

Комитет по образованию администрации  
Всеволожского муниципального района Ленинградской области  
**Муниципальное общеобразовательное бюджетное учреждение**  
**«Кудровская средняя общеобразовательная школа № 3»**  
(МОБУ «Кудровская СОШ № 3»)

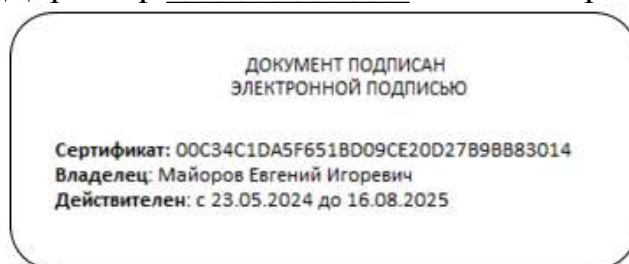
---

**ПРИНЯТО**  
на Педагогическом совете  
МОБУ «Кудровская СОШ № 3»  
Протокол № 1  
от «29» августа 2024 года

**УТВЕРЖДЕНО**  
Приказом  
МОБУ «Кудровская СОШ № 3»  
от «29» августа 2024 года № 267

**СОГЛАСОВАНО**  
на Совете родителей  
МОБУ «Кудровская СОШ № 3»  
Протокол № 1  
от «26» августа 2024 года

Директор \_\_\_\_\_ Е.И. Майоров



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА**  
**«Игрострой. Создание Web-игр (HTML, CSS, JavaScript)»**

Автор (составитель): **Баннова Антонина Юрьевна,**  
педагог дополнительного образования  
Направленность программы: **техническая**  
Уровень программы: **начальный**  
Возраст детей, осваивающих программу: **12-14 лет**  
Срок реализации программы: **1 год**

Кудрово  
2024

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеразвивающая программа «Игрострой. Создание Web-игр (HTML, CSS, JavaScript)» составлена на основании:

- Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- Основы государственной политики по сохранению и укреплению традиционных российских духовно-нравственных ценностей (утв. Указом Президента РФ от 9 ноября 2022 г. № 809);
- Концепция развития дополнительного образования до 2030 года (утв. распоряжением Правительства РФ от 31.03.2022 № 678-р);
- Целевая модель развития региональных систем дополнительного образования детей (утв. Приказом Министерства просвещения РФ от 03.09.2019 № 467);
- Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам (утв. Приказом Министерства просвещения РФ от 27.07.2022 № 629);
- СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи» (утв. Постановлением Главного государственного санитарного врача РФ от 28.09.2020 № 28);
- Устав МОБУ «Кудровская СОШ №3»;
- Положение о дополнительных общеразвивающих программах.

### Актуальность программы

Невозможно представить современное общество без различных электронных устройств, нуждающихся в программном обеспечении. Многие профессии сейчас связаны с использованием программ, все от инженера до медика нуждаются в специальном софте. Кроме того, элементы алгоритмизации и программирования входят в нашу жизнь: составить меню или график тренировок, руководить бизнес-процессами и многое другое проще делать, зная теорию алгоритмов. С каждым годом программирование все больше и больше проникает в нашу жизнь. Обучать детей программированию интереснее через создание игр, так как компьютерные игры близки детям. Язык программирования JavaScript – один из современных языков программирования. За два десятилетия, прошедших с момента его создания он завоевал немало приверженцев среди программистов и используется для решения многих прикладных задач. Определяющими факторами является удачный синтаксис языка и наличие большого числа готовых модулей и библиотек «на все случаи жизни». Так же незаменима его роль в Web-дизайне. И знания HTML, CSS и JavaScript позволяют детям в будущем освоить современную профессию фронтенд-разработчика.

**Отличительной особенностью** программы является то, что изучение основ программирования ведется через игровую и проектную деятельность.

Ключевыми навыками обучающегося в современных условиях становятся способность принимать решения на перспективу, анализировать собственные ценности, потребности и ресурсы для их реализации, планирование своей деятельности и прогнозирование возможных результатов и рисков.

Обучающиеся с достаточной степенью свободы и самостоятельности могут выбирать способы решения проблем, поставленных в программе. В курсе предусмотрена работа в парах и командах. Обязательное условие успешного прохождения курса – публичная презентация и защита результатов работы над проектами. В программе предполагается овладение следующими основными soft-компетенциями:

1. Креативностью и творческим воображением
2. Критическим и системным мышлением
3. Умением решать проблемы
4. Умением работать в команде
5. Умением работать с информацией
6. Стремлением к достижениям и т.д.

Занятия по данной программе могут проводиться как в очной форме, так и с применением дистанционных технологий и (или) электронного обучения.

Аспект новизны заключается в том, что создание Web-игр не просто написание кода на языке программирования, а еще и интерактивная среда, где результаты действий визуализированы, что делает работу с программой понятной, интересной и увлекательной.

Особенность данной программы в проектной деятельности учащихся, создании мультфильмов, анимации и даже простейших игр, делает дополнительную общеобразовательную общеразвивающую программу практически значимой для современного подростка, т.к. дает возможность увидеть практическое назначение алгоритмов и программ, что будет способствовать развитию интереса к профессиям, связанным с программированием.

Педагогическая целесообразность данной образовательной программы состоит в том, что изучая программирование у обучающихся формируется не только логическое мышление, но и навыки работы с мультимедиа; создаются условия для активного, поискового учения, предоставляются широкие возможности для разнообразного программирования.

**Основной вид деятельности:** игра. Также на занятиях практикуется учебная, познавательная и творческая деятельность.

**Цель программы** – активизировать познавательную активность обучающихся, способствовать формированию и развитию у обучающихся основ алгоритмического мышления и проектной деятельности в области программирования.

Программа нацелена на развитие логического, алгоритмического и системного мышления, создания предпосылок успешного освоения учащимися инвариантных фундаментальных знаний и умений в областях, связанных с математикой и информатикой, которые вследствие непрерывного обновления и изменения в аппаратных и программных средствах выходят на первое место в формировании научного информационно-технологического потенциала общества

**Задачи:**

***Обучающие***

Овладение базовыми понятиями объектно-ориентированного программирования и применение их при создании проектов;

Приобщение обучающихся к новым технологиям, способным помочь им в реализации собственного творческого потенциала;

Совершенствование навыков работы на компьютере и повышение интереса к программированию;

Формирование культуры и навыков сетевого взаимодействия.

***Развивающие***

Развитие познавательной деятельности учащихся в области новых информационных технологий;

Способствование развитию творческих способностей и эстетического вкуса подростков;

Способствование развитию коммуникативных умений и навыков обучающихся;

Способствование развитию логического мышления, памяти и умению анализировать;

Формирование потребности в саморазвитии;

Способствование развитию познавательной самостоятельности.

***Воспитательные***

Создание условия для повышения самооценки обучающегося, реализации его как личности;

Воспитание аккуратности, усидчивости и трудолюбия, ответственность при выполнении любой работы, целеустремленность и самостоятельность;

Развитие способности к сопереживанию;

Содействовать в формировании опыта творческой деятельности, навыка ценностных отношений;

Содействовать в позитивной социализации личности;

**Организационно-педагогические условия**

Организационно-педагогические условия реализации общеразвивающей программы обеспечивают ее реализацию в полном объеме, качество подготовки обучающихся, соответствие применяемых форм, средств, методов обучения и воспитания возрастным,

психофизическим особенностям, склонностям, способностям, интересам и потребностям обучающихся.

**Срок реализации программы – 1 год.**

**Возраст учащихся, на которых рассчитана программа – 12-14 лет.**

**Наполняемость группы – 20 детей.**

### **Календарный учебный график**

Год обучения	Период обучения		Период каникул	Кол-во учебных недель / часов	Режим занятий	Вид и сроки проведения аттестации
	начало	окончание				
1 год	16.09	31.05	30.12.2024-08.01.2025	34 / 136	2 занятия в неделю по 2 акад. часа (45 минут)	промежуточная, май

**Форма обучения – очная.**

**Форма организации образовательной деятельности учащихся – групповая.**

**Формы занятий:** аудиторные, внеаудиторные и самостоятельные занятия.

### **Условия реализации программы**

Зачисление в группы осуществляется по желанию детей и на основании личных заявлений родителей (законных представителей).

*Кадровое обеспечение:* педагог дополнительного образования

*Психолого-педагогическое обеспечение:*

- учет возрастных и индивидуальных особенностей учащихся;
- создание ситуации успеха для каждого обучающегося;
- поддержка психологического здоровья обучающихся;
- психолого-педагогическое просвещение родителей;

*Материально-техническое обеспечение:*

- кабинет;
- мебель, соответствующая возрасту обучающихся (столы, стулья);
- персональный компьютер;
- ноутбук;
- принтер;
- колонки;
- доска для письма;

*Информационное обеспечение:*

- методические рекомендации для педагога;
- дидактические материалы для работы с обучающимися (рабочие тетради, книги, дидактическая литература);
- электронные образовательные ресурсы.

## Планируемые результаты

Основные **личностные результаты**, формируемые в процессе освоения программы:

- формирование ответственного отношения к учению, способности довести до конца начатое дело на примере завершённых творческих учебных проектов;
- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой благодаря иллюстративной среде программирования мотивации к обучению и познанию;
- развитие опыта участия в социально значимых проектах, повышение уровня самооценки, благодаря реализованным проектам;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;
- развитие эстетического сознания через творческую деятельность на базе среды программирования.

К основным **метапредметным** результатам (осваиваемым обучающимися межпредметным понятиям и универсальным учебным действиям, способности их использования как в учебной, так и в познавательной и социальной практике), формируемые в процессе освоения программы, можно отнести:

- умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата, понимая, что в программировании длинная программа не значит лучшая программа;
- умение оценивать правильность решения учебно-исследовательской задачи;
- умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
- владение основами самоконтроля, принятия решений;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
- ИКТ-компетенцию;
- умение сотрудничества и совместной деятельности со сверстниками в процессе проектной и учебно-исследовательской деятельности.

Основные предметные результаты, формируемые в процессе изучения программы направлены на:

- осознание значения математики и информатики в повседневной жизни человека;
- формирование представлений об основных предметных понятиях — «информация», «алгоритм», «модель» и их свойствах;
- развитие логических способностей и алгоритмического мышления, умений составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами — линейной, условной и циклической;
- развитие представлений о числах, числовых системах;
- овладение символьным языком алгебры, умение составлять и использовать сложные алгебраические выражения для моделирования учебных проектов, моделировать реальные ситуации на языке алгебры;
- развитие пространственных представлений, навыков геометрических построений и моделирования таких процессов, развитие изобразительных умений с помощью средств ИКТ;
- формирование информационной и алгоритмической культуры, развитие основных навыков использования компьютерных устройств и программ;
- формирование умения соблюдать нормы информационной этики и права.

### **Система оценки результатов освоения программы**

Для отслеживания результативности данной программы используются следующие формы контроля:

- Опрос;
- Выполнение проектов;
- Текущий контроль, включающий в себя анализ выполненных работ; участие в конкурсах и олимпиадах.

## Механизм оценивания результатов освоения программы

Оценки Оцениваемые параметры	Низкий	Средний	Высокий
<i>Уровень теоретических знаний</i>			
	Обучающийся знает фрагментарно изученный материал. Изложение материала сбивчивое, требующее корректировки наводящими вопросами.	Обучающийся знает изученный материал, но для полного раскрытия темы требуется дополнительные вопросы.	Обучающийся знает изученный материал. Может дать логически выдержанный ответ, демонстрирующий полное владение материалом.
<i>Уровень практических навыков и умений</i>			
Решение практических задач, техника безопасности	Требуется постоянный контроль педагога за выполнением правил по технике безопасности.	Требуется периодическое напоминание о том, как работать с оборудованием.	Четко и безопасно работает с оборудованием.
Способность воспроизвести решение по образцу	Не может решить по образцу без помощи педагога.	Может решить по образцу при подсказке педагога.	Способен решить по образцу.
Степень самостоятельности выполнения проекта	Требуется постоянные пояснения педагога.	Нуждается в пояснении последовательности работы, но способен после объяснения к самостоятельным действиям.	Самостоятельно выполняет операции при проектировании.
<i>Качество выполнения работы</i>			
	Проект в целом получен, но требует серьезной доработки.	Проект требует незначительной корректировки	Проект не требует исправлений.

Промежуточная аттестация проводится в конце учебного года посредством выполнения проектной творческой работы с использованием полученных знаний в области программирования. Итоговой аттестации нет.



## УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№ п/п	Название раздела	Количество часов			Формы контроля
		Всего	Теория	Практика	
1.	Основы языка разметки HTML	24	14	10	тест
2.	Основы таблиц стилей CSS	16	10	6	тест
3.	Практикум	28	10	18	Практические работы
4.	Интерактивность HTML	12	8	4	тест
5.	Интерактивность CSS	20	10	10	тест
6.	Основы языка программирования JavaScript	20	12	8	тест
7.	Практикум	16	4	12	Практические работы
	<b>Итого</b>	<b>136</b>	<b>68</b>	<b>68</b>	

В результате освоения программы, учащиеся **получат представление о:**

- свободно распространяемых программах;
- назначении и использовании основных блоков команд, состояний, программ;
- правилах сохранения документа и необходимости присвоения правильного имени;
- возможности и способах отладки написанной программы;
- исполнителях и системах их команд, возможности непосредственного управления исполнителем;
- наличии заготовок для персонажей и сцен в соответствующих библиотеках, иерархическом устройстве библиотек и возможности импортирования их элементов;
- возможности использования встроенного растрового редактора, наличии и назначении основных инструментов;
- использовании других программ для создания собственных изображений;
- алгоритме как формальном описании последовательности действий исполнителя, приводящих от исходных данных к конечному результату;
- использовании схематического описания алгоритма;
- программном управлении исполнителем и линейных алгоритмах;
- написании программ для исполнителей, создающих геометрические фигуры на экране в процессе своего перемещения;
- необходимости программного прерывания;
- использовании циклических команд при необходимости повторений однотипных действий;
- видах циклических алгоритмов и их применении;
- достижении эффекта перемещения путем использования циклов;

- возможности распараллеливания однотипных действий за счёт использования нескольких исполнителей;
- организации интерактивности программ;
- возможности взаимодействия исполнителей между собой, в различных слоях изображения;
- видах и формах разветвленных алгоритмов, включая циклы с условием;
- управлении событиями.
- использовании метода проектов для моделирования объектов и систем;
- возможности описания реальных задач средствами программной среды;
- создании анимационных, игровых, обучающих проектов, а также систем тестирования.

Учащиеся будут уметь:

- самостоятельно устанавливать программную среду на домашний компьютер;
- изменять некоторые стандартные установки пользовательского интерфейса (например, язык отображения информации);
- использовать различные способы отладки программ, включая пошаговую отладку;
- уверенно использовать инструменты встроенного графического редактора, включая работу с фрагментами изображения и создание градиентов;
- создавать собственные изображения в других программах и импортировать их в программную среду;
- использовать графические примитивы векторного редактора для создания объектов;
- создавать изображения из пунктирных и штрих-пунктирных линий с изменением цвета и толщины линии;
- упрощать программы за счёт использования циклических команд и применять их;
- составлять простые параллельные алгоритмы;
- создавать программы и игры с использованием интерактивных технологий;
- моделировать ситуации с использованием необходимых форм ветвления алгоритма, включая цикл по условию;
- передавать сообщения исполнителям для выполнения последовательности команд (включая разные типы исполнителей).
- планировать и создавать анимации по определенному сюжету;
- создавать игры, используя интерактивные возможности программной среды;
- планировать и создавать обучающие программы для иллюстрации пройденного материала других предметных областей;

- продумывать и описывать интерактивное взаимодействие для создания простейших тренажеров;

- подходить творчески к построению моделей различных объектов и систем.

- Полученные по окончании программы знания и умения могут способствовать развитию интереса к профессиям, связанным с программированием, анимацией, мультипликацией.

На занятиях обращается внимание на соблюдение требований безопасности труда, пожарной безопасности и личной гигиены.

### **Технологии и методики**

Рекомендуемыми технологиями, используемыми в процессе реализации общеразвивающей программы - это *технологии организации проектно-исследовательской деятельности* обучающихся и применение *игровой технологии*.

Учебный исследовательский проект с точки зрения педагога — это интегративное дидактическое средство развития, обучения и воспитания, которое позволяет вырабатывать и развивать специфические умения и навыки проектирования и исследования у обучающихся:

- целеполаганию и планированию содержательной деятельности ученика;
- проблематизации (рассмотрению проблемного поля и выделению подпроблем, формулированию ведущей проблемы и постановке задач, вытекающих из этой проблемы);
- самоанализу и рефлексии (результативности и успешности решения проблемы проекта);
- представлению результатов своей деятельности и хода работы; презентации в различных формах, с использованием специально подготовленного продукта проектирования;
- поиску и отбору актуальной информации, и усвоению необходимого знания; практическому применению приобретённых знаний в различных, в том числе и нетиповых, ситуациях; выбору, освоению и использованию подходящей технологии изготовления продукта проектирования; проведению исследования (анализу, синтезу, выдвижению гипотезы, детализации и обобщению).

Использование *технологии организации проектно-исследовательской деятельности* позволяет развивать познавательные и творческие навыки. Самостоятельная работа над техническим проектом дисциплинирует обучающихся, заставляет мыслить критически и дает возможность развивать у обучающихся пространственное воображение.

*Игровая технология* – это способ обучения с применением на занятиях игр. Игра – вид деятельности, при котором в процессе игровой ситуации решается учебная задача.

Целесообразными методами, используемыми в процессе реализации общеразвивающей программы, являются: метод взаимообучения и метод проблемного обучения.

*Метод взаимообучения* своими истоками уходит в коллективный способ обучения. По мнению В.К. Дьяченко, обучение есть общение обучающихся и обучаемых. Вид общения определяет и организационную форму обучения. На занятиях, разобравшись в решении какой-либо конструкторской задачи, обучающиеся с удовольствием делятся своими знаниями с теми, кто испытывает затруднения при решении подобных задач.

*Метод проблемного обучения* основан на создании проблемной мотивации и требует особого конструирования дидактического содержания материала, который должен быть представлен как цепь проблемных ситуаций. Этот метод позволяет активизировать самостоятельную деятельность обучающихся, направленную на разрешение проблемной ситуации, в результате чего происходит творческое овладение знаниями, навыками, умениями и развитие мыслительных способностей. Практически каждую задачу, решаемую в процессе программирования, можно представить в качестве проблемной ситуации. Активизируя творческое и критическое мышление, обучающиеся способны оптимизировать собственное решение задачи.

### **Формы работы:**

Основной формой работы является учебно-практическая деятельность.

Применяются следующие формы работы с учащимися:

- занятия, творческая мастерская, собеседования, консультации, обсуждения, самостоятельная работа на занятиях;
- посещение музеев, выставок, экскурсии;
- выставки работ, конкурсы, как местные, так и выездные;
- мастер-классы.

Достижение поставленных целей и задач программы осуществляется в процессе сотрудничества обучающихся и педагога. На различных стадиях обучения ведущими становятся те или иные из них. Традиционные методы организации учебного процесса можно подразделить на: словесные, наглядные (демонстрационные), практические, репродуктивные, частично-поисковые, проблемные, исследовательские.

## **Содержание программы**

### **Раздел 1. Основы языка разметки HTML**

Свободное программное обеспечение. Параметры для скачивания и установки программной среды на домашний компьютер.

Основные элементы пользовательского интерфейса программной среды. Внешний вид рабочего окна. Блочная структура систематизации информации. Функциональные блоки. Блоки команд, состояний, программ, запуска, действий и исполнителей. Создание и сохранение документа. Очистка экрана.

Знакомство с технологиями веб-разработки.

*Теоретическая часть*

- Обзор базовых технологий разработки. Выбор инструментов. Знакомство с HTML. Создание тестовой первой страницы.
- Структура HTML-документа. Содержимое блока заголовков
- Изучение структуры веб-страницы, возможного содержимого блока <head>.
- Мета-теги. Комментарии.
- Определение структуры и дизайна индивидуального проекта.
- Обзор тегов форматирования текста. Теги физического и логического форматирования текста, списков. Организация рабочего места разработчика.
- Изучение типов гиперссылок и вариантов их применения. Файловая структура. Внутренние и внешние ссылки. Знакомство со ссылками-якорями.
- Углубленное изучение параметров тега изображения. Единицы измерения. Изучение вариантов форматирования изображений в HTML.
- Форматы изображений для веб-дизайна и принципы их использования. Растр и вектор. Принципы подготовки изображений для веб-дизайна. Анимированные изображения. Вставка видео на веб-страницу.
- Изучение средств HTML для создания таблиц. Варианты применения таблиц. Объединение и форматирование ячеек.
- Изучение вариантов определения цветов в HTML. Обзор теории цвета.
- Цветовой круг, модель Иттена. Обзор принципов создания гармоничных цветовых схем.

#### *Практическая деятельность*

- Подбор материалов для индивидуального проекта.
- Разработка блока заголовков индивидуального проекта. Создание индивидуальных элементов (заголовков, ключевых слов, иконок и т. д.) своих страниц.
- Форматирование текстового контента.
- Размещение контента в индивидуальном проекте. Создание перекрестных ссылок
- Подготовка и размещение изображений в индивидуальном проекте.
- Обработка изображений. Вставка видео и фрейма
- Организация табличного представления информации.
- Подбор цветовой схемы для индивидуального проекта
- Выбирать и запускать программную среду;
- Работать с основными элементами пользовательского интерфейса программной среды;

- Изменять размер и перемещать окно программы, выбирать необходимый режим окна;
- Вводить имя файла с помощью клавиатуры;
- Выбирать необходимый файл из нужной папки библиотеки программы;
- Создавать, копировать, переименовывать, перемещать, копировать и удалять файлы;
- Соблюдать требования техники безопасности при работе в компьютерном классе.

## **Раздел 2. Основы таблиц стилей CSS**

Компьютерная верстка. Обзор назначения, синтаксиса и применения CSS. Изучение вариантов размещения стилей. Размещение учащимися исходного контента: текста и изображений на своих страницах. Форматирование контента и применение стилей. Разработка дизайна в графических редакторах. Векторные и растровые графические редакторы. Основные инструменты графического редактора — кисточка, ластик, заливка (цветом или градиентом), рисование линий, прямоугольников, квадратов, эллипсов и окружностей, выбор фрагмента изображения и отражение его по горизонтали или вертикали, использование инструмента печать для копирования выделенной области изображения, работа с текстом. Масштаб фрагмента изображения. Палитра цветов, установка цвета переднего плана и фона, выбор цвета из изображения с помощью инструмента пипетка.

### *Теоретическая часть:*

- Обзор назначения, синтаксиса и применения каскадных стилей. Изучение вариантов размещения стилей.
- Виды селекторов CSS: селекторы тегов, классов, идентификаторов, групп. Понятие класса. Селекторы дочерних и сестринских элементов, потомков. Понятие наследования. Каскадность стилей. Расчет и определение приоритетов стилей.
- Выделять фрагменты изображения для дальнейшей работы с ними;
- Планировать работу по созданию сложных изображений путем копирования и масштабирования простых;
  - Выбирать наиболее подходящий инструмент графического редактора для создания фрагмента изображения;
  - Различать верхний и нижний цвета изображения;
  - Придумывать и создавать различные градиенты для заливки замкнутой области;
- Планировать создание симметричных изображений. Изучение вариантов определения цветов в HTML. Обзор теории цвета.
- Цветовой круг, модель Иттена. Обзор принципов создания гармоничных цветовых схем.

- Обзор применения CSS стилей для текста и списков. Цвет, шрифт, размер, начертание, трансформация, декорирование текста. Интервалы, выравнивание.
- Обзор применения CSS стилей для графических элементов. Фоновые изображения: повтор, позиция, размер. Цвет и градиент. Рамки, отступы, тени. Примеры применения различных эффектов к изображениям.
- Изучение применения блоков-контейнеров и их видов при взаимном расположении элементов. Внешние и внутренние отступы. Блочные, строчные, гибридные элементы. Подготовка к блочной верстке веб-страницы. Практика применения отступов, обтекания. Понятие потока документа.

#### *Практическая деятельность:*

- Применение стилей в индивидуальном проекте. Подключение шрифтов Google
- Применение стилизованных правил к основному контенту индивидуального проекта
- Использовать простейшие растровые и векторные редакторы для создания и редактирования изображений;
  - Изменять центр изображения;
  - Вносить изменения в изображения из встроенной библиотеки;
  - Создавать сложные графические объекты путем копирования и модификации простых объектов и их фрагментов,
  - Использовать возможности работы с цветом.
  - Подбор цветовой схемы для индивидуального проекта.
  - Создание стилизованных правил для текстов в индивидуальном проекте.
  - Применение стилей к изображениям в индивидуальном проекте.
  - Подготовка материалов к верстке одностраничного сайта

### **Раздел 3. Практикум**

Объединение содержимого HTML и разметки CSS. Создание базовых стилей веб-страницы. Вопросы адаптивности. Размещение блоков контента на страницах индивидуального проекта. Форматирование контента и применение стилей. Объединение содержимого HTML и разметки CSS. Разработка шапки и навигационных панелей. Разработка стилей блоков основного контента. Обтекание, вычисление ширины блоков. Резиновая верстка. Разработка подвала. Форматирование контента и применение стилей. Типы верстки. Изучение принципов адаптивной верстки. Наборы стилей. Медиа-запросы. Ограничения по ширине и другим параметрам экрана. Брейк-пойнты. Создание полностью адаптивной веб-страницы.

### **Раздел 4. Интерактивность HTML**

Мультимедийный проект. Описание сюжетных событий. Анимация. Создание эффекта анимации с помощью последовательной смены изображений. Имитационные модели. Интерактивные проекты. Игры.

*Теоретическая часть:*

- Определение и принципы интерактивности. Обзор базовых технологий интерактивности. Взаимодействие, реакция. Юзабилити. Выбор инструментов. Изучение границ применения технологий.
- Знакомство с формами HTML. Изучение способов создания форм в стандарте HTML5. Возможности стилизации форм.
- Освоение форм HTML. Изучение способов создания форм в стандарте HTML5. Создание релевантных контенту проекта форм данных типов.
- Освоение форм HTML. Изучение способов создания форм в стандарте HTML5. Создание релевантных контенту проекта форм данных типов. Создание реакции на заполнение формы.
- Изучение значимых атрибутов тегов. Освоение форм раскрывающегося списка текстовой области. Создание релевантных контенту проекта форм.

*Практическая деятельность:*

- Подготовка индивидуального проекта к созданию интерактивных элементов.
- Создание формы, набора полей, полей указанных типов в индивидуальном проекте
- Создание полей указанных типов в индивидуальном проекте.

### **Раздел 5. Интерактивность CSS**

Знакомство с псевдоклассами для навигации. Применение псевдоклассов к элементам тестовой страницы: ссылкам, кнопкам, меню. Состояние ссылки. Реакция на действия пользователя.

*Теоретическая часть:*

- Изучение возможностей трансформации. Масштабирование интерактивных элементов веб-страницы. Подготовка к анимации эффектов
- Продолжение изучения возможностей трансформации интерактивных элементов веб-страницы. Функции поворота, наклона и перемещения. Связанные функции. Подготовка к анимации эффектов
- Освоение параметров анимации для плавных трансформаций интерактивных элементов веб-страницы. Способы применения анимации. Список стилей для анимации, длительность, скорость и задержка
- Применение трансформаций и анимации интерактивных элементов веб-страницы на веб-странице. Создание эффектов для кнопок, меню и комплексной фотогалереи. Слои. Вопросы наложения элементов. Вопросы адаптивности при модификации интерактивных элементов

*Практическая деятельность:*

- Разработка фотогалереи. Применение масштабирования в индивидуальном проекте.



- Разработка фотогалереи. Применение поворота, наклона и перемещения к элементам в индивидуальном проекте.
- Разработка фотогалереи. Применение анимации к элементам в индивидуальном проекте.
- Анимация панелей навигации и фотогалереи в индивидуальном проекте

## **Раздел 6. Основы языка программирования JavaScript**

Алгоритм. Понятие алгоритма как формального описания последовательности действий исполнителя, приводящих от исходных данных к конечному результату. Схематическая запись алгоритма. Использование геометрических фигур для схематической записи алгоритма. Создание блок-схем в свободном векторном редакторе.

Характеристики и основы применения языка программирования JavaScript. Концепция DHTML. Структура документа. Понятие объекта и узла. Родственные отношения элементов. Понятие объектной модели документа и изучение способов обращения к объектам. Чтение и изменение свойств объектов. Первая программа

### Линейные алгоритмы

Основные признаки линейного алгоритма. Схематическое описание линейного алгоритма. Геометрические примитивы, используемые для описания линейного алгоритма.

Программное управление исполнителем. Создание программ для перемещения исполнителя по экранному полю. Понятие поворота исполнителя в определенное направление. Прямой угол. Поворот исполнителя на прямой угол по часовой стрелке и против часовой стрелки.

Создание программ для рисования линий. Изменение цвета и толщины рисуемой линии. Особенности пунктирной линии. Написание программы для исполнителя, чтобы он оставлял пунктирную линию при перемещении по экранному полю.

Прямоугольник, квадрат — основные черты. Написание программ для движения исполнителя вдоль сторон квадрата, прямоугольника. Внесение изменений в программу рисования квадрата, если необходимо получить другой размер стороны квадрата.

Прерывание программы.

### Циклические алгоритмы

Многократное повторение команд как организация цикла. Особенности использования цикла в программе. Упрощение программы путём сокращения количества команд при переходе от линейных алгоритмов к циклическим.

Схематическая запись циклического алгоритма.

Типы циклических алгоритмов. Основные конструкции программной среды, используемые для написания программ исполнителем с применением циклов.

Конечный цикл. Сокращение программы для исполнителя, рисующего линии, квадраты, прямоугольники при использовании цикла. Программа

исполнителя для рисования нескольких однотипных геометрических фигур, например, нескольких квадратов из одной вершины, но с различным значением стороны.

Конструкции программной среды спрятаться/показаться. Выполнение программы исполнителем, не показанным на поле выполнения программы.

Написание и отладка программ с применением конструкции цикл в цикле.

Бесконечный цикл. Повторяющаяся смена внешности исполнителя для имитации движения персонажа. Использование бесконечного цикла для создания анимации.

Получение различного эффекта воспроизведения программы при изменении костюма исполнителя.

Параллелизм в программной среде

Использование нескольких исполнителей. Копирование программы одного исполнителя другим. Выполнение одинаковых программ разными исполнителями с использованием различных начальных условий. Параллельное выполнение однотипных действий. Принцип суперкомпьютерных технологий. Таймер для вычисления времени выполнения программы. Уменьшение показаний таймера при использовании параллельных вычислений.

Интерактивность программ. Возможность организации диалога между исполнителями. Операторы для слияния текстовых выражений.

Взаимодействие исполнителей путём касания друг друга или цвета. Использование сенсоров при взаимодействии исполнителей. Задержка выполнения программы.

Работа исполнителей в разных слоях изображения.

Ветвление в алгоритмах

Использование ветвления при написании программ. Короткая форма. Полная форма условного оператора. Конструкции ветвления для моделирования ситуации.

Цикл пока. Повторение команд исполнителя при выполнении определенного условия.

Последовательное выполнение фрагментов программы разными исполнителями

Типы исполнителей программной среды. Системы команд исполнителей. Различные системы команд для разных типов исполнителей.

Управление событиями. Передача сообщений исполнителям для выполнения определенной последовательности команд.

Передача управления между различными типами исполнителей.

*Теоретическая часть:*

- придумывать задачи для исполнителей программной среды;
- выделять ситуации, для описания которых можно использовать линейный алгоритм, алгоритм с ветвлениями, повторениями;
- определять эффективный способ решения поставленной задачи;

- находить параллельности в выполняемых действиях и программировать их с помощью нескольких исполнителей;

- планировать последовательность событий для заданного проекта.

- Изучение видов подключения сценариев JavaScript к HTML-документу. Применение ссылок, обработчиков событий, внутренних и внешних сценариев. Синтаксис

- Изучение основных типов переменных в сценариях JavaScript, объявления переменных и операции присваивания. Имена переменных. Типы данных. Числовой тип данных и операции с ним. Методы класса Math. Сокращенная запись арифметических операций. Случайные величины. Динамическое изменение контента

- Изучение методов классов String и Date в JavaScript, конкатенации и способов ее применения для динамического изменения контента веб-страницы. Отображение текущей даты на веб-странице. Динамическое формирование контента

- Изучение условного оператора, простых и составных логических выражений. Полное и неполное ветвление. Операции сравнения. Тернарный оператор. Применение ветвления для динамического формирования текстов на веб-странице.

- Изучение пользовательских функций, их синтаксиса и применения. Параметры и аргументы. Функции-обработчики событий. Применение функций для динамической обработки форм на веб-странице.

- Изучение массивов, методов класса Array в JavaScript, решение задач обработки данных. Объявление и заполнение массива. Обработка массива. Стековые операции. Создание слайдера.

- Понятие цикла, изучение циклов с условием, цикла со счетчиком, их применения при обработке массивов и для динамического формирования текстов на веб-странице. Прерывание циклов.

- Способы внедрения интерактивных элементов (форм, слайдера и других) с минимальным изменением существующей структуры сайта. Объединение сценариев в общий блок. Изучение технологии скрытия/отображения части информации на веб-странице

*Практическая деятельность:*

- составлять и отлаживать программный код;

- использовать конструкции программной среды для создания линейных, разветвленных и циклических алгоритмов;

- организовывать параллельные вычисления;

- организовывать последовательность событий программы, передачу управления от одних исполнителей другим.

- Построение DOM своего проекта

- Написание «разорванных» сценариев и сценариев во внешнем файле в индивидуальном проекте.

- Вывод случайно чередующегося контента на веб-страницу в индивидуальном проекте.
- Применение временных параметров в индивидуальном проекте.
- Отображение разного контента в зависимости от условий в индивидуальном проекте.
- Создание формы авторизации и формы с автоматическим вычислением результата в индивидуальном проекте
- Создание слайдера с анимацией переходов.
- Создание слайдера с неограниченным количеством изображений в индивидуальном проекте. Формирование полей форм из массива.
- Внедрение ранее созданных интерактивных элементов в индивидуальном проекте. Предъявление контента по запросу пользователя

## **Раздел 7. Практикум**

Объединение содержимого HTML, разметки CSS и сценариев JavaScript. Форматирование контента и применение стилей. Изучение технического задания и создание верстки поля игры. Освоение навыков составления и чтения технического задания.

### *Теоретическая часть:*

- создавать план появления событий для отражения определенной темы;
- выбирать иллюстративный материал из встроенной библиотеки;
- выбирать метод анимации для конкретной задачи;
- планировать последовательность событий для создания эффекта анимации по выбранному сценарию.

• Разработка математической модели игры. Первичная расстановка в игре. Применение массивов для описания объектов и их свойств.

• Понятие события и состояния. Моделирование состояний и переходов. Разработка классов CSS. Разработка алгоритмов обработки событий и основного цикла игры.

• Динамический вывод рассчитанных значений на веб-страницу. Принципы обработки результатов игры. Разработка вариантов окончания игры. Генерация случайного призового промокода.

• Выделение сценариев во внешний файл. Вопросы оптимизации кода программ. Внедрение веб-страницы в другую страницу в виде фрейма. Создание отдельного приложения из HTML-проекта. Освоение пакета `nw.js`.

### *Практическая деятельность:*

• использовать возможности программной среды для создания мультимедийных проектов;

• создавать имитационные модели, интерактивные проекты и игры средствами программной среды.

• Верстка макета своего варианта игры в индивидуальном проекте

• Разработка математической модели своего варианта игры в индивидуальном проекте.

- Практика по динамическому формированию HTML-кода. Разработка алгоритмов обработки событий игры и реакций игровых объектов на события.

- Разработка сценариев окончания игры и подсчета результатов игрока в индивидуальном проекте.

- Оптимизация сценариев игры. Внедрение файла игры на главную страницу индивидуального проекта. Создание десктопного приложения. Презентация проекта.

### **Список использованных источников**

Литература для педагогов:

1. Жемчужников Д.Г. Веб-дизайн. Уровень 1: учебное пособие/ Д.Г.Жемчужников. – 2 изд., стеретип. – М.БИНОМ. Лаборатория знаний, 2020 – 144с.;
2. Жемчужников Д.Г. Веб-дизайн. Уровень 2: учебное пособие/ Д.Г.Жемчужников. – 2 изд., стеретип. – М.БИНОМ. Лаборатория знаний, 2020 – 144с.;
3. Рындак В. Г., Дженжер В. О., Денисова Л. В. Проектная деятельность школьника в среде программирования. — Оренбург: Оренб. гос. ин-т. менеджмента, 2009.
4. «Программируем, учимся и играем»; «Проектная деятельность школьника»/Учебно-методическое пособие/ В.Г. Рындак, В.О. Джинжер, Л.В. Денисова.
5. Никсон Р. HTML5 и CSS3. Мастер-класс: Пер. с англ. – СПб.: БХВ-Петербург, 2024. – 464 с.: ил.

Литература для учащихся и их родителей:

1. Крылова Е. Г. К85 Python для детей, которые пока не программируют. — СПб.: БХВ-Петербург, 2023. — 208 с.: ил.
2. Уитни Д. Программирование для детей. Пять самых крутых игр на HTML и JavaScript. – СПб.: Питер, 2020 – 224 с.: ил – (Серия «Вы и ваш ребенок»)

\_\_\_\_\_/\_\_\_\_

Календарно - тематическое планирование  
Группа №1

№ п/п	Тема занятия	Колич ество часов	Дата		Форма текущего контроля и аттестации	Вид работы
			план	факт		
1.	Инструктаж по технике безопасности. Введение в программирование	2			Опрос	
2.	Знакомство с технологиями веб разработки	2			Опрос	Презентация, демонстрация
3.	Структура HTML-документа. Содержимое блока заголовков	2			Опрос	Презентация, демонстрация
4.	Форматирование текста. Физическое и логическое форматирование. Списки.	2			Опрос	Презентация, демонстрация
5.	Гиперссылки	2			Опрос	Презентация, демонстрация
6.	Изображения	2			Упражнения	Практическая работа
7.	Изображения. Видео	2			Опрос	Презентация, демонстрация
8.	Таблицы	2			Опрос	Презентация, демонстрация
9.	Цвета	2			Опрос	Практическая работа
10.	Разработка индивидуального проекта	2			Проект	Презентация, демонстрация
11.	Основы CSS	2			Опрос	Презентация, демонстрация
12.	Селекторы. Приоритеты стилей. Каскадность	2			Проект	Практическая работа
13.	Стилевые свойства текста	2			Опрос	Презентация, демонстрация
14.	Стилевые свойства графики	2			Опрос	Презентация, демонстрация
15.	Блоки-контейнеры. Блочная модель в CSS	2			Тест	Практическая работа
16.	Практикум. Верстка одностраничного сайта	2			Опрос	Презентация, демонстрация
17.	Разработка индивидуального проекта	2			Проект	Презентация, демонстрация

18.	Практикум. Продолжение верстки	2			Проект	Практическая работа
19.	Практикум. Адаптивная верстка	2			Опрос	Презентация, демонстрация
20.	Разработка индивидуального проекта	2			Опрос	Презентация, демонстрация
21.	Знакомство с технологиями интерактивности	2			Опрос	Практическая работа
22.	HTML: формы. Общая структура. Типы полей text, email, password, submit	2			Опрос	Презентация, демонстрация
23.	HTML: формы. Типы полей	2			Проект	Презентация, демонстрация
24.	HTML: формы. Типы полей. Пример реакции веб-страницы на изменение значения в форме	2			Опрос	Практическая работа
25.	HTML: формы. Атрибуты тега input. Теги select и textarea	2			Опрос	Презентация, демонстрация
26.	CSS: навигация. Псевдоклассы ссылок	2			Опрос	Презентация, демонстрация
27.	CSS: трансформация. Функция scale()	2			упражнения	Практическая работа
28.	CSS: трансформация. Функции rotate(), skew(), translate()	2			Проект	Презентация, демонстрация
29.	CSS: анимация. Свойство transition	2			Опрос	Презентация, демонстрация
30.	CSS: применение. Панель навигации и фотогалерея. Слои	2			Опрос	Практическая работа
31.	JavaScript: основные сведения. Объектная модель (DOM)	2			Проект	Презентация, демонстрация
32.	JavaScript. Внедрение в документ. Изменение текста объекта	2			Опрос	Презентация, демонстрация
33.	JavaScript. Внедрение в документ. Изменение текста объекта	2			Опрос	Практическая работа
34.	JavaScript. Переменные. Математические функции. Чередование контента	2			Опрос	Презентация, демонстрация
35.	JavaScript. Переменные. Математические функции. Чередование контента	2			Упражнения	Презентация, демонстрация
36.	JavaScript. Строки. Дата и время. Отображение даты на странице	2			Опрос	Практическая работа
37.	JavaScript. Строки. Дата и время. Отображение даты на странице	2			Опрос	Презентация, демонстрация
38.	JavaScript. Условный оператор. Изменение контента по дате	2			Опрос	Презентация, демонстрация
39.	JavaScript. Условный оператор. Изменение контента по дате	2			упражнения	Практическая работа
40.	JavaScript. Функции. Форма авторизации. Калькулятор	2			Опрос	Практическая работа
41.	JavaScript. Функции. Форма	2			Опрос	Практическая

	авторизации. Калькулятор					работа
42.	JavaScript. Массивы. Слайдер на веб-странице	2			Защита работы	Практическая работа
43.	JavaScript. Массивы. Слайдер на веб-странице	2			Опрос	Практическая работа
44.	JavaScript. Циклы. Динамическое формирование контента. Обработка форм на устройстве пользователя	2			Опрос	Практическая работа
45.	JavaScript. Циклы. Динамическое формирование контента. Обработка форм на устройстве пользователя	2			Опрос	Практическая работа
46.	JavaScript. Внедрение интерактивных элементов. Скрытие и отображение элементов на веб-странице	2			Опрос	Презентация, демонстрация
47.	JavaScript. Внедрение интерактивных элементов. Скрытие и отображение элементов на веб-странице	2			Опрос	Презентация, демонстрация
48.	Практикум. Игра. Техническое задание. Верстка	2			упражнения	Практическая работа
49.	Практикум. Игра. Техническое задание. Верстка	2			Опрос	Практическая работа
50.	Практикум. Игра. Техническое задание. Верстка	2			Опрос	Практическая работа
51.	Практикум. Игра. Блок вычислений	2			проект	Практическая работа
52.	Практикум. Игра. Блок вычислений	2			Опрос	Презентация, демонстрация
53.	Практикум. Игра. События	2			Опрос	Презентация, демонстрация
54.	Практикум. Игра. События	2			упражнения	Практическая работа
55.	Практикум. Игра. Подсчет результатов. Окончание	2			Опрос	Презентация, демонстрация
56.	Практикум. Игра. Подсчет результатов. Окончание	2			Опрос	Презентация, демонстрация
57.	Практикум. Игра. Подсчет результатов. Окончание	2			Опрос	Практическая работа
58.	Практикум. Игра. Оптимизация программы. Внедрение на страницу. Создание desktop-приложения.	2			Опрос	Презентация, демонстрация
59.	Практикум. Игра. Оптимизация программы. Внедрение на страницу. Создание desktop-приложения.	2			Опрос	Презентация, демонстрация
60.	Практикум. Игра. Оптимизация программы. Внедрение на страницу. Создание desktop-приложения.	2			упражнения	Практическая работа
61.	Демонстрация разработанной игры	2			Защита проектов	Презентация, демонстрация
62.	Демонстрация разработанной игры	2			Защита проектов	Практическая работа



63.	Разработка индивидуального проекта	2			Проект	Практическая работа
64.	Разработка индивидуального проекта	2			Проект	Презентация, демонстрация
65.	Разработка индивидуального проекта	2			Проект	Практическая работа
66.	Безопасность в сети Интернет	2			Опрос	Практическая работа
67.	Обобщение и повторение изученного	2			Опрос	Презентация, демонстрация
68.	Обобщение и повторение изученного	2			Опрос	Практическая работа