


Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Смирнов Сергей Николаевич
Должность: врио ректора
Дата подписания: 27.03.2025 16:03:41
Уникальный программный ключ:
69e375c64f7e975d4e8830e7b4fc9ad1b65308

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Тверской государственный университет»

Утверждаю:
Руководитель ООП
СМ /С.М.Дудаков/
«07» *сентября* 2024 года



Рабочая программа дисциплины (с аннотацией)

ВЕБ-РАЗРАБОТКА

Направление подготовки
02.03.02 ФУНДАМЕНТАЛЬНАЯ ИНФОРМАТИКА
И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Направленность (профиль)
Программная инженерия в искусственном интеллекте

Для студентов 2-го курса
Очная форма

Составитель: И.С. Солдатенко

Тверь, 2024

I. Аннотация

1. Цель и задачи дисциплины

Целью освоения дисциплины является:

Изучение современных подходов, средств и технологий веб-разработки.

Задачами освоения дисциплины являются:

Изучение основных понятий, методов, средств и технологий проектирования современных веб-систем, знакомство с методологиями моделирования предметных областей и информационного обеспечения информационных систем.

2. Место дисциплины в структуре ООП

Дисциплина является элективной и относится к части учебного плана, формируемой участниками образовательных отношений.

Предварительные знания и навыки:

Для успешного освоения дисциплины необходимы навыки программирования, полученные в ходе изучения дисциплин «Программирование» и «Программирование для искусственного интеллекта».

Дальнейшее использование:

Полученные знания и навыки используются в последующем во всех дисциплинах, где требуется умение разрабатывать веб-приложения, в том числе при подготовке выпускной квалификационной работы.

3. Объем дисциплины:

2 зачетных единицы, 72 академических часа, в том числе:

- **контактная аудиторная работа:**
практические занятия 15 часов;
лабораторные работы 30 часов, в т.ч. практическая подготовка 22 часа;
- **контактная внеаудиторная работа:** контроль самостоятельной работы и курсовая работа не предусмотрены учебным планом;
- **самостоятельная работа:**
27 часов, в том числе контроль 0 часов.

4. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Планируемые результаты освоения образовательной программы (формируемые компетенции)	Планируемые результаты обучения по дисциплине
ПК-3. Способен осуществлять концептуальное моделирование проблемной области и проводить формализацию представления знаний в системах искусственного интеллекта	ПК-3.1. Разрабатывает концептуальную модель проблемной области системы искусственного интеллекта ПК-3.2. Выбирает методы представления знаний и проектирует базу знаний системы искусственного интеллекта

ПК-8. Способен разрабатывать системы анализа больших данных	ПК-8.1. Разрабатывает программные компоненты извлечения, хранения, подготовки больших данных с учетом вариантов использования больших данных, определений, словарей и эталонной архитектуры больших данных ПК-8.2. Разрабатывает программные компоненты обработки, удаленной, распределенной и объединенной аналитики, использования результатов анализа, описания и управления качеством и достоверностью больших данных
---	--

5. Форма промежуточной аттестации и семестр прохождения:
зачет в 3-м семестре.

6. Язык преподавания русский.

II. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

1. Для студентов очной формы обучения

Учебная программа – наименование разделов и тем	Всего (час.)	Контактная работа (час.)				Контроль самостоятельной работы (в том числе курсовая работа)	Самостоятельная работа, в том числе контроль (час.)
		Практические занятия		Лабораторные работы			
		всего	в т.ч. практическая подготовка	всего	в т.ч. практическая подготовка		
1. Введение в профессию	12	1		0		0	3
2. История интернета. HTML	10	1		2	2	0	2
3. CSS. Блочная верстка	12	1		4	2	0	2
4. Веб: история, архитектура, HTTP+URI+HTML. REST	14	1		2	2	0	3
5. Javascript: DOM, синтаксис, прототипно-ориентированная архитектура языка. Асинхронные приложения	20	2		6	4	0	3
6. Технологии: Git, Github	10	1		2	2	0	2
7. Информационная архитектура веб-приложений	12	1		0		0	3
8. Технологии: Node.js + NPM	12	1		4	2	0	2
9. Введение в React. CSS и JS-фреймворки. SPA	14	2		4	2	0	2

10. JSON, API, микросервисы, MVC	12	2		2	2	0	2
11. Технологии: использование отладчика в браузере	6	1		2	2	0	0
12. Redux, Saga	10	1		2	2	0	3
ИТОГО	72	15		30	22	0	27

III. Образовательные технологии

Учебная программа – наименование разделов и тем	Вид занятия	Образовательные технологии
1. Введение в профессию	<ul style="list-style-type: none"> лекция 	<ul style="list-style-type: none"> традиционные (фронтальная лекция), цифровые (показ презентаций, выполнение компьютерных лабораторных работ), технология проблемного обучения, групповая проектная работа творческое задание
2. История интернета. HTML	<ul style="list-style-type: none"> лекция лабораторная работа домашние задания 	
3. CSS. Блочная верстка		
4. Веб: история, архитектура, HTTP+URI+HTML. REST		
5. Javascript: DOM, синтаксис, прототипно-ориентированная архитектура языка. Асинхронные приложения		
6. Git, Github		
7. Информационная архитектура веб-приложений	<ul style="list-style-type: none"> лекция 	
8. Технологии: Nodejs + NPM	<ul style="list-style-type: none"> лекция лабораторная работа домашние задания проектная работа 	
9. Введение в React. CSS и JS-фреймворки. SPA		
10. JSON, API, микросервисы, MVC		
11. Технологии: использование отладчика в браузере		
12. Redux, Saga		

IV. Оценочные материалы для проведения текущей и промежуточной аттестации

№	Примерная формулировка заданий	Вид/способ	Критерии оценивания
ПК-3. Способен осуществлять концептуальное моделирование проблемной области и проводить формализацию представления знаний в системах искусственного интеллекта			
1	Лабораторная работа №1 «HTML» Оформление с помощью HTML и CSS страницы по заданному образцу. Лабораторная работа направлена на изучение всех элементов языка HTML. Примерный фрагмент задания приведен в разделе VI.	вид: лабораторная работа способ: на компьютере результаты: HTML-файл.	Максимум – 3 б. Работа состоит из трех разделов, каждый из которых оцениваться в 0.5 баллов. За неиспользование корректного HTML-элемента снимается 0.1 балл.
2	Лабораторная работа №2 «HTML+CSS» Самостоятельное изучение и выполнение упражнений на основы HTML и CSS на ресурсе https://htmlacademy.ru . Лабораторная работа направлена на получение навыка использования языков HTML и CSS.	вид: лабораторная работа способ: на компьютере результаты: решенные упражнения на https://htmlacademy.ru .	Максимум – 6 б. Работа состоит из выполнения 100 элементарных упражнений электронного учебника. За одно корректно выполненное упражнение дается 0.03 балла.
3	Лабораторная работа №3 «HTML-верстка» Самостоятельное групповое задание, построенное по принципам проблемного обучения. Дается графический макет страницы. Необходимо сверстать его с помощью HTML и CSS как можно ближе к оригиналу. Пример макета приведен в разделе VI.	вид: самостоятельная групповая работа способ: на компьютере результаты: HTML и CSS файлы со сверстанным макетом.	Максимум – 6 б. Критерии оценки: <ul style="list-style-type: none"> • верстка идентична оригиналу, использованы принципы блочной верстки – макс. балл, • верстка идентична оригиналу, однако использованы принципы табличной верстки – 1.5 балла, • за каждый некорректно сверстанный раздел страницы (шапка, подвал, меню, ...) снимается по 0.5 баллов. <p>Если задание выполнено не в группе, а индивидуально, то полученная сумма баллов делится на два.</p>
4	Лабораторная работа №4 «Git» Задание направлено на изучение принципов работы с распределенной системой контроля версий Git и ресурса Github. Состоит из трех заданий. Первые	вид: групповая лабораторная работа способ:	Максимум – 3 б.

	<p>два выполняются индивидуально, а третье - строго в паре. Пример формулировки одного из подзаданий:</p> <ol style="list-style-type: none"> 1. Скачайте полученный после форка репозиторий локально. 2. Создайте ветку feature/damascus, перейдите в нее. 3. В файле damascus.html вынесите css из div элемента в тег <style>. 4. Отправьте изменения на сервер в ветку feature/damascus. 5. Следующие пункты выполняются после того, как студент 1 выполнит пункты 2-4 в своей части задания: <ol style="list-style-type: none"> 1. Получите изменения с удаленного репозитория. 2. Перейдите в ветку feature/athens. 3. Получите изменения с удаленного репозитория для ветки feature/athens. 4. В файле athens.html замените тег <h2> на <h3>. 5. Отправьте изменения в удаленный репозиторий в ветку feature/athens. 	<p>на компьютере</p> <p>результаты: репозиторий на Github.</p>	<p>Состоит из восьми частей. Первые семь частей (Задания 1 и 2) оцениваются в 1.05 балла каждое. Последняя часть (Задание 3) в 0.45 баллов.</p> <p>Если задание выполнено не в группе, а индивидуально, то полученная сумма баллов делится на два.</p>
5	<p>Лабораторная работа №5 «JavaScript» Разработка веб-приложения «Игра крестики-нолики» на языке JavaScript. Формулировка задания приведена в разделе VI.</p>	<p>вид: лабораторная работа</p> <p>способ: на компьютере</p> <p>результаты: приложение на JS.</p>	<p>Максимум – 5 б.</p> <p>Критерии оценки:</p> <ul style="list-style-type: none"> • приложение разработано полностью и корректно – макс. балл, • отсутствие «ничьей» - минус 0.5 баллов, • игра «не останавливается» после последнего хода – минус 0.5 баллов, • игровое поле не соответствует макету – минус 0.5 баллов, • отсутствует подсветка выигрышной комбинации – минус 0.5 баллов.
6	<p>Лабораторная работа №6 «JavaScript. Часть II» Разработка индивидуального веб-приложения на языке JavaScript. Пример задания приведен в разделе VI.</p>	<p>вид: лабораторная работа</p> <p>способ: на компьютере</p> <p>результаты: приложение на JS.</p>	<p>Максимум – 5 б.</p> <p>Критерии оценки: Приложение разработано полностью и корректно – макс. балл.</p>
7	<p>Лабораторная работа №7 «Node.js. Часть I» Необходимо разработать простой веб-сервер на платформе Node.js:</p> <ul style="list-style-type: none"> • Пользователь должен видеть страницу приветствия по адресу http://domain/start 	<p>вид: лабораторная работа</p> <p>способ: на компьютере</p> <p>результаты: приложение на Node.js.</p>	<p>Максимум – 5 б.</p> <p>Критерии оценки:</p> <ul style="list-style-type: none"> • Приложение разработано полностью и корректно – макс. балл.

	<ul style="list-style-type: none"> • Когда запрашивается http://domain/upload, пользователь должен иметь возможность загрузить картинку со своего компьютера и просмотреть её в своем браузере. 		<ul style="list-style-type: none"> • Изображение загружается, но не отображается – минус 1 балл, • Загруженную картинку можно посмотреть только непосредственно после загрузки и никак иначе – минус 1 балл.
8	<p>Лабораторная работа №8 «Node.js. Часть II» Необходимо выполнить индивидуальную доработку веб-сервера, созданного в рамках выполнения лабораторной работы №7. Пример доработки: Добавить фильтр формата загружаемого файла (на стороне сервера). Если загружается не PNG, то пользователь должен получить страницу с сообщением об ошибке.</p>	<p>вид: лабораторная работа способ: на компьютере результаты: приложение на Node.js.</p>	<p>Максимум – 7 б. Критерии оценки: Приложение разработано полностью и корректно – макс. балл.</p>
9	<p>Лабораторная работа №9 «React» Необходимо разработать веб-приложение «Игра крестики-нолики» на платформе React. Приложение разрабатывается с помощью электронного тьюториала с заданиями для самостоятельной проработки.</p>	<p>вид: лабораторная работа способ: на компьютере результаты: приложение на React.</p>	<p>Максимум – 5 б. Критерии оценки: Приложение разработано полностью и корректно – макс. балл.</p>
10	<p>Модульная контрольная работа 1 Тест на знание основ языков HTML, CSS, JavaScript. Примеры вопросов:</p> <ol style="list-style-type: none"> 1. HTML изначально придуман для <ol style="list-style-type: none"> 1. для упорядочения научной информации и обмена информацией между учеными из разных организаций 2. для рекламирования товаров и услуг коммерческих компаний 3. в качестве научного эксперимента 4. случайно, однако впоследствии приобрел огромную популярность 2. Какой CSS-код написан правильно? <ol style="list-style-type: none"> 1. <code>div {border: 1px solid #ccc;}</code> 2. Никакой 3. <code><div> {border: 1px solid #ccc;}</code> 4. <code><div> {border: 1px solid #hhh;}</code> 5. <code>div {border: 1px solid #hhh;}</code> 	<p>вид: тестирование способ: на компьютере</p>	<p>Максимум – 15 б. Критерии оценки: За каждый вопрос дается 0.15 или 0.2 балла. Результатом является сумма баллов, полученных за корректные ответы.</p>
<p>ПК-8. Способен разрабатывать системы анализа больших данных</p>			
11	<p>Модульная контрольная работа 2 Необходимо разработать веб-приложение средней сложности на платформе React с использованием библиотек bootstrap, redux, router и других. Суть задания – разработать приложение для некоторой клиники, в котором доктора всегда могут видеть список приёмов пациентов, события, оповещения,</p>	<p>вид: самостоятельная лабораторная работа способ: на компьютере результаты:</p>	<p>Максимум – 15 б. Критерии оценки: Приложение разработано полностью и корректно – макс. балл.</p>

	сообщения, клиентов, а также найти информацию своих коллегам. Приложение выполняется с помощью электронного тьюториала, содержащего некоторое количество тем для самостоятельного изучения.	приложение на React.	
--	--	----------------------	--

Шкала оценивания овладения компетенциями:

Индикатор считается выполненным, если студент набрал как минимум пороговое количество баллов за те виды активности, которые отвечают за данный индикатор. Типовые оценочные материалы с привязкой к отдельным компетенциям приведены в таблице выше.

№	Индикатор	Текущая аттестация	
		Порог	Максимум
1	ПК-3.1, ПК-3.2	18	45
2	ПК-8.1, ПК-8.2	6	15

Шкала и критерии выставления оценок за дисциплину:

Шкала и критерии выставления оценок «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно» описаны в локальной нормативной документации Тверского государственного университета (Положение о рейтинговой системе обучения студентов ТвГУ). Положительная оценка может быть выставлена только в том случае, если выполнены все индикаторы.

V. Учебно-методическое и информационное обеспечение дисциплины

1) Рекомендуемая литература

а) Основная литература:

1. Кириченко, А. В. Web на практике. CSS, HTML, JavaScript, MySQL, PHP для fullstack-разработчиков / А. В. Кириченко, А. П. Никольский, Е. В. Дубовик. — Санкт-Петербург: Наука и Техника, 2021. — 432 с. — ISBN 978-5-94387-271-6. — Текст: электронный // Лань: электронно-библиотечная система. — Режим доступа: — URL: <https://e.lanbook.com/book/191460>
2. Технология разработки интернет ресурсов: курс лекций: [16+] / авт.-сост. И.А. Журавлёва; Министерство науки и высшего образования Российской Федерации, Северо-Кавказский федеральный университет. – Ставрополь: Северо-Кавказский Федеральный университет (СКФУ), 2018. – 171 с.: ил. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=562579>
3. Крахоткина, Е.В. Технологии разработки Internet-приложений: учебное пособие / Е.В. Крахоткина; Северо-Кавказский федеральный университет. – Ставрополь: Северо-Кавказский Федеральный университет (СКФУ), 2016. – 124 с: ил. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=459070>

б) Дополнительная литература:

1. Вагин, Д.В. Современные технологии разработки веб-приложений: учебное пособие: [16+] / Д.В. Вагин, Р.В. Петров; Новосибирский государственный технический университет. – Новосибирск: Новосибирский государственный технический университет, 2019. – 52 с.: ил. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=573960>

2) Программное обеспечение

Компьютерный класс факультета прикладной математики и кибернетики № 46 (170002, Тверская обл., г.Тверь, Садовый переулок, д.35)	
Adobe Acrobat Reader DC - Russian	бесплатно
Apache Tomcat 8.0.27	бесплатно
Cadence SPB/OrCAD 16.6	Государственный контракт на поставку лицензионных программных продуктов 103 - ГК/09 от 15.06.2009
GlassFish Server Open Source Edition 4.1.1	бесплатно
Google Chrome	бесплатно
Java SE Development Kit 8 Update 45 (64-bit)	бесплатно
JetBrains PyCharm Community Edition 4.5.3	бесплатно
JetBrains PyCharm Edu 3.0	бесплатно
Kaspersky Endpoint Security 10 для Windows	Акт на передачу прав ПК545 от 16.12.2022
Lazarus 1.4.0	бесплатно
Mathcad 15 M010	Акт предоставления прав ИС00000027 от 16.09.2011
MATLAB R2012b	Акт предоставления прав № Us000311 от 25.09.2012
Многофункциональный редактор ONLYOFFICE бесплатное ПО	бесплатно
ОС Linux Ubuntu бесплатное ПО	бесплатно
MiKTeX 2.9	бесплатно
MSXML 4.0 SP2 Parser and SDK	бесплатно
NetBeans IDE 8.0.2	бесплатно
NetBeans IDE 8.2	бесплатно
Notepad++	бесплатно
Oracle VM VirtualBox 5.0.2	бесплатно
Origin 8.1 Sr2	договор №13918/M41 от 24.09.2009 с ЗАО «СофтЛайн Трейд»
Python 3.1 pygame-1.9.1	бесплатно
Python 3.4 numpy-1.9.2	бесплатно
Python 3.4.3	бесплатно
Python 3.5.1 (Anaconda3 2.5.0 64-bit)	бесплатно
WCF RIA Services V1.0 SP2	бесплатно
WinDjView 2.1	бесплатно

R Studio	бесплатно
Anaconda3 2019.07 (Python 3.7.3 64-bit)	бесплатно

3) Современные профессиональные базы данных и информационные справочные системы

- ЭБС «IPR BOOKS» <http://www.iprbookshop.ru>
- ЭБС «Университетская библиотека онлайн» <https://biblioclub.ru>
- ЭБС «Лань» <http://e.lanbook.com>

4) Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

- Электронная образовательная среда ТвГУ <http://lms.tversu.ru>
- Научная библиотека ТвГУ <http://library.tversu.ru>
- Самоучитель по HTML4 <http://htmlbook.ru/samhtml>
- Сайт интерактивных онлайн-курсов <https://htmlacademy.ru>
- Система GitHub <https://github.com>
- Сайт распределенной системы управления версиями <https://git-scm.com>
- Современный учебник JavaScript <https://learn.javascript.ru>
- MDN Web Docs <https://developer.mozilla.org/en-US/docs/Web/Tutorials>
- Node.js для начинающих <http://nodebeginner.ru>
- Сайт платформы Node.js <https://nodejs.org>
- Полное руководство по React <https://learn-reactjs.ru/home>
- Сайт библиотеки React <https://reactjs.org>

VI. Методические материалы для обучающихся по освоению дисциплины

Важной составляющей данного раздела РПД являются требования к рейтинг-контролю с указанием баллов, распределенных между модулями и видами работы обучающихся.

Максимальная сумма баллов по учебной дисциплине, заканчивающейся экзаменом, по итогам семестра составляет 60 баллов (30 баллов - 1-й модуль и 30 баллов - 2-й модуль).

Обучающемуся, набравшему 40–54 балла, при подведении итогов семестра (на последнем занятии по дисциплине) в рейтинговой ведомости учета успеваемости и зачетной книжке может быть выставлена оценка «удовлетворительно».

Обучающемуся, набравшему 55–57 баллов, при подведении итогов семестра (на последнем занятии по дисциплине) в графе рейтинговой ведомости учета успеваемости «Премияльные баллы» может быть добавлено 15 баллов и выставлена экзаменационная оценка «хорошо».

Обучающемуся, набравшему 58–60 баллов, при подведении итогов семестра (на последнем занятии по дисциплине) в графе рейтинговой ведомости учета успеваемости «Премияльные баллы» может быть добавлено 27 баллов и

выставлена экзаменационная оценка «отлично». В каких-либо иных случаях добавление премиальных баллов не допускается.

Обучающийся, набравший до 39 баллов включительно, сдает экзамен.

Распределение баллов по модулям устанавливается преподавателем и может корректироваться.

1. Структура рейтинговых баллов

Название работы	Кол-во баллов
ТЕКУЩАЯ АТТЕСТАЦИЯ	
Первый модуль	
Лабораторная работа №1 «HTML»	1.5
Лабораторная работа №2 «HTML+CSS»	3
Лабораторная работа №3 «HTML-верстка»	3
Лабораторная работа №4 «Git»	1.5
Лабораторная работа №5 «JavaScript»	3
Лабораторная работа №6 «JavaScript. Часть II»	3
Модульная контрольная 1	15
ИТОГО за первый модуль	30
Второй модуль	
Лабораторная работа №7 «Node.js. Часть I»	3
Лабораторная работа №8 «Node.js. Часть II»	7
Лабораторная работа №9 «React»	5
Модульная контрольная 2	15
ИТОГО за второй модуль	30
ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ	
Экзамен	40

2. Примеры заданий для самостоятельных лабораторных работ

2.1 Пример задания для лабораторной №1 «HTML»

Оформите с помощью HTML и CSS содержимое страницы

3. Элементы внутренней разметки (inline elements) 3.1. Элемент FONT

Элемент FONT изменяет размер, цвет и шрифт для текста. Начиная с версии HTML 4.0 перенесен в категорию "нежелательные элементы" (deprecated) и подлежит замене на элемент SPAN.

3.1.1. Атрибут SIZE: размер текста

```
<!-- ОФОРМИТЕ КАЖДУЮ СТРОЧКУ ЭЛЕМЕНТОМ FONT С СООТВЕТСТВУЮЩИМ ЗНАЧЕНИЕМ SIZE -->
```

Этот текст первого размера

Этот текст второго размера

Этот текст третьего размера (таким оформляется обычный текст)

Этот текст четвёртого размера

Этот текст пятого размера

Этот текст шестого размера (как заголовок H1)

Этот текст седьмого размера

3.1.2. Атрибут COLOR: цвет текста

```
<!-- каждую строчку СООТВЕТСТВУЮЩИМ ЦВЕТОМ -->
```

красный (Red или #FF0000)

зелёный (Green или #008000) синий (Blue или #0000FF)

чёрный (Black или #000000)

серый (Gray или #808080)

светло-серый Silver или #C0C0C0)

3.1.3. Атрибут FACE: шрифтовое оформление текста

```
<!-- каждую строчку СООТВЕТСТВУЮЩИМ ШРИФТОМ-->
```

текст без засечек (Arial, Tahoma, Verdana, Helvetica, sans-serif)

текст с зачечками (Georgia, Times, serif)

моноширинный текст (Courier, mono)

курсивный текст (Cursive)

декоративный текст (Fantasy)

3.2. Элемент SPAN

Элемент SPAN позволяет оформлять содержимое с использованием каскадных таблиц стилей.

Пример:

Обратите внимание на цвет и фон данного текста, а также на рамку вокруг него.

так, чтобы она выглядела как на макете:

3. Элементы внутренней разметки (inline elements)

3.1. Элемент FONT

Элемент FONT изменяет размер, цвет и шрифт для текста. Начиная с версии HTML 4.0 перенесен в категорию "нежелательные элементы" (deprecated) и подлежит замене на элемент SPAN.

3.1.1. Атрибут SIZE: размер текста

Size=1 Size=2 Size=3 Size=4 Size=5 Size=6 Size=7

3.1.2. Атрибут COLOR: цвет текста

Black = "#000000" : пример текста

White = "#FFFFFF" :

Gray = "#808080" : пример текста

Silver = "#C0C0C0" : пример текста

Red = "#FF0000" : пример текста

Green = "#008000" : пример текста

Blue = "#0000FF" : пример текста

Другие цвета:

Lime = "#00FF00" Olive = "#808000" Maroon = "#800000" Navy = "#000080" Purple = "#800080" Teal = "#008080" Fuchsia = "#FF00FF" Aqua = "#00FFFF"

3.1.3. Атрибут FACE: шрифтовое оформление текста

FACE=FANTASY: пример текста (example text)

FACE=MONOSPACE: пример текста (example text)

FACE=CURSIVE: пример текста (example text)

FACE=SANS-SERIF: пример текста (example text)

3.2. Элемент SPAN

Элемент SPAN позволяет оформлять содержимое с использованием каскадных таблиц стилей.

Пример:

Обратите внимание на цвет и фон **данного текста**, а также на рамку вокруг него.

2.2 Пример задания для лабораторной №3 «HTML-верстка»

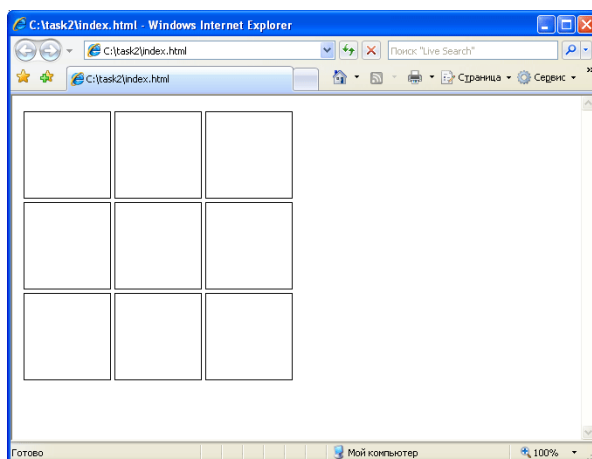
Сверстать с помощью стилей как можно ближе к оригиналу:



2.3 Пример задания для лабораторной №5 «JavaScript»

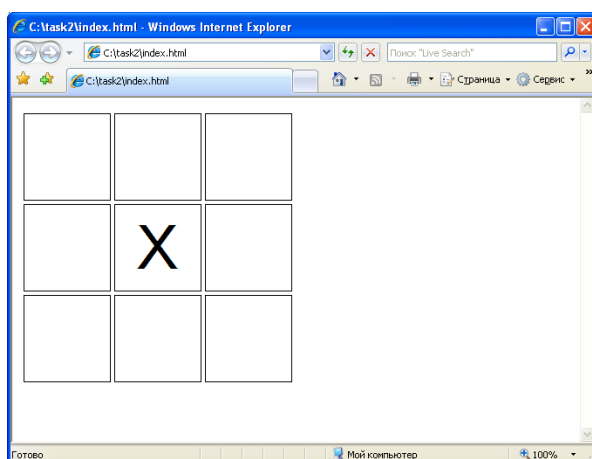
Напишите при помощи JavaScript игру «Крестики-нолики» (для двух игроков-людей). Весь код игры должен содержаться на одной HTML-странице.

Окно с неначатой игрой должно выглядеть примерно следующим образом:

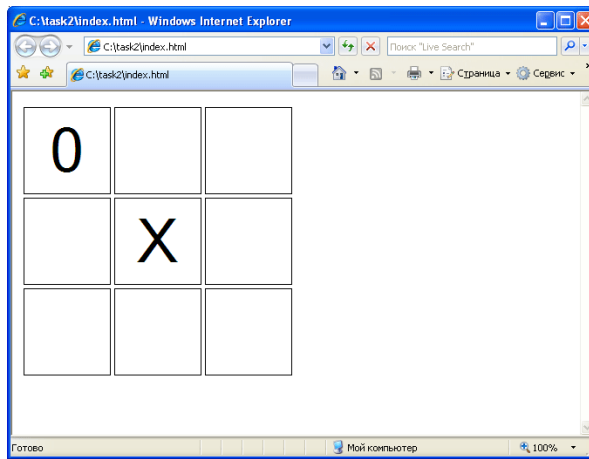


Пустые белые квадраты должны быть элементами формы типа `<input type="button">`, внешний вид которых должен быть изменен при помощи CSS. Внешний вид вашего поля для игры в крестики-нолики может по вашему желанию отличаться от того, что приведен на рисунке - главное, чтобы оно было задано при помощи кнопок и чтобы они выглядели «интересно».

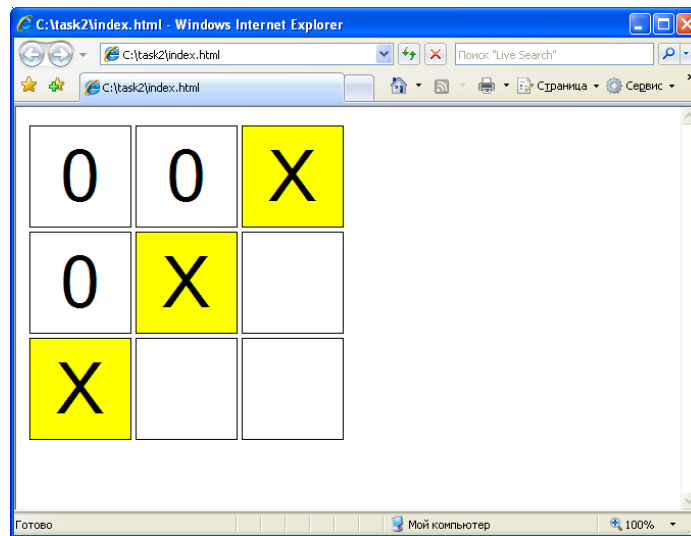
В игру играют двое игроков. Сначала первый игрок нажимает на любую кнопку. На нажатой кнопке должен отобразиться символ 'X' (надписи на кнопках задаются при помощи атрибута `value` тега `input`, шрифт надписи на кнопке можно изменить также при помощи CSS):



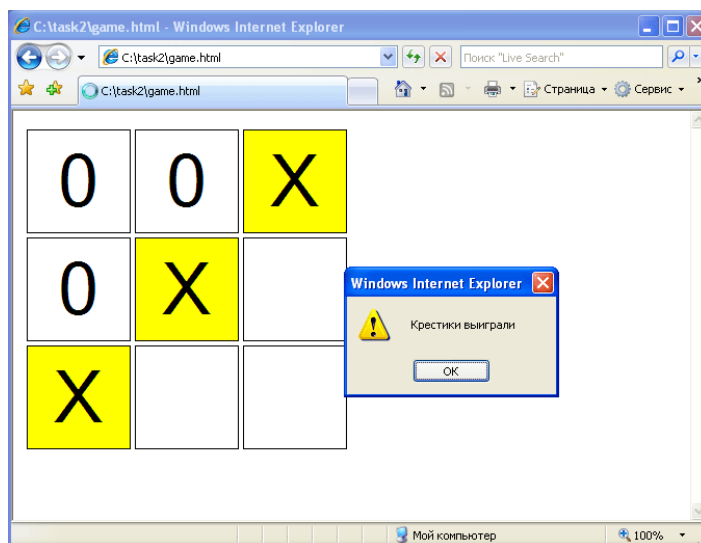
После этого ход делает второй игрок - нажимает на любую свободную кнопку. Если второй игрок нажимает на ту же кнопку, что и первый игрок, то ход не засчитывается и второй игрок должен снова ходить. Если второй игрок нажимает на свободную кнопку, то на ней отображается символ 'O':



Так поочередно игроки делают свои ходы. Как только выполняется условие выигрыша (вертикаль, горизонталь или диагональ полностью заполнены одинаковыми символами), необходимо подсветить эту строку каким-либо цветом...

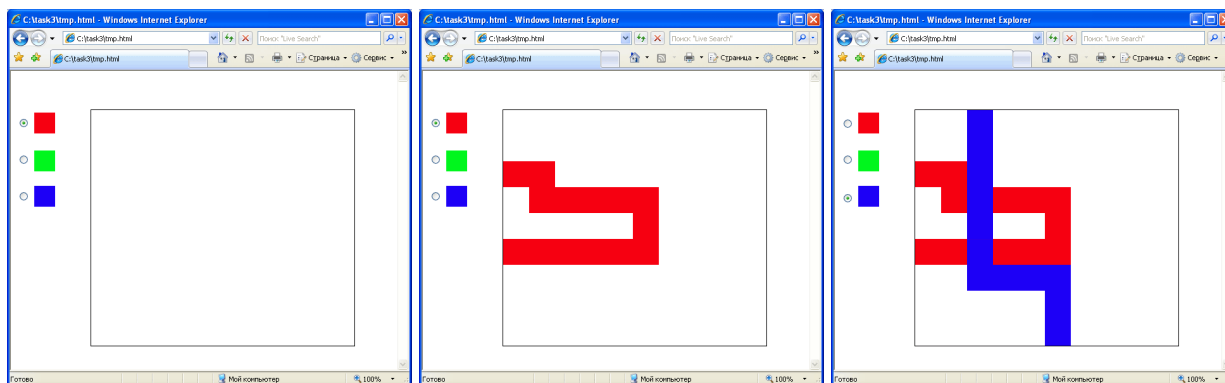


... и выдать окно с надписью "Крестики выиграли" или "Нолики выиграли" - в зависимости от того, кто на самом деле выиграл.



2.4 Пример задания для лабораторной №6 «JavaScript. Часть II»

Разработайте простейший квадратный графический редактор с тремя цветами. Рабочее поле текстового редактора состоит из NxN квадратиков - небольших квадратных картинок белого цвета. При наведении на любой квадратик мышью, он меняет свой текущий цвет на выбранный. "Выбранный" цвет выбран в палитре, которая приводится слева.



3. Указания для обучающихся

Организуя свою учебную работу, студенты должны, во-первых, выявить рекомендуемый режим и характер учебной работы по изучению теоретического курса, практическому применению изученного материала, по выполнению заданий для самостоятельной работы, по использованию информационных технологий и т.д. Во-вторых, ознакомиться с указанным в методическом материале по дисциплине перечнем учебно-методических изданий, рекомендуемых студентам для подготовки к занятиям и выполнения самостоятельной работы, а также с методическими материалами на бумажных и/или электронных носителях, выпущенных кафедрой своими силами и предоставляемые студентам во время занятий.

Самостоятельная работа студентов, предусмотренная учебным планом, должна соответствовать более глубокому усвоению изучаемого курса, формировать навыки исследовательской работы и ориентировать студентов на умение применять теоретические знания на практике.

1. Работа с учебными пособиями.

Для полноценного усвоения курса студент должен, прежде всего, овладеть основными понятиями этой дисциплины. Необходимо усвоить определения и понятия, уметь приводить их точные формулировки, приводить примеры объектов, удовлетворяющих этому определению. Кроме того, необходимо знать круг фактов, связанных с данным понятием. Требуется также знать связи между понятиями, уметь устанавливать соотношения между классами объектов, описываемых различными понятиями.

2. Самостоятельное изучение тем.

Самостоятельная работа студента является важным видом деятельности, позволяющим хорошо усвоить изучаемый предмет и одним из условий достижения необходимого качества подготовки и профессиональной переподготовки

специалистов. Она предполагает самостоятельное изучение студентом рекомендованной учебно-методической литературы, различных справочных материалов, написание рефератов, выступление с докладом, подготовку к лекционным и практическим занятиям, подготовку к зачёту и экзамену.

3. Подготовка к практическим занятиям.

При подготовке к практическим занятиям студентам рекомендуется следовать методическим рекомендациям по работе с учебными пособиями, приведенным выше.

4. Составление конспектов.

В конспекте отражены основные понятия темы. Для наглядности и удобства запоминания используются схемы и таблицы.

VII. Материально-техническое обеспечение

Для аудиторной работы

Учебная аудитория № 212 (170002, Тверская обл., г.Тверь, Садовый переулок, д.35)	Набор учебной мебели, мультимедийный комплекс (доска, проектор, панель управления).
--	---

Для самостоятельной работы

Помещение для самостоятельной работы обучающихся: Компьютерный класс факультета прикладной математики и кибернетики № 46 (170002, Тверская обл., г.Тверь, Садовый переулок, д.35)	Компьютер, экран, проектор, кондиционер.
--	--

VIII. Сведения об обновлении рабочей программы дисциплины

№ п.п.	Обновленный раздел рабочей программы дисциплины	Описание внесенных изменений	Реквизиты документа, утвердившего изменения
1.	I. Аннотация 3. Объем дисциплины 5. Форма промежуточной аттестации и семестр прохождения II. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий	Изменения в количестве зачетных единиц, часов на контактную и самостоятельную работу, замена формы промежуточной аттестации	От 16.01.2025 года, протокол № 7 ученого совета факультета

2.			
3.			
4.			
5.			