элементами html-страницы с помощью JavaScript
- способы загрузки документов и изображений на html-страницу с помощью JavaScript.

Уметь:

- писать собственные скрипты и программы на JavaScript
- тестировать и отлаживать созданные программы для взаимодействия с html-страницами;
- создавать и стилизовать элементы веб-страницы с помощью JavaScript
- создавать программы на JavaScript с использованием условных конструкций
- создавать программы с использованием циклов на JavaScript
- применять комментарии в программах
- создавать программы с использованием логических переменных и операторов
- использовать массивы, применять методы обхода массивов в своих программах на JavaScript
- создавать функции в программах
- обрабатывать события на html-странице, связанные с движением мыши
- создавать формы на html-странице и настраивать их функционал
- выполнять загрузку текста на html-страницу с помощью JavaScript
- выполнять загрузку изображений на html-страницу с помощью JavaScript
- создавать и защищать свои проекты по разработке программного обеспечения
- применять развитое алгоритмическое и креативное мышление при решении задач программирования
- проводить техническую презентацию созданных проектов.

Владеть навыками в области:

- Разработки программного обеспечения на языке JavaScript
- Использования различных типов данных в программировании
- Применения функций в программах
- Обработки событий на html-странице с помощью JavaScript
- Технологии создания и настройки форм для приема и отправки данных со страницы
- Тестирования и отладки созданных программ на JavaScript
- Подготовки презентаций проектов и защиты проектов.

2. УЧЕБНЫЙ ПЛАН ПРОГРАММЫ

В процессе преподавания курса «Программирование на JavaScript для детей» используются как классические методы обучения (лекции), так и различные виды практической работы обучающихся по заданию преподавателя, которые направлены на развитие навыков разработки программного обеспечения, креативных качеств и на поощрение интеллектуальных инициатив учащихся.

Формы организации образовательного процесса (индивидуальные, групповые и т.д.) и другие виды занятий по программе определяются содержанием программы. Образовательная деятельность обучающихся предусматривает следующие виды учебных занятий и учебных работ: лекции, практические и семинарские занятия, мастер-классы, тренинги, семинары по обмену опытом, проведение открытых занятий, консультации и другие виды учебных занятий и методической работы, определенные учебным планом.

№	Наименование дисциплины/раздела/темы	Количество академических часов			СРС,	Форма аттестации /контроля
		Всего	в т.ч. аудиторных			
			теория	практич. занятия		
1	Знакомство с JavaScript	0,5	0,5		-	Практическое задание
2	Работа со строками. Операторы условий	2	0,5	1	0,5	Практическое задание
3	Работа с массивами. Знакомство с циклами в JavaScript	2,5	1	0,5	1	Практическое задание
4	Функции в JavaScript. Понятие объектов в JavaScript	4	1	1,5	1,5	Практическое задание, Контрольные вопросы
5	Взаимодействие с html-элементами с помощью JavaScript. Модель DOM	3	0,5	1	0,5	Практическое задание
6	Создание тегов в JavaScript	3	1	1	1	Практическое задание
7	События в JavaScript	3	1	1	1	Практическое задание
8	Загрузка документов и изображений на странице	3	1	1	1	Практическое задание, Контрольные вопросы
9	Работа с формами на странице	2	1	1	1	Практическое задание
10	Верстка формы на основе Material design by Google	3	1	1	1	Практическое задание

11	Отправка формы	2	1	0,5	0,5	Практическое задание
12	Знакомство с технологией AJAX. Итоги курса	4	0,5	1,5	2	Практическое задание. Контрольные вопросы. Защита проекта.
ИТОГО		32	10	11	11	

3. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Календарный учебный график формируется при осуществлении обучения в течение всего календарного года. По мере набора групп слушателей по программе составляется календарный график, учитывающий объемы лекций, практики, самоподготовки, другие формы организации занятий.

Темы / недели	1	2	3	4	5	6	7	8	9	10	11	12	Итого часов
1. Знакомство с JavaScript	0,5												0,5
2. Работа со строками. Операторы условий		2											2
3. Работа с массивами. Знакомство с циклами в JavaScript			2,5										2,5
4. Функции в JavaScript. Понятие объектов в JavaScript				3,5									3,5
Промежуточный контроль				0,5									0,5
5. Взаимодействие с html-элементами с помощью JavaScript. Модель DOM					3								3
6. Создание тегов в JavaScript						3							3
7. События в JavaScript							3						3
8. Загрузка документов и изображений на странице								2,5					2,5
Промежуточный контроль								0,5					0,5
9. Работа с формами на странице									2				2
10. Верстка формы на основе Material design by Google										3			3
11. Отправка формы											2		2
12. Знакомство с технологией AJAX. Итоги курса.												3	3
Итоговая аттестация												1	1
ИТОГО	0,5	2	2,5	4	3	3	3	3	2	3	2	4	32

4. РАБОЧАЯ ПРОГРАММА

1. Знакомство с JavaScript

- Знакомство с языком JavaScript, особенности, применение языка;
- Понятие переменной, объявление переменных, типы данных;
- Работа с JavaScript в браузере;
- Функции alert и prompt;
- Решение задач на арифметические вычисления и работу с разными типами переменных.

Практическое задание: работа с переменными, функциями prompt() и alert(), решение задач на пройденную тему.

2. Работа со строками. Операторы условий

- Методы и свойства строк;
- Решение задач со строками;
- Булевы переменные и логические операторы;
- Условный оператор в JS;
- Решение задач с применением условных операторов.

Практическое задание: написание программ с применением строк, условных, логических операторов.

3. Работа с массивами. Знакомство с циклами в JavaScript

- Понятие массива. Методы и свойства массивов;
- Понятие цикла;
- Цикл for;
- Методы перебора массива.

Практическое задание: решение задач на работу с массивами с применением различных методов массивов.

4. Функции в JavaScript. Понятие объектов в JavaScript

- Понятие функции. Написание функций в JS;
- Область видимости переменных;
- Объекты в JavaScript, создание объектов;
- Методы и свойства объектов.

Практическое задание: создать собственные функции для решения задачи, разделить программу на логические блоки и написать чистый и эффективный код.

5. Взаимодействие с html-элементами с помощью JavaScript. Модель DOM

- Знакомство с моделью Document Object Model (DOM)
- Метод getElementBy*;
- Изменение CSS-свойств тегов с помощью JS;
- Практика - написание кода на взаимодействие с различными HTML-элементами с помощью JavaScript.

Практическое задание: написание кода на работу с DOM, событиями, взаимодействие с HTML-элементами.

6. Создание тегов в JavaScript

- Различные методы вставки тегов в DOM;
- Изменение текстового контента;
- Оператор деструктуризации;
- Методы работы со стилями;
- Работа с атрибутами с помощью JS.

Практическое задание: моделирование работы методов append, prepend, before, after и replaceWith в различных задачах.

7. События в JavaScript

- Понятие события. Onclick;
- Onclick; Обработка события нажатия на кнопку;
- Обработка событий, связанных с действиями мыши;
- Методы addEventListener и removeEventListener.

Практическое задание: написание кода на обработку событий, применение обработчиков для различных тегов.

8. Загрузка документов и изображений на странице

- Отслеживание загрузки документа. Загрузка изображений;
- Событие scroll;
- Создание функции для ленивой загрузки изображений;
- Методы call и apply.

Практическое задание: написание кода для обработки загрузки страницы и ленивой загрузки изображений с использованием полученных знаний.

9. Работа с формами на странице

- Формы. Элементы форм;
- Взаимодействие с формами с помощью JS;
- Тег input в формах и его атрибуты;
- Начало работы с формой - написание кода для ввода данных в форму.

Практическое задание: создание формы, написание кода для обработки ввода данных в форму, стилизация элементов формы.

10. Верстка формы на основе Material design by Google

- Знакомство с концепцией Material design by Google;
- Верстка формы с применением принципов material design;
- Событие focus для настройки поведения формы.

Практическое задание: написание кода по отладке дизайна и работы элементов формы.

11. Отправка формы

- Доработка функциональности формы при получении данных;
- Автозаполнение полей;
- Знакомство с сервисом SimpleForm для отправки данных из форм;
- Загрузка готового проекта в репозиторий.

Практическое задание: доработка кода для формы, создание скрипта для отправки формы на почту.

12. Знакомство с технологией Ajax. Итоги курса

- Знакомство в AJAX - технологией обращения к серверу без перезагрузки страницы;
- Создание ajax-запросов с помощью средств JS;
- Доработка формы - проверка статуса запроса и работоспособности формы;
- Подведение итогов курса. Презентация созданных на курсе проектов;
- Дальнейшее развитие и обучение в сфере разработки на JavaScript.

Практическое задание: доработка формы, подготовка выступления и презентация работ, созданных на курсе.

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

5.1. Контроль знаний, умений и навыков

Формой подведения итогов реализации дополнительной образовательной программы "Программирование на JavaScript для детей" выступает текущая, промежуточная и итоговая аттестация.

Образовательный процесс осуществляется на основании учебного плана и регламентируется расписанием занятий для каждой учебной группы.

В целях оценки показателей знаний, умений и навыков обучающихся по дополнительной образовательной общеразвивающей программе "Программирование на JavaScript для детей" проводится текущий и промежуточный контроль знаний, а также итоговая аттестация.

Виды текущего контроля:

- устный ответ на поставленный вопрос;
- проверка результатов выполнения практических заданий

Виды промежуточного контроля:

- тестирование устное/письменное/с помощью электронных форм
- проверка результатов выполнения практических работ/проектов по итогам учебного модуля.

Тестирование - это форма измерения знаний обучающихся, основанная на применении тестов. Материалы для промежуточного и итогового тестирования предоставляются вместе с комплектом учебно-методических материалов к программе.

Итоговая аттестация

Итоговая аттестация проводится с целью установления уровня знаний обучающихся с учетом прогнозируемых результатов обучения и требований к результатам освоения образовательной программы.

Итоговая аттестация обучающихся осуществляется в форме презентации созданных в курсе проектов, которая сопровождается демонстрацией созданных проектов и скриптов и презентацией. Презентация – это электронный документ, предназначенный для визуальной демонстрации выполненной работы. Как правило, презентация имеет сюжет, сценарий и структуру, созданную для удобного восприятия информации.

Выдача обучающимся документов о дополнительном образовании (сертификат о прохождении курса) осуществляется при условии успешного прохождения итоговой аттестации.

5.2. Критерии оценивания освоения программы при проведении различных форм контроля:

Тестирование (Приложение 1. Примерные вопросы для промежуточного тестирования).
Процент результативности (правильных ответов при выполнении тестовых заданий):

Выполнение теста	Итоговая оценка
70% и более правильных ответов	"Зачтено"
Менее 70% правильных ответов	"Не зачтено"

Проверка выполнения практических работ (Приложение 2. Примерные задания для проверки усвоения качества учебного материала). Система оценивания:

"Зачтено" – необходимый уровень выполнения задания достигнут, обучающийся демонстрирует хорошее знание теоретической и практической части материала занятия/учебного модуля, достигнуты промежуточные и/или итоговые результаты работы над заданием.

"Не зачтено" - необходимый результат/уровень освоения не достигнут, обучающийся не усвоил теоретические основы и/или изученные практические приемы и методы разработки, инструменты создания чат-ботов, не достиг промежуточных и итоговых результатов при выполнении задания.

Проверка результатов демонстрации проектов на итоговой аттестации:

Критерии оценки	БАЛЛЫ
Портфолио с решенными задачами	0-4 балла
Применение изученных программных средств, техническая сложность программ	0-8 балла
Самостоятельность работы над проектом	0-4 балла
Компетентность докладчика (ответы на вопросы)	0-4 балла
Итоговая оценка: «Не зачтено» «Зачтено»	0-13 баллов 14-20 баллов

6. УЧЕБНО-МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ (УЧЕБНИКИ, РАЗДАТОЧНЫЕ МАТЕРИАЛЫ, МЕТОДИЧЕСКИЕ УКАЗАНИЯ, ПЛАКАТЫ, СЛАЙДЫ, ИНТЕРНЕТ-РЕСУРСЫ)

6.1. Информационные и учебно-методические условия реализации программы

Учебно-методический комплект

Для реализации целей и задач обучения по общеобразовательной общеразвивающей программы "Программирование на JavaScript для детей" используется комплект материалов преподавателя, который включает:

1. Текстовое методическое пособие с описанием теоретического материала и практических работ.
2. Раздаточный материал для учащихся - описание дополнительной самостоятельной работы учащихся по каждому занятию с примерами и рекомендациями по выполнению.
3. Описание мероприятий по контролю знаний – тестовые вопросы, практические задания.
4. Рекомендации по проведению итоговой аттестации и защиты проектов.
5. Дополнительные материалы – примеры кодов, скрипты.
6. Дополнительные материалы – инструкции по установке необходимого программного обеспечения, описание технических требований к компьютерному оборудованию.

Материалы преподавателя размещаются на учебном портале преподавателей, размещенном на сервере информационно-телекоммуникационной сети "Интернет", и доступны по ссылке для всех преподавателей курса. Материалы обучающихся раздаются в печатном виде или рассылаются преподавателем индивидуально каждому обучающемуся.

6.2. Рекомендованная литература для освоения программы:

Основная:

1. Морган, Н. JavaScript для детей. Самоучитель по программированию / Ник Морган; пер. с англ. Станислава Ломакина; [науч. ред. Д. Абрамова]. — М. : Манн, Иванов и Фербер, 2016. — 288 с.
2. Джереми Мориц. Учимся кодить на JavaScript/ Д.Мориц. — СПб: Питер, 2019 г. — 256 с.
3. Грацианова Т.Ю. Информатика. Программирование в примерах и задачах / Т.Ю. Грацианова, Москва: ВМК МГУ (Лаборатория знаний), 2020. – 393 с.

Дополнительная:

5. Эликс Том, Баррио Алексис. Роботоделы. Короткое замыкание в школе / Том Э., Алексис Б. В - Эксмо.: Москва, 2019. - 224 с.
6. Воронин, И. Программирование для детей. От основ к созданию роботов / И. Воронин, В. Воронина. - СПб.: Питер, 2019. - 304 с.
7. Е. В. Тимофеева, Наталья Авакян Наглядный школьный курс: Информатика. Удобно и понятно. / Е. В. Тимофеева, Н.Авакян, Москва: Эксмо, 2021. – 178 с. (электронное издание)

6.3. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее - сеть "Интернет"), рекомендованных для освоения программы:

- <https://javascript30.com/> - руководства и уроки по JavaScript
- <http://jsforcats.com/> - руководство для начинающих в изучении JavaScript
- <https://www.codecademy.com/catalog/language/javascript> - онлайн-платформа для изучения JavaScript
- <https://www.tutorialrepublic.com/javascript-tutorial/> - изучение основ JavaScript от базового уровня до продвинутого, реальные примеры из проектов и полезные ссылки.
- <https://learnjavascript.online/> - онлайн курс по JavaScript от Google
- http://window.edu.ru/catalog/resources?p_rubr=2.1.6 - единое окно доступа к образовательным ресурсам, раздел "Информатика и ОКТ"

7. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

7.1. Материально-техническая и ресурсная база

Для реализации программы предполагается использование учебных аудитории для проведения занятий лекционного типа, занятий семинарского типа, выполнения проектных работ, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации:

- Компьютерные классы, обеспечивающие доступ в Интернет, и оснащенные мультимедиа проектором или иными средствами визуализации учебного материала, магнитной доской или флипчартом.
- Электронный информационно-образовательный портал, размещенный на сервере в информационно-телекоммуникационной сети "Интернет".
- Специальное программное обеспечение для разработки на Java Script, необходимое для реализации образовательных задач курса.
- Стандартное программное обеспечение для работы над разработкой учебно-методических материалов.
- Мастерские и аудитории для проведения открытых занятий.

Специальных помещений, предполагающих наличие какого-либо специального оборудования для реализации данной программы, не предусматривается.

7.2. Кадровое обеспечение программы

Образовательный процесс по программе осуществляется педагогом дополнительного образования с профильным высшим или средним профессиональным образованием.

К занятию педагогической деятельностью по дополнительной общеобразовательной программе также допускаются лица, обучающиеся по образовательным программам высшего образования по специальностям и направлениям подготовки, соответствующим направленности дополнительных общеобразовательных программ, и успешно прошедшие промежуточную аттестацию не менее чем за два года обучения.

Реализация дополнительной общеобразовательной (общеразвивающей) программы обеспечивается руководящими и педагогическими работниками организации, а также лицами, привлекаемыми к реализации программы на условиях гражданско-правового договора.

У педагогического работника, реализующего дополнительную общеобразовательную программу, должны быть сформированы основные компетенции, необходимые для обеспечения успешного достижения обучающимися планируемых результатов освоения программы, в том числе умения:

- обеспечивать условия для успешной деятельности, позитивной мотивации, а также самомотивирования обучающихся;
- осуществлять самостоятельный поиск и анализ информации с помощью современных информационно-поисковых технологий;
- разрабатывать программы учебных предметов, выбирать учебники и учебно-методическую литературу, рекомендовать обучающимся дополнительные источники информации, в том числе Интернет-ресурсы;
- реализовывать педагогическое оценивание деятельности обучающихся;
- работать с текстовыми редакторами, электронными таблицами, электронной почтой и браузерами, мультимедийным оборудованием.

Приложения. Примерные вопросы и задачи для проведения контроля знаний

Приложение 1. Примерные вопросы для промежуточного тестирования

1. Почему код ниже не будет работать?

```
<script type="javascript/text">
  console.log("Hi!")
</script>
```

- Запись console.log необходимо прописывать лишь в отдельных файлах
- Неверно записан атрибут type
- Необходима точка с запятой после console.log("Hi!")

2. Какие функции выполняет JS?

- Создает разметку на странице сайта
- Выполняет работу с сервером
- Отвечает за работу с базами данных
- Создает стилевое оформление сайта
- Отвечает за функции на стороне клиента

3. Что будет записано в переменную test?

```
var a = 5;
var test = 5 != a ? "Yes" : "No";
"Yes"
```

- Будет ошибка
- 5
- a
- "No"

4. Где можно использовать JavaScript?

- Мобильные приложения
- Серверные приложения
- Прикладное программное обеспечение
- Веб-приложения
- Можно во всех перечисленных

5. Какая переменная записана неверно?

- var number = 12,5;
- var num = "STRING";
- var isDone = 0;
- var b = false;

6. Где верно указано имя переменной?

- `ver num;`
- `var num-1;`
- `var 2num;`
- `var num`
- `var num_1;`

7. Какие значения можно хранить в переменных?

- Только числа и строки
- Строки, числа с точкой, простые числа и булевы выражения
- Строки, числа с точкой и простые числа

8. В чем отличие между локальной и глобальной переменной?

- Глобальные можно переопределять, локальные нельзя
- Локальные видны повсюду, глобальные только в функциях
- Локальные можно переопределять, глобальные нельзя
- Глобальные видны повсюду, локальные только в функциях
- Отличий нет

9. В какой строке верно указан вывод данных?

- `prompt("Hello")`
- `documentWrite("Hello");`
- `write("Hello");`
- `console.log("Hello");`
- `print(Hello);`

10. Какие циклы есть в языке JavaScript?

- `for`, `while`, `do while`, `foreach`
- `for`, `forMap`, `foreach`, `while`, `do while`
- `for`, `while`, `do while`
- `for`, `forMap`, `foreach`, `while`

11. Что такое условный оператор?

- Оператор сравнения значений
- Конструкция для создания определенной переменной
- Конструкция, что выполняет код несколько раз

12. Как можно объявить функцию в Javascript?

- `function myFunction(...)`
- `myFunction function(...)`
- `declare function myFunction(...)`
- `var function myFunction(...)`

13. Как можно вызвать функцию myFunction в javascript ?

- call function myFunction(...)
- funcall myFunction(...)
- myFunction(...)
- call myFunction(...)

14. Где верно указан запуск всплывающего окна?

- Нет верных вариантов
- info ("Hi")
- alert ("Hi")
- new alert ("Hi")

15. Какое количество сообщений будет выведено в консоль?

```
for(var i = 10; i < 35; i += 5) {  
  console.log(i);  
}
```

- 25
- Такой цикл работать не будет
- 15
- 5
- 6

16. Где встраивается javascript?

- внутри <head>
- внутри <html>
- внутри <body>
- внутри

17. Выберите правильный способ использования "If" в javascript

- (if ...){...}
- if(...){...}
- call if (...) {...}

18. Выберите правильный способ объявления массива(array) в Javascript

- var names = ['leto', 'zima', 'vesna']
- var names = array('leto', 'zima', 'vesna')
- var name = 'leto', 'zima', 'vesna'
- var names = [0]=> 'leto', [1]=> 'zima',[2]=> 'vesna'

19. Какой результат будет иметь "variable" ?

```
var variable;  
if ( 10 > 4){  
variable = true;  
}  
else {  
variable = 55;  
}
```

- true
- 55
- Undefined

20. Javascript не может изменить html-тег. Верно ли это утверждение?

- Да, верно
- Нет, неверно

21. С помощью javascript можно установить любой стиль для html-тега. Верно ли это утверждение?

- Да
- Нет

Приложение 2. Примерные задания для оценки качества освоения учебного материала

Задание 1. Используя знания о записи арифметических операторов, сократите запись указанных выражений с помощью операторов +=, *=, -=, /=.

Сделайте вывод в консоль, убедитесь в правильности записи:

a = a + 10 привести к виду a += 10

a = a * 2 привести к виду a *= 2

a = a - 5 привести к виду a -= 5

a = a / 3 привести к виду a /= 3

Задание 2. Реализуйте в консоли следующий код для подсчета площади прямоугольника. Измените код для подсчета: количества секунд в минутах, площади круга, объема параллелепипеда, средней скорости, получая от пользователя нужные данные.

```
var a = +prompt('Введите ширину прямоугольника'),
    b = +prompt('Введите длину прямоугольника'),
    c = a * b;
alert('Площадь прямоугольника равна ' + c);
console.log('Площадь прямоугольника равна ' + c);
```

Задание 3. На вход подается два числа. Напишите программу, которая печатает все числа, расположенные между этими двумя числами

Задание 4. С помощью изученных методов работы со строками реализуйте следующие действия со строкой – подсчет длины строки, перевод символов в верхний регистр и обратно, поиск подстроки, замену подстроки.

Пример реализации:

```
var string = 'JS is the best programming language!',
    indexOfBest = string.search('best');
console.log(indexOfBest);
```

Задание 5. Используя изученные методы работы с массивами, реализуйте следующие задачи:

- Найти произведение всех элементов массива.
- Найти произведение только четных элементов
- Найти произведение только положительных элементов
- Найти произведение элементов последовательности, введенной пользователем

Задание 6. Модифицируйте созданный на занятии код для объекта car. Необходимо:

- добавить для car свойство fuelCapacity, fuel(емкость бензобака и текущее количество литров в нем)
- усовершенствовать метод drive: добавить расход топлива (можно взять 1 единица времени == 1 единица топлива) и если топлива не хватает, то останавливаемся.

При выполнении задания опирайтесь на следующий код:

```
var car = {
  color: 'red',
  brand: 'BMX',
  isDriving: false,
  speed: 120,
  drive: function(time) {
    console.log('Поехали');
    this.isDriving = true;
    distance = time * this.speed;
    console.log('Проехали ' + distance + ' км');
    this.stop();
    return distance;
  },
  stop: function() {
    this.isDriving = false;
    console.log('Плавная остановочка');
  }
}
```

Задание 7. Закрепите работу с html тегами, изученными на занятии.

Необходимо создать 10 параграфов с рандомным цветом текста и контентом, содержащим информацию о цвете.

- Вставить 10 дивов и в каждый из них по параграфу, содержащему номер дива и свой цвет.
- Вставить тег style с css-стилями в тег head

Пример базовой реализации задания:

```
var p = document.getElementById('text1');
p.textContent = 'Я новый текст!';

for (var i = 0; i < 10; i++) {
  var text = document.createElement('p'),
      color = '#' + (0x1000000 + (Math.random() * 0xfffff).toString(16).substr(1,6));

  text.style.color = color;
  text.textContent = 'Мой цвет: ' + color;
  document.body.insertBefore(text, p);
}
```

Задание 8. Самостоятельно реализуйте изученный на занятии код "ленивой" загрузки изображений на html-страницу. Опирайтесь на следующий код:

```
document.addEventListener("DOMContentLoaded", function(event) {
  console.log('Страница загружена');
  var img = document.getElementById('image1');
  if (img.hasAttribute('data-src')) {
    var data = img.getAttribute('data-src');
    console.log('Старый путь у картинки: ' + img.getAttribute('src'));
    img.setAttribute('src', data);
    console.log('Новый путь у картинки: ' + img.getAttribute('src'));
    img.removeAttribute('data-src');
  }
});
```

Задание 9. Промоделируйте работу методов append, prepend, before, after и replaceWith на своем примере.

Например, создайте текстовый список изученных тем по программированию и обработайте работу указанных методов на этом тексте.

Задание 10. Самостоятельно создайте html-страницу с формой ввода данных от пользователя (имя, номер заказа, адрес электронной почты, кнопка "Отправить"). Стилизируйте форму. Добавьте вид курсора при наведении, и эффект при нажатии на кнопку.