

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

по выполнению внеаудиторной работы при реализации программы дополнительного
общеобразовательного образования

**«Подготовка к аттестации за 7 класс для дистанционного семейного
обучения»**

ВВЕДЕНИЕ

Настоящие методические рекомендации разработаны для обучающихся 7-х классов, углубляющихся в изучение информатики. В 7 классе акцент смещается на практическую веб-разработку, освоение основ программирования на JavaScript, работу с мультимедиа (видеомонтаж) и глубокое погружение в мир искусственного интеллекта. Документ призван помочь обучающемуся не только успешно освоить эти сложные темы, но и развить навыки самостоятельного создания цифровых продуктов, критического анализа информации и этичного использования современных технологий.

Материал структурирован таким образом, чтобы обучающийся мог пройти путь от понимания принципов кодирования информации до создания интерактивных веб-страниц, монтажа видеороликов и использования ИИ для решения творческих и практических задач.

Цель документа

Основная цель рекомендаций обеспечить глубокое понимание теоретических основ информатики, сформировать устойчивые практические навыки в области веб-разработки, мультимедиа и программирования, а также развить компетенции в области искусственного интеллекта и цифровой безопасности.

В рамках реализации данной цели предполагается решение следующих задач:

1. Разъяснить принципы кодирования информации, работы естественных и формальных языков.
2. Сформировать навыки создания статических и интерактивных веб-страниц с использованием HTML, CSS и JavaScript.
3. Обучить основам видеомонтажа в CapCut, включая работу с текстом, масками, анимацией, эффектами и звуком.
4. Развить понимание принципов работы искусственного интеллекта, научить эффективному промптингу и использованию ИИ для генерации контента (изображений, видео, веб-сайтов).
5. Обеспечить освоение базовых принципов 3D-моделирования, работы с материалами, светом и рендерингом.

6. Повысить уровень самостоятельности при выполнении проектных работ, включая создание биографических сайтов, социальных роликов и сайтов-портфолио с помощью ИИ.

Структура документа

Рекомендации разделены на ключевые блоки, соответствующие основным видам учебной деятельности в 7 классе:

1. Контроль знаний (тестирование).
2. Технологический практикум: Веб-разработка, Мультимедиа и 3D-моделирование: работа с HTML/CSS, CapCut, Adele Design и 3D-редакторами.
3. Алгоритмический блок: Программирование на JavaScript: освоение основ JS, создание интерактивных элементов.
4. Искусственный интеллект и Цифровое творчество: применение ИИ для генерации контента, автоматизации и создания проектов

Каждый раздел содержит:

1. Пошаговые алгоритмы действий («чек-листы»).
2. Типичные сценарии выполнения заданий с разбором нюансов.
3. Систему критериев для самопроверки.

Методологические основы

Рекомендации базируются на следующих принципах:

1. Проектно-ориентированный подход — обучение через создание реальных цифровых продуктов (сайты, видео, 3D-модели).
2. Междисциплинарность — интеграция знаний из области дизайна, программирования, логики и этики.
3. Актуальность технологий — использование современных инструментов и платформ (HTML/CSS/JS, CapCut, ИИ-сервисы)..
4. Развитие критического мышления — формирование навыков анализа информации, оценки достоверности и этичности использования ИИ.

5. Самостоятельность и инициатива — поощрение экспериментов, поиска решений и творческого подхода к задачам.

Целевая аудитория

Методические рекомендации предназначены для обучающихся 7 класса и направлены на поддержку их учебной деятельности по информатике.

Ожидаемые результаты реализации

Использование данных рекомендаций позволит:

1. Уверенно ориентироваться в теоретических основах информатики, включая кодирование информации и языки;
2. Создавать и оформлять веб-страницы с использованием HTML и CSS, добавлять интерактивность с помощью JavaScript;
3. Выполнять полноценный видеомонтаж в CapCut, включая работу с текстом, масками, анимацией, эффектами и звуком;
4. Эффективно использовать инструменты искусственного интеллекта для генерации изображений, видео, текстов и веб-сайтов;
5. Получить базовые навыки 3D-моделирования, работы с материалами, светом и рендерингом;
6. Развивать навыки цифрового творчества, создавая собственные уникальные проекты;
7. Развивать самостоятельность в обучении;
8. Формировать ответственное отношение к учебной деятельности;
9. Повышать уровень подготовки к текущему и итоговому контролю.

Особенности применения

Рекомендации могут применяться:

1. Как руководство при выполнении практических работ и проектов;
2. Во время выполнения домашних заданий;
3. При подготовке к диагностическим работам;

4. Для устранения пробелов в знаниях при самостоятельном изучении пропущенных тем.

Структура работы с документом

Для наиболее эффективного использования рекомендаций рекомендуется придерживаться следующего порядка работы:

1. Ознакомиться с теоретическим материалом по изучаемой теме (например, синтаксис HTML-тегов, принципы работы ИИ).
2. Открыть соответствующий раздел данных рекомендаций.
3. Выполнить подготовительные действия (установка ПО, подготовка медиафайлов, создание аккаунта в онлайн-сервисе).
4. Следовать алгоритму выполнения задания, обращая внимание на советы по оформлению и технические нюансы.
5. Провести самоаудит работы по критериям оценивания.
6. В случае возникновения ошибок — использовать методы отладки (для кода) или правки (для дизайна, видео).

Методические рекомендации являются практическим помощником в изучении информатики. Их использование способствует более осознанному освоению учебного материала, развитию учебной самостоятельности и формированию устойчивого интереса к предмету.

1. ТЕСТИРОВАНИЕ

Цели проведения тестирования

Тестирование в информатике 7 класса направлено на оценку глубокого понимания теоретических основ, а также способности применять знания в контексте современных технологий:

- Проверка понимания аппаратного и программного обеспечения, памяти компьютера и принципов работы компьютерных сетей;
- Оценка знаний по кодированию информации (текст, цвет, звук), естественным и формальным языкам;
- Контроль владения терминологией (HTML, CSS, JavaScript, промптинг, рендеринг, ИИ).
- Проверка умения анализировать информацию, включая критическую оценку контента, созданного ИИ;
- Формирование у обучающихся навыков самоконтроля и ответственности за результат.

Планируемые результаты

Обучающийся должен уметь:

- Идентифицировать компоненты компьютера и понимать их функции;
- Различать естественные и формальные языки, понимать принципы измерения и кодирования информации;
- Понимать базовые принципы работы HTML и CSS, структуру веб-страницы;
- Различать контент, созданный человеком, от контента, сгенерированного ИИ (изображения, видео, тексты);
- Понимать основные понятия искусственного интеллекта, промптинга и его применения;
- Различать контент, созданный человеком, от контента, сгенерированного ИИ (изображения, видео).

Методические рекомендации

Тесты по информатике 7 класса требуют не только знания определений, но и понимания взаимосвязей между различными компонентами и технологиями. При выполнении следует:

- анализировать визуальные данные, если в вопросе есть изображение окна программы, внимательно изучите активные кнопки и вкладки;
- стремиться «убирать лишнее» в вопросах про устройства или понятия часто можно отсеять заведомо неверные варианты, вспомнив основную функцию или определение.;
- стремиться «оптимизировать» ответ, облегчить содержание и конкретизировать ответ, особенно важно в заданиях с написанием кода;
- обращать особенное внимание к деталям, особенно это касается вопросов по системам счисления и алгоритмам, где важна каждая цифра или порядок действий.

Все задания основаны исключительно на изученном учебном материале, поэтому перед выполнением тестовой работы рекомендуется повторить основные понятия, определения, формулы и правила по соответствующей теме.

Временной регламент

Стандартный тест из 10-15 вопросов рассчитан на 25–30 минут. Практические задания внутри теста могут занимать больше времени.

Пример структуры теста

Общее количество заданий — 10, разделенных на три тематических модуля. Ниже представлен детальный разбор каждого блока с примерами вопросов.

Блок 1. Цифровая грамотность и Теоретические основы информатики (3-4 задания):

- Устройство компьютера: Функции памяти, процессора, устройств ввода/вывода (например, «Какой тип памяти используется для временного хранения данных при работе программ?»).

- Кодирование информации: Принципы кодирования текста, цвета (RGB, CMYK), звука (например, «Сколько бит необходимо для кодирования 256 различных цветов?»).
- Языки: Различие между естественными и формальными языками (например, «Приведите пример формального языка»).
- Задания про виды информации: Определение формы представления информации (например, «К какой форме представления информации относится звуковой файл?»).
- Измерение информации: Расчет информационного объема (например, «Какой объем памяти займет изображение размером 100x100 пикселей, если каждый пиксель кодируется 16 битами?»).

Блок 2. Веб-технологии и Информационные технологии (3 задания):

- HTML/CSS: Базовые теги, атрибуты, селекторы, свойства (например, «Какой HTML-тег используется для создания заголовка первого уровня?», «Какое CSS-свойство отвечает за цвет текста?»).
- Облачные сервисы: Продвинутое функции Яндекс Документов, Таблиц, Презентаций (например, «Как создать интерактивную диаграмму в Яндекс Таблицах?»).
- Графика и мультимедиа: Различие растровой и векторной графики, форматы изображений (PNG, SVG), основы видеомонтажа (например, «В чем основное отличие растровой графики от векторной?»).

Блок 3. Искусственный интеллект и Программирование (JavaScript) (3-4 задания):

- Искусственный интеллект: Понятия ИИ, машинного обучения, промптинга (например, «Как написать промпт для генерации изображений с ИИ?»).
- JavaScript: Переменные, типы данных, условные ветвления, функции (например, «Что выведет на экран следующий код JavaScript: `let x = 5; if (x > 3) {console.log("Больше");} else {console.log("Меньше");}`?»).

- Веб-разработка с ИИ: Применение ИИ для создания веб-сайтов (например, «Как ИИ может помочь в создании макета сайта?»).
- Отладка кода: Поиск ошибок в простых JavaScript-фрагментах (например, «Найдите ошибку в следующем коде: `let a = 10; let b = "5"; console.log(a + b)`).

Уровень	Описание	Пример задания
Базовый (40%)	Знание терминов, простых определений и базовых синтаксических конструкций.	Назовите основной тег для создания абзаца в HTML.
Повышенный (40%)	Применение знаний в знакомой ситуации, анализ фрагментов кода, решение задач по кодированию.	Опишите, как с помощью CSS изменить цвет текста на веб-странице.
Творческий (20%)	Анализ, синтез, решение проблем, критическая оценка, понимание сложных концепций.	Предложите, как можно использовать ИИ для автоматизации создания презентаций, и объясните преимущества такого подхода

2. ТЕХНОЛОГИЧЕСКИЙ ПРАКТИКУМ: ВЕБ-РАЗРАБОТКА, МУЛЬТИМЕДИА И 3D-МОДЕЛИРОВАНИЕ

Цели практических работ

Практическая деятельность в 7 классе становится более комплексной и творческой, охватывая создание полноценных цифровых продуктов. Она направлена на освоение инструментов для веб-разработки, видеомонтажа, графического дизайна и 3D-моделирования:

- Освоение базовых принципов и инструментов HTML для структурирования контента и CSS для его стилизации;
- Развитие навыков работы с Яндекс Документами, Таблицами, Презентациями на уровне, достаточном для создания сложных проектов;
- Формирование умений в веб-дизайне с использованием Adele Design, включая работу с типографикой, цветом и блочной моделью;
- Получение практического опыта в видеомонтаже с помощью CapCut, включая работу с текстом, масками, анимацией, эффектами, переходами и звуком.
- Освоение основ 3D-моделирования, работы с материалами, светом и рендерингом.

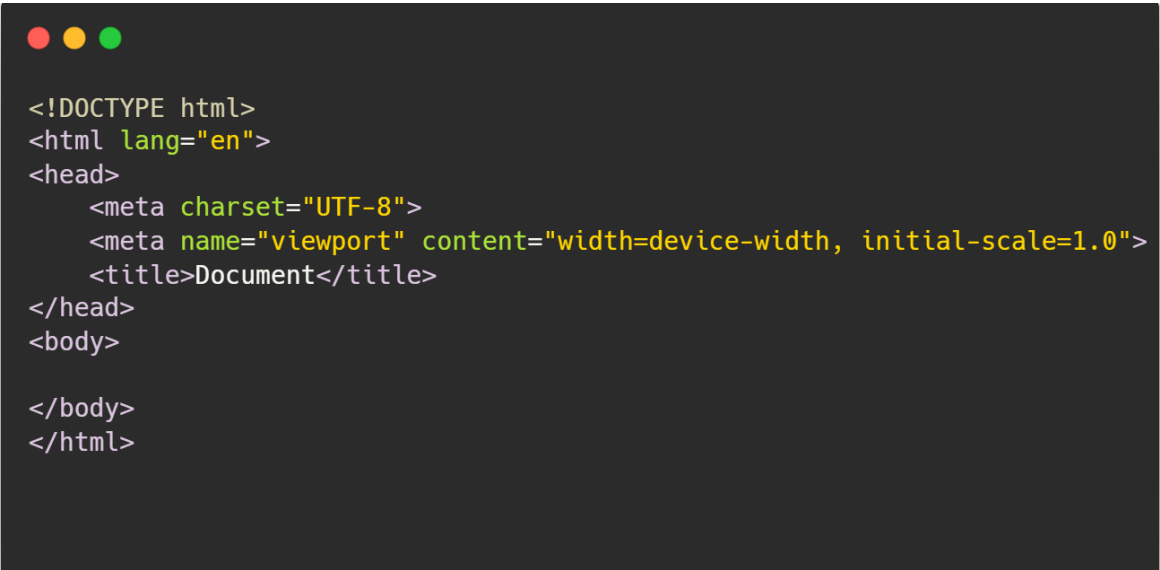
Планируемые результаты

- Веб-страницы: создание статических веб-страниц с использованием HTML-тегов (заголовки, абзацы, списки, таблицы, изображения, видео, гиперссылки) и стилизация их с помощью CSS (цвета, шрифты, отступы, блочная модель);
- Яндекс Документы: создание сложных документов, таблиц с формулами и диаграммами, интерактивных презентаций с мультимедиа;
- Веб-дизайн: разработка макетов сайтов (например, сайта-магазина) в Adele Design, понимание принципов UI/UX;
- Видеомонтаж: создание короткометражных фильмов или социальных роликов в CapCut, применение эффектов, переходов, цветокоррекции;
- 3D-моделирование: создание простых 3D-моделей (например, чашки), применение материалов, настройка освещения и рендеринг изображений.

Рекомендации по выполнению работ

- Прежде чем открывать редактор, набросайте структуру (какие будут заголовки, сколько слайдов, какие данные нужны для таблицы), например в блокноте или даже в физической тетради ручкой;
- Документы: Используйте встроенные стили для форматирования заголовков, создавайте оглавления. Активно применяйте инструменты правки и совместного доступа
- Таблицы: для сложных расчетов используйте формулы. Изучите функции СУММ, СРЗНАЧ, МИН, МАКС. Создавайте диаграммы для визуализации данных
- Презентации: Создавайте интерактивные презентации с гиперссылками между слайдами и на внешние ресурсы. Вставляйте видео и аудио, настраивайте анимацию и переходы
- Интерфейс: Освойте базовые инструменты Adele Design для создания форм, кнопок, блоков. Поймите, как работает блочная модель (margin, padding, border).
- Типографика и цвет: Выбирайте шрифты, которые легко читаются. Используйте контрастные цвета для текста и фона. Создайте гармоничную цветовую схему.
- Импорт и организация: загрузите все необходимые видеофрагменты, изображения, аудиофайлы. Разложите их по папкам для удобства.
- Базовый монтаж: Обрежьте лишнее, склейте фрагменты. Используйте переходы между сценами.
- Текст и анимация: Добавляйте заголовки, субтитры. Используйте маски для создания интересных эффектов. Анимлируйте объекты и текст.
- Цветокоррекция и экспорт: Настройте яркость, контрастность, насыщенность. Экспортируйте видео в нужном формате и разрешении.
- Планирование структуры: прежде чем писать код, нарисуйте на бумаге или в графическом редакторе, как будет выглядеть ваш сайт. Определите основные блоки (шапка, меню, контент, подвал).;
- HTML каркас: Начните с создания HTML-структуры. Используйте семантические теги (<header>, <nav>, <main>, <footer>, <section>, <article>) для лучшей

организации. Не забывайте про обязательные теги (<!DOCTYPE html>, <html>, <head>, <body>):



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>

</body>
</html>
```

- Соблюдайте правило «белого пространства» и не перегружайте работы содержанием.

- Гиперссылки и мультимедиа: убедитесь, что все ссылки ведут на правильные страницы или внешние ресурсы. Изображения и видео должны быть оптимизированы по размеру для быстрой загрузки страницы:

- Начните с простых 3D форм, постепенно усложняя их. Изучите основные инструменты для создания и редактирования объектов.

- CSS — стиль: после создания HTML-каркаса приступайте к стилизации. Используйте внешние таблицы стилей (<link rel="stylesheet" href="style.css">). Активно применяйте селекторы (по тегу, классу, ID) и свойства для управления цветом, шрифтами, размерами, отступами (margin, padding, border):

```

*,
*::before,
*::after {
  box-sizing: border-box;
  margin: 0;
  padding: 0;
}

html {
  scrollbar-gutter: stable;
}

body {
  font-family: var(--body);
  background: var(--bg);
  color: var(--text);
  line-height: 1.5;
  font-size: 14px;
  overflow-x: hidden;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
  scrollbar-gutter: stable;
}

```

- Моделирование: изучите интерфейс программы для 3D-моделирования. Начните с создания простых примитивов (куб, сфера, цилиндр).
- Материалы и свет: Применяйте различные материалы (пластик, металл, стекло) к объектам. Настройте источники света для создания реалистичных теней и отражений.
- Рендеринг: Визуализируйте вашу 3D-сцену, чтобы получить финальное изображение. Экспериментируйте с настройками рендеринга.

<i>Критерии оценивания практических работ</i>	
Функциональность	корректно ли работает сайт? Правильно ли отображается видео? Соответствует ли 3D-модель заданию?

Техническая грамотность	соблюдены ли стандарты HTML/CSS? Правильно ли применены эффекты в CapCut? Корректно ли настроены материалы и свет в 3D-модели.
Дизайн и эстетика	привлекателен ли внешний вид сайта? Гармонично ли смонтировано видео? Эстетична ли 3D-модель?
Структурированность	логично ли организован код? Понятна ли структура проекта? Удобен ли интерфейс?
Оригинальность и творческий подход	есть ли в работе уникальные элементы, авторские решения?
Соблюдение регламента	сохранен ли проект в нужном формате? Соблюдены ли требования к именованию файлов

3. АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ (JAVASCRIPT)

В 7 классе мы переходим от блочного программирования к текстовому, осваивая JavaScript. Это язык, который делает веб-страницы интерактивными. Это фундамент для создания динамичных сайтов, игр и веб-приложений.

Цели и задачи раздела:

- изучение синтаксиса, переменных, типов данных, операторов и базовых конструкций;
- освоение операторов if, else if, else для создания логики поведения программы;
- понимание принципов модульности кода, создание и вызов собственных функций;
- работа с пользовательским вводом (prompt) и выводом информации (alert, console.log);
- развитие навыка поиска и исправления ошибок в JavaScript-коде с использованием инструментов разработчика браузера;
- создание динамических элементов (кнопки, формы, меню, модальные окна) с помощью JavaScript.

Основные концепции JavaScript:

Для успешного освоения JavaScript вы должны уверенно ориентироваться в следующих концепциях:

Переменные: используются для хранения данных. Объявляются с помощью let, const или var: `let userName = "Ризван";`

Типы данных: понимать разницу между строками ("текст"), числами (123), булевыми значениями (true/false).

Операторы: Арифметические (+, -, *, /), сравнения (==, ===, >, <), логические (&&, ||, !).

Ввод-вывод: `alert("Сообщение");` — для вывода информации.

Условное ветвление: `if (условие) { // код } else { // код }` — для выполнения разных действий в зависимости от условия.

Функции: Блоки кода, которые можно вызывать по имени. Позволяют избежать дублирования кода и делают его более организованным.

Методические рекомендации

Определите, какие элементы на вашем сайте должны быть интерактивными (кнопки, формы, меню, модальные окна). Продумайте логику их работы:

Подключение JavaScript.

Всегда подключайте JavaScript-файл перед закрывающим тегом `</body>` для лучшей производительности. Используйте `<script src="script.js"></script>`.

Работа с DOM (Document Object Model)

JavaScript взаимодействует с HTML-элементами через DOM. Научитесь получать доступ к элементам по их ID или классу.

- `document.getElementById("myButton");`
- `document.querySelector(".myClass").`

Обработка событий

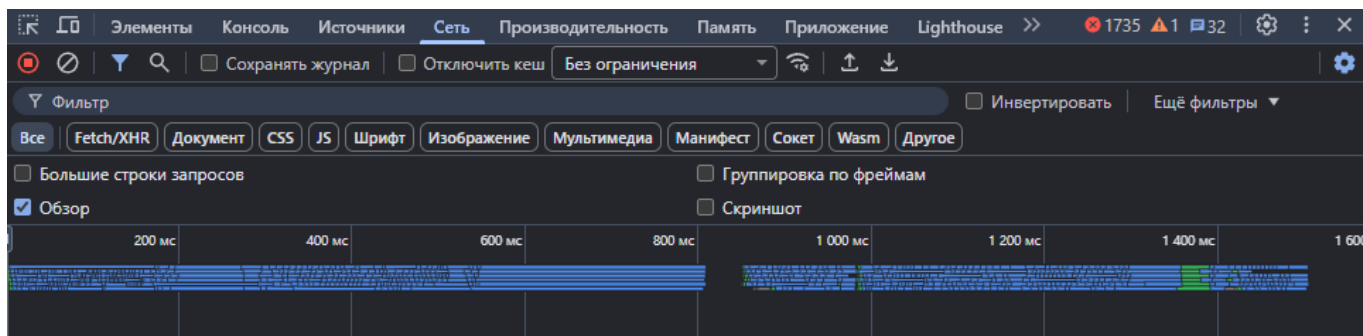
Используйте обработчики событий для реагирования на действия пользователя (клики, наведение мыши, ввод текста):

- `myButton.addEventListener("click", function() { // код при клике }).`

Отладка кода (Debugging)

Если ваш JavaScript-код не работает, как ожидалось:

- Консоль разработчика (F12): Самый главный инструмент. Ищите ошибки в разделе «Console». Сообщения об ошибках укажут на строку и тип проблемы;
- `console.log()`: Используйте эту команду для вывода значений переменных или сообщений в консоль, чтобы отслеживать ход выполнения программы.
- Точки останова (Breakpoints): В инструментах разработчика можно установить точки останова, чтобы пошагово выполнять код и видеть изменения значений переменных:



<i>Критерии оценки JavaScript-проекта</i>	
Работоспособность:	все интерактивные элементы функционируют согласно заданию.
Корректность кода	синтаксис JavaScript соблюден, нет явных ошибок
Логика	программа выполняет действия в правильной последовательности, условия и функции работают корректно
Читаемость кода	код хорошо структурирован, используются осмысленные имена переменных и функций, есть комментарии
Интерактивность	сайт реагирует на действия пользователя, предоставляет обратную связь
Отладка	продемонстрировано умение находить и исправлять ошибки в коде

4. ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И ЦИФРОВОЕ ТВОРЧЕСТВО

В 7 классе мы не только изучаем, что такое ИИ, но и активно используем его как мощный инструмент для творчества, автоматизации и решения задач. Это открывает новые горизонты в веб-разработке, генерации контента и создании интерактивных помощников.

Цели и задачи раздела

- Понимание ИИ: освоение базовых понятий искусственного интеллекта, его видов и областей применения;
- Эффективный промптинг: развитие навыков составления точных и детализированных запросов (промптов) для ИИ-моделей;
- Генерация контента: практическое применение ИИ для создания изображений, видео, текстов и даже веб-сайтов;
- Автоматизация с ИИ: знакомство с возможностями ИИ в автоматизации рутинных задач;
- ИИ в робототехнике: понимание роли ИИ в управлении роботами и создании интерактивных помощников;
- Цифровое творчество: использование ИИ как инструмента для реализации творческих идей в веб-дизайне, создании презентаций и других проектах.

Рекомендации по работе с ИИ

Качество результата от ИИ напрямую зависит от качества вашего запроса (промпта). Эффективный промпт должен быть:

- Четким и конкретным: избегайте двусмысленности;
- Детализированным: укажите стиль, настроение, цвета, объекты, ракурс, освещение (для изображений);
- Контекстным: объясните ИИ, для чего нужен результат (например, «Создай рекламный баннер для магазина электроники в стиле минимализм»);
- Итеративным: если первый результат не идеален, изменяйте промпт, добавляйте или убирайте детали, пробуйте разные формулировки.

Генерация изображений и видео с помощью ИИ

- Выбор инструмента: Используйте доступные ИИ-сервисы для генерации изображений;
- Промптинг: Экспериментируйте с различными промтами, чтобы получить желаемый результат. Изучайте примеры успешных промтов;
- Редактирование: Часто сгенерированные ИИ изображения требуют доработки в графических редакторах (например, Adobe Design) для финальной полировки.

Веб-разработка с помощью ИИ

- Идея и структура: Опишите ИИ, какой сайт вы хотите создать (тематика, основные разделы, функционал);
- Генерация макета/кода: ИИ может сгенерировать HTML/CSS-код или даже предложить готовый макет. Используйте это как отправную точку.
- Доработка: всегда проверяйте и дорабатывайте сгенерированный код.

<i>Критерии оценивания проектов с ИИ</i>	
Эффективность промпта:	насколько точно и качественно ИИ сгенерировал контент по вашему запросу?
Креативность	насколько оригинально и интересно вы использовали ИИ для реализации своей идеи?
Доработка	насколько качественно вы доработали сгенерированный ИИ контент (изображения, код, текст)?
Понимание	продемонстрировано ли понимание принципов работы ИИ и его ограничений?
Интерактивность	сайт реагирует на действия пользователя, предоставляет обратную связь
Этика	соблюдены ли этические нормы при использовании ИИ (например, указание авторства, отсутствие плагиата)?

ЗАКЛЮЧЕНИЕ

1. Постоянное обучение: Мир IT меняется очень быстро. Регулярно изучайте новые технологии, языки программирования, инструменты и тренды в веб-разработке, ИИ и мультимедиа.

2. Практика — это ключ к успеху: Теория важна, но без постоянной практики в кодировании, дизайне, видеомонтаже и промптинге вы не сможете освоить материал. Создавайте собственные проекты!

3. Дебаггинг ваш лучший друг: Ошибки в коде или дизайне это нормально. Умение находить, анализировать и исправлять их (дебаггинг) — один из самых ценных навыков программиста и разработчика.

4. Сначала логика, потом дизайн: сначала добейтесь того, чтобы программа или таблица работала правильно, и только потом приступайте к украшению, подбору шрифтов и картинок.

5. Чистота и структура кода: Пишите аккуратный, читаемый HTML, CSS и JavaScript. Используйте комментарии, осмысленные имена переменных и функций. Это облегчит работу вам и другим разработчикам».

6. Здоровье прежде всего: Правильная осанка за компьютером — залог отсутствия усталости.

7. Умение искать ответы: если программа выдает ошибку или вы не знаете, как сделать действие, используйте поиск в интернете или встроенную справку. Умение находить решение — это главный навык айтишника.

8. Эффективный промптинг: научитесь четко и точно формулировать свои мысли для ИИ. Это навык будущего, который позволит вам максимально использовать потенциал искусственного интеллекта.

9. Этика общения: помните о правилах сетевого этикета (нетикета) при переписке в почте или работе в общих документах. Уважайте чужой труд и личное пространство.

10. Оптимизация ресурсов: при создании веб-сайтов и видеороликов думайте об оптимизации изображений, видео и кода для быстрой загрузки и эффективной работы.