ПРИЛОЖЕНИЕ 1

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ В.Ф. УТКИНА»

Кафедра «Радиотехнических систем»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

по дисциплине

«СЕТЕВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ»

Направление подготовки
11.05.01 «Радиоэлектронные системы и комплексы»

Направленность (профиль) подготовки
Радиоэлектронная борьба
Радионавигационные системы и комплексы
Радиоэлектронные системы передачи информации
Уровень подготовки
Специалитет

Программа подготовки
Специалитет

Квалификация выпускника – инженер

Форма обучения – очная

Рязань 2024

Оценочные материалы – это совокупность учебно-методических материалов (контрольных заданий, описаний форм и процедур), предназначенных для оценки качества освоения обучающимися данной дисциплины как части основной образовательной программы.

Цель – оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям основной образовательной программы в ходе проведения текущего контроля и промежуточной аттестации.

Основная задача – обеспечить оценку уровня сформированности общекультурных и профессиональных компетенций, приобретаемых обучающимся в соответствии с этими требованиями.

Контроль знаний проводится в форме текущего контроля и промежуточной аттестации.

Текущий контроль успеваемости проводится с целью определения степени усвоения учебного материала, своевременного выявления и устранения недостатков в подготовке обучающихся и принятия необходимых мер по совершенствованию методики преподавания учебной дисциплины (модуля), организации работы обучающихся в ходе учебных занятий и оказания им индивидуальной помощи.

К контролю текущей успеваемости относятся проверка знаний, умений и навыков, приобретённых обучающимися на лекционных, практических и лабораторных занятиях. При выполнении лабораторных работ применяется система оценки «зачтено – не зачтено». Количество лабораторных работ по каждому модулю определено графиком, утвержденным заведующим кафедрой.

На практических занятиях допускается использование либо системы «зачтено – не зачтено», либо рейтинговой системы оценки, при которой, например, правильно решенная задача оценивается определенным количеством баллов. При поэтапном выполнении учебного плана баллы суммируются. Положительным итогом выполнения программы является определенное количество набранных баллов.

Промежуточный контроль по дисциплине осуществляется проведением экзамена по дисциплине. Форма проведения экзамена – устный ответ по утвержденным экзаменационным билетам, сформулированным с учетом содержания учебной дисциплины. В процессе подготовки к устному ответу экзаменуемый может составить в письменном виде план ответа, включающий в себя определения, выводы формул, рисунки и т.п.

Паспорт оценочных материалов по дисциплине

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или её части)	Вид, метод, форма оценочного мероприятия
1	*2*	*3*	*4*
1	**Введение в язык HTML** Структура HTML документа; Теги заголовка; - Title - Base - Link - Meta - Script Тело документа тег BODY; Элементы форматирования и разметки HTML – документа; - Типы HTML элементов HTML-атрибуты; Глобальные атрибуты.	ПК-1	Практическое занятие, Лабораторная работа, Экзамен

	HTML текст Тэги оформления HTML текста		
2	**HTML-ссылки** - Ссылки на внешние ресурсы - гиперссылки - Метод доступа Путь; - Абсолютный путь - Отнсительный путь Якорь; Изображение – ссылка; Вставка изображений в HTML – документ; - Атрибуты для тега <img> Адрес изображения; Размеры изображения; Форматы графических файлов; - Формат JPEG - Формат GIF - Формат PNG - Формат APNG - Формат SVG - Формат BMP - Формат ICO Тег <map>; Тег <area>; Типы атрибутов (краткое описание); Шаги создания карты – изображения.	ПК-1	Практическое занятие, Лабораторная работа, Экзамен
3	**CSS (CASCADING STYLE SHEETS, КАСКАДНЫЕ ТАБЛИЦЫСТИЛЕЙ) ПРАВИЛА ВНЕДРЕНИЯ** Подключение CSS к HTML; - Встраивание (внедрение в тег) - Подключение (внедрение в документ) - Связывание - Импорт Способы добавления стилей к документу; - Вложение стилей - Ссылка на документ стилей - импорт стилей - Добавление стилей в строки	ПК-1	Практическое занятие, Экзамен

	web – страницы Селекторы; - Селектор по элементу - Селекторы по индентификатору - Селекторы по классу - Контекстный селектор - Группировка селекторов Псевдоэлементы и псевдоклассы; Приоритет использования стилей; Основные средства создания и поддержки интернет ресурсов.		
4	**Java Script** Литериалы; Переменные; Выражения; Размещение javascript-кода; - Javascript – код на странице - Javascript-код во внешнем файле Javascript обработчики событий Функции; - Параметры функций Ветвления в программе – оператор if; - Оператор if - Оператор switch Циклы; - Цикл for - Цикл while	ПК-1	Практические занятия, Лабораторные работы, Экзамен
5	Заключение	ПК-1	Экзамен

Перечень компетенций с указанием этапов их формирования

При освоении дисциплины формируются следующие компетенции: ПК-1.

Указанные компетенции формируются в соответствии со следующими этапами (на лекционных, практических и лабораторных занятиях и при самостоятельной работе):

формирование и развитие теоретических знаний, предусмотренных указанными компетенциями (лекционные занятия, самостоятельная работа студентов);

закрепление теоретических знаний, умений, предусмотренных компетенциями, в ходе работы во время лекционных, практических и лабораторных занятий и при самостоятельной работе, а так же в процессе сдачи экзамена.

Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Сформированность каждой компетенции в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

1. пороговый уровень является обязательным для всех обучающихся по завершении освоения дисциплины;
2. продвинутый уровень характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;
3. эталонный уровень характеризуется максимально возможной выраженностью компетенций и является важным качественным ориентиром для самосовершенствования.

При достаточном качестве освоения более 80% приведенных знаний, умений и навыков преподаватель оценивает освоение данной компетенции в рамках настоящей дисциплины на эталонном уровне, при освоении более 60% приведенных знаний, умений и навыков – на продвинутом, при освоении более 40% приведенных знаний умений и навыков – на пороговом уровне. При освоении менее 40% приведенных знаний, умений и навыков компетенция в рамках настоящей дисциплины считается неосвоенной. Условием получения положительной оценки (зачтено) является посещение лекционных занятий и выполнение самостоятельных расчетных заданий.

Уровень сформированности каждой компетенции на различных этапах ее формирования в процессе освоения данной дисциплины оценивается в ходе текущего контроля успеваемости.

Оценке сформированности в рамках данной дисциплины подлежат компетенции:

– ПК-1 – Способен моделировать, анализировать и верифицировать результаты моделирования разработанных принципиальных схем аналоговых блоков радиофотонных устройств

Принимается во внимание знания обучающимися общепрофессиональных и профессиональных компетенций предусмотренных бакалавриатом по направлению 11.03.01.

Сформированность уровня компетенций не ниже порогового является основанием для допуска обучающегося к промежуточной аттестации по данной дисциплине (экзамену).

Формой промежуточной аттестации по данной дисциплине является экзамен, оцениваемый по принятой в ФГБОУ ВО «РГРТУ» четырёхбальной системе: «отлично» и «хорошо», «удовлетворительно» и «неудовлетворительно».

Уровень освоения сформированности знаний, умений и навыков по дисциплине оценивается в форме шкалы оценивания:

Шкала оценивания	Критерии оценивания
«Отлично»	**Студент**: показывает глубокое и прочное усвоение материала; исчерпывающе, последовательно, грамотно и логически стройно излагает теоретический материал; правильно формулирует определения; умеет сделать выводы по излагаемому материалу; безупречно отвечает на вопросы билета и на

	дополнительные вопросы в рамках рабочей программы дисциплины; умеет правильно выполнять практические задания, предусмотренные программой.
«Хорошо»	**Студент** показывает достаточно полное знание материала; знание основных теоретических понятий; достаточно последовательно, грамотно и логически стройно излагает материал; умеет сделать достаточно обоснованные выводы по излагаемому материалу; отвечает на все вопросы билета; показывает умение правильно выполнять практические задания, предусмотренные программой, показывает способ исправления допущенных погрешностей при изложении материала.
«Удовлетворительно»	**Студент** показывает общее знание изучаемого материала, умеет строить ответ в соответствии со структурой излагаемого вопроса; знает основные понятия дисциплины и показывает общее владение понятийным аппаратом; уметь устранить допущенные погрешности в ответе на теоретические вопросы и/или при выполнении практических заданий под руководством преподавателя, либо (при неправильном выполнении практического задания) по указанию преподавателя выполнить другие практические задания того же раздела дисциплины.
«Неудовлетворительно»	**Студент** не выполняет практическое задание, не владеет понятийным аппаратом дисциплины, не знает значительную часть материала рабочей программы, допускает существенные ошибки при изложении материала; не умеет строить ответ в соответствии со структурой излагаемого вопроса; не умеет делать выводы по излагаемому материалу. **Студент** после начала экзамена отказывается его сдавать, нарушает правила сдачи экзамена (списывание, использование не разрешенных средств связи и т.п.).
«Зачтено»	**Студент** показывает общее знание изучаемого материала, умеет строить ответ в соответствии со структурой излагаемого вопроса; знает основные понятия дисциплины и показывает общее владение понятийным аппаратом; уметь устранить допущенные погрешности в ответе на теоретические вопросы и/или при выполнении практических заданий под руководством преподавателя, либо (при неправильном выполнении практического задания) по указанию преподавателя выполнить другие практические задания того же раздела дисциплины.
«Не зачтено»	**Студент** не выполняет практическое задание, не владеет понятийным аппаратом дисциплины, не знает значительную часть материала рабочей программы, допускает существенные ошибки при изложении материала; не умеет строить ответ в соответствии со структурой излагаемого вопроса; не умеет делать выводы по излагаемому материалу. **Студент** после начала зачёта отказывается его сдавать, нарушает правила сдачи экзамена (списывание, использование не разрешенных средств связи и т.п.).

Типовые контрольные задания или иные материалы

Вопросы для подготовки к экзамену

Блок «Вопросы»

---

1. Что такое HTML и почему его нельзя полностью отнести к языкам программирования?

- Ответ: HTML (HyperText Markup Language) — это гипертекстовый язык разметки, используемый для создания веб-страниц. Его нельзя отнести к языкам программирования, поскольку он не выполняет вычислений и не содержит логики программирования. HTML определяет структуру и оформление текста, изображений и других элементов на странице, указывая браузеру, как их отображать.

---

2. Что такое WWW (World Wide Web), и как HTML связан с ней?

- Ответ: WWW (Всемирная паутина) состоит из множества взаимосвязанных документов, которые представлены в виде веб-сайтов и веб-страниц. HTML используется для создания и разметки этих веб-страниц, что делает их доступными для отображения в браузерах.

---

3. Опишите структуру HTML-документа. Какие основные теги используются для разделения заголовка и тела документа?

- Ответ: Структура HTML-документа включает в себя два основных раздела: заголовок и тело. Заголовок обозначается тегами `<head></head>`, а тело документа тегами `<body></body>`. Вся структура оборачивается тегами `<html></html>`.

---

4. Что делает тег `<title>`, и где отображается его содержимое?

- Ответ: Тег `<title>` определяет название документа, которое отображается в заголовке окна браузера. Это обязательный элемент заголовка документа, и он помогает идентифицировать веб-страницу.

---

5. Какие теги используются для создания разделов и организации информации на веб-странице?

- Ответ: Для создания разделов и организации информации используются теги `<div>` и семантические теги, такие как `<section>`, `<article>`, `<header>`, `<footer>`, и `<nav>`. Тег `<div>` обычно применяется для группировки элементов и задания стилей.

---

6. Что такое DOM (Document Object Model), и как она представлена в HTML?

- Ответ: DOM (Document Object Model) представляет собой объектную модель документа, в которой каждый элемент HTML является узлом в дереве, начиная с корневого элемента `<html>`. Это модель, которая позволяет скриптам изменять структуру, стиль и содержание документа.

---

7. Объясните концепцию "родительских" и "дочерних" элементов в HTML. Приведите пример.

- Ответ: В HTML "родительский" элемент — это элемент, который содержит в себе другие

элементы, называемые "дочерними". Например, `<div><p>Текст</p></div>`, где `<div>` — это родитель, а `<p>` — его дочерний элемент.

---

8. Какая разница между одиночными и парными тегами в HTML? Приведите примеры.

- Ответ: Парные теги имеют открывающий и закрывающий теги, например `<p></p>`, тогда как одиночные теги не имеют закрывающего тега, например `<br>` или `<img>`.

---

9. Для чего используется тег `<meta>`, и какие задачи он помогает решить?

- Ответ: Тег `<meta>` определяет метаданные документа, такие как ключевые слова для поисковых систем, автор страницы, кодировку символов и другие параметры. Он используется для улучшения SEO и обеспечения правильного отображения страницы.

---

10. Как создать комментарий в HTML, и зачем они нужны при разработке веб-страниц?

- Ответ: Комментарии в HTML создаются с помощью `<!-- комментарий -->`. Они используются для того, чтобы объяснить код, сделать заметки для разработчиков, а также временно отключить части кода без их удаления.

---

11. Что такое атрибуты тегов? Приведите примеры атрибутов, которые могут использоваться с тегом `<body>`.

- Ответ: Атрибуты тегов уточняют свойства элемента и задаются в открывающем теге. Примеры атрибутов для тега `<body>`: `bgcolor` для задания цвета фона, `text` для цвета текста, `link` для цвета ссылок, `vlink` для цвета посещенных ссылок.

---

12. Опишите назначение тега `<base>`. В каком случае он используется?

- Ответ: Тег `<base>` указывает базовый URL для всех относительных ссылок в документе. Он используется, чтобы упростить управление ссылками на ресурсы, например, когда нужно определить адрес для всех изображений или документов.

---

13. Каким тегом создается гиперссылка, и какой атрибут определяет адрес перехода?

- Ответ: Гиперссылка создается с помощью тега `<a>`. Атрибут `href` указывает адрес перехода. Пример: `<a href="https://example.com">Ссылка</a>`.

---

14. Что делает атрибут `target="_blank"` при создании ссылки?

- Ответ: Атрибут `target="_blank"` открывает ссылку в новом окне или новой вкладке браузера.

---

15. Как сделать ссылку на адрес электронной почты в HTML? Приведите пример.

- Ответ: Для создания ссылки на адрес электронной почты используется атрибут `mailto:`. Пример: `<a href="mailto:example@mail.com">Отправить письмо</a>`.

---

16. Чем отличается абсолютный путь от относительного в ссылках? Приведите примеры.

- Ответ: Абсолютный путь указывает полный URL, включая доменное имя, например, `http://example.com/page.html`. Относительный путь указывает расположение файла относительно текущего документа, например, `../images/photo.jpg`.

---

17. Что такое якорь в HTML, и как с его помощью организовать внутреннюю навигацию по странице?

- Ответ: Якорь — это внутренняя ссылка, которая указывает на определенный раздел страницы. Для создания якоря используется атрибут `id` и ссылка с символом `#`. Пример: `<a href="#section1">Перейти к разделу 1</a>` и `<div id="section1">Раздел 1</div>`.

---

18. Какие теги используются для выделения текста жирным шрифтом и курсивом? В чем их отличие от `<strong>` и `<em>`?

- Ответ: Для выделения текста жирным шрифтом используется тег `<b>`, а для выделения курсивом — `<i>`. Теги `<strong>` и `<em>` также делают текст жирным и курсивным соответственно, но придают ему дополнительное семантическое значение. `<strong>` указывает на важность текста, а `<em>` — на акцент или выделение.

---

19. Как создать список с нумерацией в HTML? Какой тег используется для создания маркированного списка?

- Ответ: Для создания нумерованного списка используется тег `<ol>`, а для маркированного списка — тег `<ul>`. Элементы списка обозначаются тегом `<li>`. Пример: `<ol><li>Элемент 1</li><li>Элемент 2</li></ol>` и `<ul><li>Элемент 1</li><li>Элемент 2</li></ul>`.

---

20. Для чего используется тег `<blockquote>`, и чем он отличается от тега `<q>`?

- Ответ: Тег `<blockquote>` используется для выделения больших блоков цитат с отступами по бокам. Тег `<q>` применяется для вставки коротких цитат и отображается в тексте с кавычками. Пример использования `<blockquote>`: `<blockquote>Это большая цитата</blockquote>`.

---

21. Каким образом можно задать подчеркивание текста в HTML?

- Ответ: Для подчеркивания текста используется тег `<u>`. Пример: `<u>Подчеркнутый текст</u>`. Ранее подчеркивание применялось для указания ссылок, но сейчас это не рекомендуется, чтобы избежать путаницы с гиперссылками.

---

22. Что делает тег `<hr>`, и какие параметры можно использовать для его стилизации?

- Ответ: Тег `<hr>` создает горизонтальную линию, которая используется для разделения контента на странице. Его можно стилизовать с помощью атрибутов `width`, `size`, `align` и `noshade`. Например: `<hr width="50%" size="5" align="center" noshade>`.

---

23. Какие теги используются для создания таблицы в HTML? Опишите их предназначение.

- Ответ: Для создания таблицы используются следующие теги:
  - `<table>` — определяет таблицу.
  - `<tr>` — определяет строку в таблице.
  - `<td>` — определяет ячейку в строке.
  - `<th>` — определяет заголовок столбца (обычно отображается жирным и центрированным).

---

24. Как задать заголовок таблицы, и где он отображается на веб-странице?

- Ответ: Заголовок таблицы задается с помощью тега `<caption>`. Он отображается над таблицей и используется для описания ее содержимого. Пример: `<table><caption>Моя таблица</caption>...</table>`.

---

25. Чем отличаются теги `<thead>`, `<tbody>`, и `<tfoot>`?

- Ответ:
  - `<thead>` содержит строки заголовков таблицы.
  - `<tbody>` содержит основные строки данных таблицы.
  - `<tfoot>` используется для строк с итогами, которые отображаются внизу таблицы. Эти теги помогают структурировать таблицу и сделать ее более понятной.

---

26. Объясните, как создать вложенные списки в HTML.

- Ответ: Вложенные списки создаются путем добавления одного списка в другой. Пример:
  ```html
  <ul>
    <li>Элемент 1
      <ul>
        <li>Вложенный элемент 1.1</li>
        <li>Вложенный элемент 1.2</li>
      </ul>
    </li>
    <li>Элемент 2</li>
  </ul>
  ```.

---

27. Как вставить изображение на веб-страницу с помощью тега `<img>`, и какие атрибуты

необходимо указать?

- Ответ: Тег `<img>` используется для вставки изображения. Обязательный атрибут — `src`, который указывает путь к изображению. Дополнительно можно использовать атрибут `alt` для описания изображения. Пример: `<img src="image.jpg" alt="Описание изображения">`.

---

28. Для чего используется атрибут `alt` у тега `<img>`?

- Ответ: Атрибут `alt` используется для указания альтернативного текста, который отображается, если изображение не удалось загрузить. Он также важен для доступности, помогая людям с ограниченными возможностями понять содержание изображения.

---

29. Как задать альтернативные источники для видео и аудио элементов в HTML5?

- Ответ: В HTML5 альтернативные источники указываются с помощью нескольких тегов `<source>` внутри элементов `<video>` и `<audio>`. Это обеспечивает поддержку разных форматов медиафайлов. Пример:

```html
<video controls>
  <source src="video.mp4" type="video/mp4">
  <source src="video.ogg" type="video/ogg">
  Ваш браузер не поддерживает видео.
</video>
```.

---

30. Объясните, как работает тег `<map>` для создания карт-изображений.

- Ответ: Тег `<map>` используется для создания карты-изображения, с активными областями, на которые можно кликнуть для перехода по ссылкам. Активные области описываются с помощью тега `<area>`. Пример:

```html
<img src="image.jpg" usemap="#mapname">
<map name="mapname">
  <area shape="rect" coords="34,44,270,350" href="page1.html" alt="Область 1">
  <area shape="circle" coords="75,75,30" href="page2.html" alt="Область 2">
</map>
```.

---

31. Какие элементы HTML используются для создания форм ввода данных?

- Ответ: Для создания форм используются элементы:
  - `<form>` — определяет форму.
  - `<input>` — определяет поля ввода данных.
  - `<textarea>` — определяет многострочное текстовое поле.
  - `<button>` — создает кнопку.
  - `<select>` — создает выпадающий список.
  - `<label>` — связывает метку с элементом формы.

---

32. Объясните различия между тегами `<input>` с типами "text", "password", и "submit".

- Ответ:
  - `<input type="text">` создает текстовое поле для ввода обычного текста.
  - `<input type="password">` создает поле для ввода пароля, при котором символы скрыты.
  - `<input type="submit">` создает кнопку для отправки формы на сервер.

---

33. Что такое атрибут `placeholder` у тега `<input>`, и для чего он используется?

- Ответ: Атрибут `placeholder` задает текст-подсказку, который отображается внутри поля ввода до тех пор, пока пользователь не начнет вводить данные. Пример: `<input type="text" placeholder="Введите имя">`.

---

34. Какой тег используется для создания выпадающего списка, и как можно задать значения для этого списка?

- Ответ: Для создания выпадающего списка используется тег `<select>`, а его элементы задаются с помощью тегов `<option>`. Пример:
  ```html
  <select>
    <option value="1">Первый вариант</option>
    <option value="2">Второй вариант</option>
  </select>
  ```.

---

35. Для чего нужен тег `<label>`, и как связать его с элементом ввода данных?

- Ответ: Тег `<label>` улучшает доступность формы, связывая метку с элементом ввода. Для связи используется атрибут `for`, значение которого соответствует `id` элемента ввода. Пример: `<label for="username">Имя пользователя:</label><input type="text" id="username">`.

---

36. Какие существуют способы подключения CSS-стилей к HTML-документу?

- Ответ: Существует три основных способа подключения CSS:
  1. Встроенные стили (inline) — с помощью атрибута `style` внутри HTML-тега, например: `<p style="color: red;">Текст</p>`.
  2. Внутренние стили (internal) — внутри тега `<style>` в секции `<head>`, например:
     ```html
     <style>
       body { background-color: lightblue; }
     </style>
     ```
  3. Внешние стили (external) — подключение внешнего файла CSS с помощью тега `<link>`, например: `<link rel="stylesheet" href="styles.css">`.

---

37. Как подключить внешний файл JavaScript к HTML-документу?

- Ответ: Внешний файл JavaScript подключается с помощью тега `<script>` с атрибутом `src`, например:

```html
<script src="script.js"></script>
```

Этот тег обычно размещается внизу тела документа перед закрывающим тегом `</body>`, чтобы ускорить загрузку страницы.

---

38. Что такое атрибут `defer` у тега `<script>`, и как он влияет на загрузку скрипта?

- Ответ: Атрибут `defer` у тега `<script>` указывает, что скрипт должен загружаться параллельно с загрузкой страницы, но выполняться только после того, как HTML-документ будет полностью обработан. Это помогает ускорить загрузку страницы и гарантирует, что скрипты не замедлят отображение контента.

---

39. Какая разница между атрибутами `type` и `language` у тега `<script>`?

- Ответ:
  - `type` — указывает MIME-тип скрипта, например, `type="text/javascript"`. Этот атрибут используется в современной разработке.
  - `language` — устаревший атрибут, который определял язык скрипта, например, `language="JavaScript"`. Сейчас его использование не рекомендуется.

---

40. Как подключить несколько внешних таблиц стилей к одному HTML-документу?

- Ответ: Несколько таблиц стилей подключаются с помощью нескольких тегов `<link>` в секции `<head>`. Пример:

```html
<link rel="stylesheet" href="styles1.css">
<link rel="stylesheet" href="styles2.css">
```

---

41. Почему важно использовать семантические теги в HTML5? Приведите примеры таких тегов.

- Ответ: Семантические теги в HTML5 делают структуру документа более понятной как для разработчиков, так и для поисковых систем. Они помогают улучшить SEO и доступность сайта. Примеры семантических тегов: `<header>`, `<footer>`, `<article>`, `<section>`, `<nav>`, `<aside>`.

---

42. Как теги `<article>`, `<section>`, и `<aside>` помогают структурировать контент на странице?

- Ответ:
  - `<article>` используется для автономного содержимого, например, статей или блогов.
  - `<section>` организует контент в логические разделы.
  - `<aside>` содержит дополнительную информацию, например, боковую панель с дополнительным контентом, связанным с основной статьей.

---

43. Что такое SEO, и как элементы HTML могут повлиять на оптимизацию страницы для поисковых систем?

- Ответ: SEO (Search Engine Optimization) — это процесс оптимизации веб-страницы для повышения ее видимости в поисковых системах. Элементы HTML, такие как теги `<title>`, `<meta>`, `<h1>`, семантические теги и правильное использование атрибутов `alt`, помогают улучшить ранжирование страницы в поисковых системах.

---

44. Какие теги и атрибуты можно использовать для управления видимостью веб-страницы для поисковых роботов?

- Ответ: Для управления видимостью веб-страницы для поисковых роботов используются теги `<meta>` с атрибутами `name="robots"` и значениями `content="noindex, nofollow"`, чтобы запретить индексацию и переход по ссылкам.
Пример:
  ```html
  <meta name="robots" content="noindex, nofollow">
  ```.

---

45. Какой тег используется для указания связанного языка контента, и как его правильно применить?

- Ответ: Тег `<html>` может содержать атрибут `lang` для указания языка содержимого документа. Это помогает поисковым системам и программам чтения с экрана корректно отображать и интерпретировать текст. Пример: `<html lang="ru">` для русского языка.

---

46. Какой стандарт HTML является актуальным на сегодняшний день, и какие нововведения он внес?

- Ответ: Актуальным стандартом является HTML5. Он добавил поддержку мультимедиа (видео и аудио), новые семантические теги, улучшенные формы, поддержку локального хранения данных (localStorage), встроенные API, такие как геолокация, и многое другое.

---

47. В чем разница между HTML4 и HTML5? Приведите основные отличия.

- Ответ: Основные отличия между HTML4 и HTML5:
  - HTML5 предоставляет поддержку мультимедиа без необходимости использования сторонних плагинов.
  - HTML5 включает новые семантические элементы, такие как `<header>`, `<footer>`, `<article>`, `<section>`.
  - В HTML5 улучшена работа с формами (новые типы ввода, атрибуты валидации) и добавлены API для взаимодействия с веб-приложениями.

---

48. Объясните, зачем используются теги `<header>` и `<footer>`, и как они влияют на структуру

страницы.

- Ответ: Теги `<header>` и `<footer>` помогают структурировать контент страницы. `<header>` используется для обозначения заголовка или вступления, содержащего логотип, навигацию или заголовок документа. `<footer>` содержит нижнюю часть страницы, например, контактную информацию, ссылки на политику конфиденциальности и авторские права.

---

49. Что означает тег `<nav>`, и когда его следует использовать?

- Ответ: Тег `<nav>` обозначает раздел страницы, предназначенный для навигационных ссылок. Его следует использовать для группировки основных навигационных элементов, таких как меню или ссылки на разделы сайта. Это улучшает семантическую структуру документа.

---

50. Какой атрибут позволяет автоматически обновлять страницу через определенное время, и как это делается?

- Ответ: Атрибут `<meta http-equiv="refresh">` позволяет автоматически обновлять страницу через заданный интервал времени. Пример:

```html
<meta http-equiv="refresh" content="30">
```

В этом примере страница будет обновляться каждые 30 секунд.

Блок «Задачи»

Задача 1 (таблица, секторы)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Создайте таблицу, состоящую из 5 строк и 5 столбцов, укажите таблице класс **table**
7. Первую строку оберните в тег **thead** и столбцы переделайте в заглавные столбцы **(th)**
8. Последнюю строку оберните в тег **tfoot**
9. Оставшиеся строки оберните в тег **tbody**
10. Обозначьте столбцы первой строки заголовками, прописав текст (*первый - Предмет, второй - Кабинет, третий - Время, четвертый - Корпус, пятый – Время*) в тегах **th**
11. Заполните строки, которые находятся в **tbody** любыми произвольными данными
12. В последней строке, которая находится в **tfoot**, объедините все столбцы в один, используя соответствующий атрибут для **td**
13. В получившимся столбце напишите текст "Всего: 3"
14. В CSS файле создайте ***селектор по классу*** **td,th** и создайте сплошную границу толщиной в 1 пиксель и каким-либо цветом; и задайте внутренний отступ сверху и снизу по 5px, слева и справа по 10px
15. Создайте ***селектор по элементу*** **table** и объедините границы ближайших столбцов при помощи специального свойства, так же задайте таблице ширину 500px;
16. Создайте ***селектор по ID*** **thead** и **tfoot**, измените цвет фона

CSS	HTML

table
{
width: 500px;
border-collapse: collapse;
}
th.m1
{
border: 1px solid green;
padding: 5px 10px;
color:brown;
background-color:blue;
}
td.m2
{
border: 1px solid green;
padding: 5px 10px;
color:brown;
background-color: pink;
}
thead#k1
{
background-color:blue;
}
tfoot#k2
{
background-color:red;
color:pink;
text-align: right;
}
	<HTML>
	<HEAD>
	<TITLE>Группа </TITLE>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</HEAD>
	<body>
	<table>
	<thead id="k1">br>
	<tr >
	<th class="m1">№</th>
	<th class="m1">Предмет</th>
	<th class="m1">Кабинет</th>
	<th class="m1">Корпус</th>
	<th class="m1">Время</th>
	</tr>
	</thead>
	<tbody>
	<tr>
	<td class="m2">1</td>
	<td class="m2">Математика</td>
	<td class="m2">448</td>
	<td class="m2">Центральный</td>
	<td class="m2">8:10-9:45</td>
	</tr>
	<tr >
	<td class="m2">2</td>
	<td class="m2">Физика</td>
	<td class="m2">350</td>
	<td class="m2">Центральный</td>
	<td class="m2">9:55-11:30</td>
	</tr>
	<tr>
	<td class="m2">3</td>
	<td class="m2">Сетевые информационные технологии</td>
	<td class="m2">502</td>
	<td class="m2">Лабораторный</td>
	<td class="m2">11:40-13:15</td>
	</tr>
	</tbody>
	<tfoot id="k2">
	<tr>
	<td colspan="5">Всего: 3</td>
	</tr>
	</tfoot>
	</table>
	</BODY>
	</HTML>

Задача 2 (Селекторы)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа состоящее из вашей фамилии и 3 нумерованных списков (*Одежда, Мебель, Еда*), по 3 пункта в каждом.
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Установите цвет всех пунктов с помощью селектора **по элементу**
7. Установите цвет названий списков через **селектор по ID**
8. Установите цвет всех пунктов *Одежда* с помощью **селектора по классу**
9. Установите цвет *фамилии* и всех пунктов *Мебель* с помощью **селектора по классу** и группируйте их, минимизируя код
10. Цвета не должны повторяться

CSS	**HTML**

<style>
li
{
color:red;
}
p#k1
{
color:green;
}
p#k2
{
color:blue;
}
p#k3
{
color:brown;
}
li.m1
{
color:yellow;
}
p,li.m2
{
color:pink;
}
</style>
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	 Иванов
	 Одежда
	
	<li class="m1"> …
	<li class="m1"> …
	<li class="m1"> …
	
	 Мебель
	
	<li class="m2"> …
	<li class="m2"> …
	<li class="m2"> …
	
	 Еда
	
	  …
	  …
	  …
	
	</body>
	</html>

Задача 3 (Фон, границы)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа состоящее из текста с параграфами
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Установить фоновое изображение страницы.
7. Сделайте так, чтобы фоновое изображение повторялось только по вертикали.
8. Используйте свойство background, чтобы фоновое изображение **НЕ прокручивалось** вместе с остальной частью страницы.
9. Сделайте так, чтобы фоновое изображение было по центру и на всю страницу.
10. Установите границы параграфов сплошными, а затем сделайте левую пунктирной и скруглите углы границ параграфов
11. Используйте свойство margin, чтобы выровнять элемент <h1> по центру, а элементу <h2> задать параметры:

left: 10px
right: 15px
top: 20px
bottom: 25px

CSS	**HTML**

body
{
background-image: url("https://... ");
background-repeat:repeat-y;
background-attachment:fixed;
background-size: cover;
background-position: center;
}
p
{
border:4px dotted red;
border-left-style: dotted;
border-radius: 5px;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	
	…….
	
	</body>
	</html>

Задача 4 (Поля, отступ, высота, ширина)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа, состоящего из текса с 3 заголовками и параграфами
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Используйте свойство margin, чтобы выровнять элемент <h1> по центру, а элементу <h2> задать параметры:

left: 10px
right: 15px
top: 20px
bottom: 25px

7. Установите верхний отступ элемента h3 20px, а элементу p отступы со всех сторон 20px
8. Установите высоту элемента <h1> 100px, а ширину 30%

CSS	HTML

h1
{
height: 100px;
width: 30%;
margin: 0 auto;
}
h2
{
margin:10px 15px 20px 25px;
}
h3
{
padding: 20px
}
p
{
padding-top:20px;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	 …
	 …
	 …
	
	…….
	
	…….
	</body>
	</html>

Задача 5 (Бокс модель, селекторы, изображение)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа состоящее из элемента и изображения.
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Установите ширину одного элемента на "300 пикселей". Добавьте сплошную красную рамку размером 2 пикселя к элементу . Добавьте пробел в 25 пикселей снаружи, слева от элемента .
7. Установите изображению параметры:

left: 10px
right: 15px
top: 20px
bottom: 25px

CSS	HTML

div#d1
{
width: 300px;
border: 2px solid red;
padding: 25px;
margin-left: 25px;
}
IMG#m1
{
float: left;
margin: 10px 15px 20px 25px;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	
	…
	<IMG id="m1" SRC="1.jpg"… class="left" >
	…
	
	</body>
	</html>

Задача 6 (Текст, ссылки, шрифт)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа состоящее из 3 ссылок и теста с заголовком и параграфами
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Выровняйте элементы и <h1> по центру и измените цвет текста
7. Сделайте отступ в первой строке элемента на 20 пикселей.
8. Оформите текст в <h1> с заглавными буквами, а текст в с заглавной буквы
9. Измените шрифт элементов на жирный, а <h1> на «italic»

CSS	**HTML**

p
{
color: red;
text-indent: 20px;
text-align: center;
text-transform: capitalize;
font-weight: bold;
}
h1
{
text-align: center;
text-transform: uppercase;
font-style: italic;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	 …
	
	…….
	
	</body>
	</html>

Задача 7 (Ссылки, псевдоклассы)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа состоящее из 3 ссылок
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Измените цвет ссылок.
7. Установите цвет при наведении курсора мыши на ссылку
8. Установите цвет ссылок, которые вы посетили
9. Удалите подчеркивание из ссылок, но добавите его при наведении курсора мыши на ссылку

CSS	**HTML**

//ссылка при наведении курсора мыши //
A:hover
{
color: red;
text-decoration: underline;
}
//посещённая ссылка//
A:visited
{
color: blue;
text-decoration: none;
}
//не посещённая ссылка//
A:link
{
color: green;
text-decoration: none;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	<A HREF="doc1.html">… </A>
	<A HREF="doc2.html">… </A>
	<A HREF="doc3.html">… </A>
	</body>
	</html>

Задача 8 (Переполнение, полоса прокрутки)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа с элементом div большим по содержанию
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Добавьте полосу прокрутки.
7. Задайте ширину div в 40%
8. Добавьте свойство max-height в CSS, чтобы скроллер появлялся после 200 px
9. Сделайте внутренний отступ с помощью свойства padding в 1%;
10. Установите для div выравнивание по левому краю

CSS	**HTML**

div
{
width:40%;
height:200PX;
FLOAT: left;
padding: 1%;
overflow:scroll;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	 …
	</body>
	</html>

Задача 9 (Списки)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа с 2 ненумерованными списками и 1 нумерованным
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Установите стиль списка square
7. У первого ненумерованного списка замените маркеры на изображение, у второго удалите их, а у нумерованного замените их на римские цифры

CSS	**HTML**

ul#u1
{
list-style-type: square;
list-style-image: url("bling.jpg");
}
ul
{
list-style-type: none;
}
ol
{
list-style-type: upper-roman;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	<ul id="u1">
	  …
	  …
	
	  …
	  …
	
	  …
	  …
	
	</body>
	</html>

Задача 10 (Таблицы)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа с таблицей. В столбцах укажите предмет и аудиторию, заполните их на 3 строки, в последней строке укажите общее количество занятий
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Установите сплошную границу таблицы зелёного цвета в 2 пикселя
7. Соедините границы таблицы в единую рамку
8. Установите ширину таблицы 70%
9. Выровняйте текст в <td> справа
10. Установите отступ в 15 пикселей у <th> элемента
11. У <th>, <tfoot> и <td> элементов установите цвет фона
12. В последней строке, объедините все столбцы в один, используя соответствующий атрибут для td

CSS	HTML

table, th, td, tfoot
{
border: 2px solid green;
border-collapse: collapse;
}
td
{
text-align: right;
background-color: red;
}
th
{
padding: 15px;
background-color: blue;
}
table
{
width:70%
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	<table>
	<tr>
	<th>…</th>
	<th>…</th>
	</tr>
	<tr>
	<td>…</td>
	<td>…</td>
	</tr>
	<tr>
	<td>…</td>
	<td>…</td>
	</tr>
	<tfoot>
	<tr>
	<td colspan="2">Всего: 3</td>
	</tr>
	</tfoot>
	</table>
	</body>
	</html>

Задача 11 (Псевдоэлементы)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа с текстом из 3 параграфов
4. Создайте новый CSS файл - **style.css**
5. Подключите CSS файл к HTML файлу
6. Установите цвет фона первой строки первого параграфа на красный
7. Установите цвет фона первой буквы элемента на зелёный во втором параграфе
8. Вставьте изображение до и после любого элемента третьего параграфа используя псевдоэлементы ::**before** и ::**after.**

CSS	HTML

p.m3:before
{
content: url("https://... ");
}
p.m3:after
{
content: url("https://... ");
}
p.m2:first-letter
{
background-color: red;
}
p.1:first-line
{
background-color: green;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	 …
	 …
	 …
	</body>
	</html>

Задача 12 (Прозрачность, цвета, селекторы)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа с текстом
4. Создайте новый CSS файл – **style.css**
5. Подключите CSS файл к HTML файлу
6. Установите прозрачность изображений на 50%
7. Используя **hover** *«удалите»* прозрачность изображения, когда на него наводится курсор мыши.
8. Используйте background-color:hsl, чтобы изменить оттенок, насыщенность и яркость фона элементов . Установите оттенок на красный, насыщенность на 100% и яркость на 50%.
9. Установите прозрачность фона одного из элементов на 40%

CSS	**HTML**

IMG
{
opacity: 0.5;
}
IMG:hover
{
opacity: 1.0;
}
p#d1
{
background-color:hsl(0, 100%, 50%);
}
p#d2
{
background-color:hsl(0, 100%, 50%);
opacity: 0.4;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body>
	<IMG SRC="1.jng"… >
	 …
	 …
	</body>
	</html>

Задача 13 (Фоны)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа с текстом
4. Создайте новый CSS файл – **style1.css**
5. Подключите CSS файл к HTML файлу
6. Добавьте 2 фоновых изображения в <**body**>
7. Установите ширину фоновых изображений на 400 пикселей, а высоту - на 400 пикселей.
8. Расположите два фоновых изображения в каждом верхнем углу. img1.gif слева и img2.gif справа.
9. Создайте новый CSS файл – **style2.css**
10. Подключите CSS файл к HTML файлу
11. Используя **background-clip** с помощью **padding-box** фон распространится до внешнего края отступа. Под границей фон не рисуется

CSS	HTML

style1.css
body
{
background-image: url("https://1... "), url("https://2... ");
background-size: 400px 400px;
background-repeat: no-repeat, no-repeat;
background-position: left top, right top;
}
 	
style2.css
p
{
background-color: yellow;
background-clip: padding-box;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body>
	 …
	</body>
	</html>

Задача 14 (Дисплей, видимость)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа с текстом из 3 параграфов и ненумерованного списка
4. Создайте новый CSS файл – **style.css**
5. Подключите CSS файл к HTML файлу
6. Скройте 1 элемент , чтобы он занимал столько же места, как и раньше
7. Скройте 2 элемент , чтобы он не занимал места
8. Отобразите элементы списка в виде встроенных элементов
9. Отобразите элементы в виде блочных элементов.

CSS	HTML

p.m1
{
visibility:hidden;
}
p.m2
{
display:none;
}
li
{
display:inline;
}
strong
{
display:block;
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	
	  …
	  …
	
	 …
	 …
	 …
	</body>
	</html>

Задача 15 (Трансформация, анимации)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа с 6 одинаковыми элементами ** **
4. Создайте новый CSS файл – **style.css**
5. Подключите CSS файл к HTML файлу
6. Установите для всех ** width: 100px; height: 100px; margin: 50px; background-color: lightblue; border: 1px solid black;**

7. С помощью свойства **transform** переместите элемент ** ** на 100 пикселей вправо и на 50 пикселей вниз.
8. С помощью свойства **transform** поверните элемент ** ** на 45 градусов.
9. С помощью свойства **transform** наклоните элемент ** ** на 20 градусов вдоль оси X и на 30 градусов вдоль оси Y.
10. С помощью свойства **transform** измените размер ** ** на половину его ширины, но удвойте его высоту
11. С помощью свойства **transform** поверните элемент ** ** на 150 градусов вокруг его оси X.
12. Добавьте 2 секундную анимацию, бесконечно повторяющуюся для элемента ** **, который меняет цвет с красного на синий. Назовите анимацию **"example".**

CSS	**HTML**

div
{
width: 100px;
height: 100px;
margin: 50px;
background-color: lightblue;
border: 1px solid black;
}
div#m1
{
transform:translate(100px, 200px);
}
div#m2
{
transform:rotate(45deg);
}
div#m3
{
transform:skew(20deg, 30deg);
}
div#m4
{
transform:scale(0.5, 2);
}
div#m5
{
transform:rotateX(150deg);
}
div#m6
{
animation-name:example;
animation-duration: 2s;
animation-iteration-count:infinite;
}
@keyframes example
{
from
{background-color: red;}
to
{background-color: blue;}
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	 …
	
	 …
	
	 …
	
	 …
	
	 …
	
	 …
	</body>
	</html>

Задача 16 (Теневые эффекты, градиент)

1. Создайте папку
2. В этой папке создайте новый HTML документ - **index.html**
3. В **index.html** создайте тело HTML документа с текстом из 2 одинаковыми элементами ** ** и 2 параграфами
4. Создайте новый CSS файл – **style.css**
5. Подключите CSS файл к HTML файлу
6. Установите текстовую тень "2 пикселя" по горизонтали и "2 пикселя" по вертикали для 1 элемента ** .**
7. Установите "10 пикселей" по горизонтали и "10 пикселей" по вертикали для тени блока для 1 элемента ** .**
8. Установите "2 пикселя" по горизонтали и "2 пикселя" по вертикали, "зеленую" текстовую тень для 2 элемента ** ** с радиусом размытия "5 пикселей" и "3 пикселя" по горизонтали и "3 пикселя" по вертикали "красную" текстовую тень с радиусом размытия "10 пикселей".
9. Установите линейный градиентный фон для элемента ** **, идущий сверху вниз, переходя от "белого" к "зеленому".

CSS	**HTML**

p#k1
{
text-shadow: 2px 2px;
}
div#m1
{
box-shadow: 10px 10px;
}
p#k2
{
text-shadow: 2px 2px 5px green, 3px 3px 10px red;
}
div# m2
{
background-image:linear-gradient(white, green);
}
	<html>
	<head>
	<title>группа</title>
	<link type="text/css"
	rel="stylesheet"
	href="style.css">
	</head>
	<body >
	 …
	 …
	 …
	 …
	</body>
	</html>

Составил:

к.т.н., доцент каф. РТС _____ (Макарова О.Н)

Заведующий кафедрой РТС

д.т.н., профессор _____ (Кошелев В.И.)

Оператор ЭДО ООО "Компания "Тензор"

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ

ПОДПИСАНО **ФГБОУ ВО "РГРТУ", РГРТУ,** Кошелев Виталий Иванович, Заведующий кафедрой РТС **17.12.24** 14:46 (MSK) Простая подпись