

Частное образовательное учреждение высшего образования
«САНКТ-ПЕТЕРБУРГСКИЙ УНИВЕРСИТЕТ ТЕХНОЛОГИЙ
УПРАВЛЕНИЯ И ЭКОНОМИКИ»

РАССМОТРЕНО И ОДОБРЕНО

УТВЕРЖДАЮ

На заседании кафедры
информационных технологий и
математики
Протокол № 9 от 25.05.2023 г.

Первый проректор
С.В. Авдашкевич
28.06.2023

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Дисциплина:	Б1.В.ДВ.02.01 Интернет-программирование
Направление подготовки:	09.03.03 Прикладная информатика
Направленность (профиль):	Прикладная информатика в экономике
Уровень высшего образования:	Бакалавриат
Форма обучения:	очная, заочная
Разработчики:	Кандидат технических наук, доцент Бородянский Ю.М. Старший преподаватель Смирнова С. Л.

Санкт-Петербург
2023

1. Цели и задачи дисциплины:

Цель освоения дисциплины:

изучение основных концепций и принципов Web-дизайна и Internet-программирования.

Задачи дисциплины:

- подготовка будущих специалистов к использованию принципов Web-дизайна и Internet-программирования и средств их реализации в составе информационных систем для решения экономических задач.

2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы высшего образования

Планируемые результаты освоения ОП ВО (код и содержание компетенций)	Планируемые результаты обучения по ОП ВО (индикаторы достижения компетенций)	Примечание
ПК-2 Способен разрабатывать и адаптировать прикладное программное обеспечение	ПК-2.1 Знать инструменты и методы модульного тестирования; инструменты и методы прототипирования пользовательского интерфейса; инструменты и методы тестирования нефункциональных и функциональных характеристик ИС; методы оценки качества программных систем; основы программирования; основы современных операционных систем; основы современных систем управления базами данных; основы управления взаимоотношениями с клиентами и заказчиками (CRM); программные средства и платформы инфраструктуры информационных технологий организаций; современные стандарты информационного взаимодействия систем; современные структурные языки программирования; теория тестирования; языки программирования и работы с базами данных; языки современных бизнес-приложений.	06.015 Профессиональный стандарт «Специалист по информационным системам» 06.022 Профессиональный стандарт «Системный аналитик»
	ПК-2.2 Уметь алгоритмизировать деятельность; кодировать на языках программирования; тестировать результаты прототипирования.	
	ПК-2.3 Владеть навыками проведения анализа результатов тестов; координирования и проведения оценки готовых систем; обучения участников рабочей группы методике оценки готовых систем; оформления отчета о степени соответствия готовых систем требованиям; подготовки методике оценки готовых систем на соответствие требованиям; принятия решения о пригодности архитектуры; разработки прототипа ИС в соответствии с требованиями; осуществления сбора, обработки и анализа результатов оценки готовых систем на соответствие требованиям; согласования пользовательского интерфейса с заказчиком; тестирования прототипа ИС на проверку корректности архитектурных решений.	

Планируемые результаты обучения по ОП ВО (индикаторы достижения компетенций)	Планируемые результаты обучения по дисциплине
ПК-2.1. Знать инструменты и методы модульного тестирования; инструменты и методы прототипирования пользовательского ин-терфейса; инструменты и методы тестирования нефункциональных и функциональных характеристик ИС; методы оценки качества программных систем; основы программирования; ос-новы современных операционных систем; основы современных систем управления базами данных; основы управления взаимоотношениями с клиентами и заказчиками (CRM); программ-ные средства и платформы инфраструктуры информационных технологий организаций; современные стандарты информаци-онного взаимодействия систем; современные структурные язы-ки программирования; теория тестирования; языки программи-рования и работы с базами данных; языки современных бизнес-приложений.	- основы языка гипертекстовой разметки HTML - основы форматирования web-страниц с помощью CSS - принципы валидации web-страниц - основы программирования на стороне клиента на языке JavaScript - основы программирования на стороне сервера на языке PHP
ПК-2.2. Уметь алгоритмизировать деятельность; кодировать на языках программирования; тестировать результаты прототипирова-ния.	- ориентироваться в современных информационных технологиях - выполнять верстку web-страниц на языке HTML с применением CSS - применять основы серверных и клиентских скриптовых языков (на примере JavaScript и PHP) для создания интерактивных web-страниц
ПК-2.3. Владеть навыками проведения анализа результатов тестов; ко-ординирования и проведения оценки готовых систем; обучения участников рабочей группы методике оценки готовых систем; оформления отчета о степени соответствия готовых систем требо-ваниям; подготовки методики оценки готовых систем на со-ответствие требованиям; принятия решения о пригодности ар-хитектуры; разработки прототипа ИС в соответствии с требо-ваниями; осуществления сбора, обработки и анализа результа-тов оценки готовых систем на соответствие требованиям; согла-сования пользовательского интерфейса с заказчиком; тести-рования прототипа ИС на проверку корректности архитектурных решений.	- навыками разработки прототипа ИС в соответствии с требованиями средствами Web-дизайна и Internet-программирования.

3. Содержание, объем дисциплины и формы проведения занятий

№ п/п	Наименование темы дисциплины	Компетенции	Оценочные средства текущего контроля		
			ЗНАТЬ	УМЕТЬ	ВЛАДЕТЬ
			ПК-2.1	ПК-2.2	ПК-2.3
1	Разработка статического информационного контента.	ПК-2	Собеседование, опрос/ Контрольная работа №1 (10)	Задача №1 (10) Задача №2 (10) Задача №3 (10)	Расчетно-графическая работа №1 (20)
2	Разработка динамического контента на встроенных языках.	ПК-2	Собеседование, опрос/ Контрольная работа №2 (10)	Задача №4 (10) Задача №5 (10)	Расчетно-графическая работа №1 (20)
3	Разработка динамического контента с помощью CMS.	ПК-2	Доклад, сообщение/ Реферат №1 (10)	Расчетно-графическая работа №1 (20)	Расчетно-графическая работа №1 (20)
Количество баллов (100 баллов):			100		

09.03.03 Прикладная информатика, направленность (профиль) "Прикладная информатика в экономике"
 Рабочая программа дисциплины
 Дисциплина: Б1.В.ДВ.02.01 Интернет-программирование
 Форма обучения: очная, заочная
 Разработана для приема 2023/2024 учебного года

Содержание учебного материала, лабораторные работы и практические занятия, курсовая работа
<p>Тема 1: Разработка статического информационного контента. Принципы гипертекстовой разметки. URL – универсальный локатор ресурса. Описание языка HTML. CSS – каскадные таблицы стилей. Псевдо классы и псевдо элементы. CSS – элементы. Практические занятия/самостоятельная работа: Разработка web сайта на языке HTML с использованием CSS. Форматирование страниц CSS таблицами. Лабораторная работа: -</p>
<p>Тема 2: Разработка динамического контента на встроенных языках. Введение в JavaScript. Объектная модель JavaScript. Основы синтаксиса JavaScript. Объекты в JavaScript. DOM-модель и работа с элементами web-страницы. Практические занятия/самостоятельная работа: Программирование функций и событий на JavaScript. Использование JavaScript для доступа и управления HTML DOM объектов. Лабораторная работа: -</p>
<p>Тема 3: Разработка динамического контента с помощью CMS. Системы управления контентом: (Content management system, CMS) Joomla, uCoz, WordPress. Сравнительные характеристики. Разработка сайта в WordPress. Разработка сайта в uCoz. Создание веб сайтов с Joomla. Практические занятия/самостоятельная работа: Разработка мультимедийного портала в WordPress. Разработка новостного портала в uCoz. Создание сайта фотостудии на Joomla. Лабораторная работа: -</p>
<p>Курсовая работа: не предусмотрено учебным планом</p>

Очная форма обучения

Вид учебной работы	Всего часов	Семестр 5
Аудиторные занятия (АЗ):	54	54
Лекционные занятия (Лек)	18	18
Лабораторные занятия (Лаб)	0	0
Практические занятия (Пр)	36	36
Самостоятельная работа студента (СР)	82	82
Курсовая работа	0	0
Другие виды самостоятельной работы*	82	82
Контроль самостоятельной работы (КСР)	8	8
Контактная работа (КоР)	62	62
Форма промежуточной аттестации	0	Экзамен
Подготовка к экзамену и сдача экзамена (СР, КоР)	36	36
Общая трудоемкость дисциплины, часы/ЗЕТ	180/5	180/5

* Подготовка к аудиторным занятиям, подготовка к зачету (при наличии)

№	Наименование темы дисциплины	Семестр/Курс	Количество учебных часов				СР	Практическая подготовка
			В том числе по видам аудиторных занятий					
			Лек	Пр	Лаб			
1	Разработка статического информационного контента.	5	6	12	0	20	12	
2	Разработка динамического контента на встроенных языках.	5	6	14	0	22	14	
3	Разработка динамического контента с помощью CMS.	5	6	10	0	40	10	
Итого:			18	36	0	82	36	

* Практическая подготовка при реализации дисциплин организована путем проведения практических занятий и (или) выполнения лабораторных и (или) курсовых работ и предусматривает выполнение работ, связанных с будущей профессиональной деятельностью.

Заочная форма обучения

Вид учебной работы	Всего часов	Семестр 6
Аудиторные занятия (АЗ):	12	12

Вид учебной работы	Всего часов	Семестр 6
Лекционные занятия (Лек)	6	6
Лабораторные занятия (Лаб)	0	0
Практические занятия (Пр)	6	6
Самостоятельная работа студента (СР)	151	151
Курсовая работа	0	0
Другие виды самостоятельной работы*	151	151
Контроль самостоятельной работы (КСР)	8	8
Контактная работа (КоР)	20	20
Форма промежуточной аттестации	0	Экзамен
Подготовка к экзамену/зачету и сдача экзамена/зачета (СР, КоР)	9	9
Общая трудоемкость дисциплины, часы/ЗЕТ	180/5	180/5

* Подготовка к аудиторным занятиям

№	Наименование темы дисциплины	Семестр/ Курс	Количество учебных часов				Практическая подготовка
			В том числе по видам аудиторных занятий			СР	
			Лек	Пр	Лаб		
1	Разработка статического информационного контента.	6	2	2	0	40	12
2	Разработка динамического контента на встроенных языках.	6	2	2	0	50	14
3	Разработка динамического контента с помощью CMS.	6	2	2	0	61	10
Итого:			6	6	0	151	36

* Практическая подготовка при реализации дисциплин организована путем проведения практических занятий и (или) выполнения лабораторных и (или) курсовых работ и (или) путем выделения часов из часов, отведенных на самостоятельную работу, и предусматривает выполнение работ, связанных с будущей профессиональной деятельностью.

4. Способ реализации дисциплины

Без использования онлайн-курса.

5. Учебно-методическое обеспечение дисциплины:

Основная литература:

1. ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА WEB-ПРИЛОЖЕНИЙ. Учебное пособие для вузов / Тузовский А. Ф. - Национальный исследовательский Томский политехнический университет (г. Томск), 2022 г. - 218 с. - ISBN 978-5-534-00515-8 – Режим доступа: <https://urait.ru/book/proektirovanie-i-razrabotka-web-prilozheniy-490128>

2. Янцев В. В. — JavaScript. Как писать программы - Издательство Лань, 2022 г. - 200 с. - ISBN 978-5-8114-8559-8 – Режим доступа: <https://e.lanbook.com/book/197547>

3. ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ ВЫСОКОГО УРОВНЯ PYTHON 3-е изд., пер. и доп. Учебное пособие для вузов / Федоров Д. Ю. - Санкт-Петербургский государственный экономический университет (г. Санкт-Петербург), 2022 г. - 210 с. - ISBN 978-5-534-14638-7 – Режим доступа: <https://urait.ru/book/programmirovanie-na-yazyke-vysokogo-urovnya-python-492920>

Дополнительная литература:

1. РАЗРАБОТКА ИНТЕРНЕТ-ПРИЛОЖЕНИЙ. Учебное пособие для вузов / Сысолетин Е. Г., Ростунцев С. Д. ; под науч. ред. Доросинского Л.Г. - Уральский федеральный университет имени первого Президента России Б.Н. Ельцина (г. Екатеринбург), 2022 г. - 90 с. - ISBN 978-5-9916-9975-4 – Режим доступа: <https://urait.ru/book/razrabotka-internet-prilozheniy-492224>

2. ОСНОВЫ ПРОГРАММИРОВАНИЯ НА PYTHON. Учебное пособие для вузов / Чернышев С. А. - Санкт-Петербургский государственный университет промышленных технологий и дизайна (г. Санкт-Петербург); Санкт-Петербургский государственный

3. РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЙ. Учебное пособие для вузов / Полуэктова Н. Р. - Крымский федеральный университет имени В.И. Вернадского (г. Симферополь)., 2022 г. - 204 с. - ISBN 978-5-534-13715-6 – Режим доступа: <https://urait.ru/book/razrabotka-veb-prilozheniy-496682>

6. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения

1. Операционная система
2. Пакет прикладных офисных программ
3. Антивирусное программное обеспечение
4. LMS Moodle
5. Вебинарная платформа
6. WordPress

7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», информационных справочных систем и профессиональных баз данных, необходимых для освоения дисциплины

1. ibooks.ru : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <https://ibooks.ru>. - Текст: электронный

2. Электронно-библиотечная система СПбУТиЭ : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <http://libume.ru>. - Текст: электронный

3. Юрайт : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <https://urait.ru>. - Текст: электронный

4. [eLibrary.ru](http://elibrary.ru) : научная электронная библиотека [Электронный ресурс] : профессиональная база данных. - Режим доступа: <http://elibrary.ru>. - Текст: электронный

5. Архив научных журналов НЭИКОН [Электронный ресурс] : профессиональная база данных. - Режим доступа: arhiv.neicon.ru. - Текст: электронный

6. КиберЛенинка : научная электронная библиотека [Электронный ресурс] : информационная справочная система. - Режим доступа: <http://cyberleninka.ru>. - Текст: электронный

7. Лань : электронно-библиотечная система [Электронный ресурс] : профессиональная база данных. - Режим доступа: <https://e.lanbook.com>. - Текст: электронный

8. [it-world.ru](https://www.it-world.ru) [Электронный ресурс] : информационная справочная система . - Режим доступа: <https://www.it-world.ru>. - Текст: электронный

9. Connect: IT-технологии : информационная справочная система. - Режим доступа: <https://www.connect-wit.ru/>. - Текст: электронный

10. Виртуальный компьютерный музей [Электронный ресурс] : информационная справочная система . - Режим доступа: <https://www.computer-museum.ru>. - Текст: электронный

11. Компьютерра : информационная справочная система . - Режим доступа: <https://www.computerra.ru/>. - Текст: электронный

12. Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации: профессиональная база данных. - Режим доступа: <https://digital.gov.ru>. - Текст: электронный

13. Федеральная служба по надзору в сфере связи, информационных технологий и массовых коммуникаций: профессиональная база данных . - Режим доступа: <https://rkn.gov.ru>. - Текст: электронный

14. [Math-Net.Ru](https://www.mathnet.ru/): профессиональная база данных . - Режим доступа: <https://www.mathnet.ru/>. - Текст: электронный

8. Материально-техническое обеспечение дисциплины

1. Учебная аудитория для проведения занятий лекционного типа, семинарского типа - практических занятий, для групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оборудованная: рабочими местами для обучающихся, оснащенной специальной мебелью; рабочим местом преподавателя, оснащенной специальной мебелью, персональным компьютером с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета; техническими средствами обучения - мультимедийным оборудованием (проектор, экран, колонки) и маркерной доской; лицензионным программным обеспечением

2. Учебная аудитория для проведения занятий семинарского типа - практических занятий – компьютерный класс, оборудованный рабочими местами для обучающихся, оснащенный специальной мебелью, персональными компьютерами с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета; рабочим местом преподавателя, оснащенный специальной мебелью, персональным компьютером с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета; техническими средствами обучения - мультимедийным оборудованием (проектор, экран, колонки) и маркерной доской; лицензионным программным обеспечением

3. При применении электронного обучения, дистанционных образовательных технологий используются: виртуальные аналоги учебных аудиторий - вебинарные комнаты на вебинарных платформах, рабочее место преподавателя, оснащенное персональным компьютером (планшет, мобильное устройство) с возможностью подключения к сети «Интернет», доступом к электронной информационно-образовательной среде Университета и к информационно-образовательному portalу Университета imeos.ru, веб-камерой, микрофоном и гарнитурой (в т.ч. интегрированными в устройствами), программным обеспечением; рабочее место обучающегося оснащено персональным компьютером (планшет, мобильное устройство) с возможностью подключения к сети «Интернет», доступом к электронной информационно-образовательной среде Университета и к информационно-образовательному portalу Университета imeos.ru, веб-камерой, микрофоном и гарнитурой (в т.ч. интегрированными в устройства). Авторизация на информационно-образовательном portalе Университета imeos.ru и начало работы осуществляются с использованием персональной учетной записи (логина и пароля). Лицензионное программное обеспечение

4. Помещение для самостоятельной работы, оборудованное специальной мебелью, персональными компьютерами с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде Университета, лицензионным программным обеспечением

9. Оценочные материалы по дисциплине

Описание оценочных средств (показатели и критерии оценивания, шкалы оценивания) представлено в приложении к основной профессиональной образовательной программе «Каталог оценочных средств текущего контроля и промежуточной аттестации».

Процедуры оценивания знаний, умений, навыков и (или) опыта деятельности приводятся в соответствующих методических материалах и локальных нормативных актах Университета.

Для оценивания учебных достижений студентов в Университете действует балльно-рейтинговая система.

Если оценка, соответствующая набранной в семестре сумме рейтинговых баллов, удовлетворяет студента, то она является итоговой оценкой по дисциплине при проведении промежуточной аттестации в форме экзамена/зачета с оценкой/зачета.

Условием сдачи экзамена/зачета с оценкой/зачета с целью повышения итоговой оценки по дисциплине является сдача студентом экзамена, за который он получает экзаменационные баллы без учета баллов, полученных за текущий контроль:

Шкала оценивания учебных достижений по дисциплине, завершающейся зачетом без оценки

Баллы по дисциплине	60 и менее		61-73		74-90		91-100
Итоговая оценка по дисциплине	Незачет		Зачет				
Баллы в международной шкале ECTS с буквенным обозначением уровня	50 и менее	51-60	61-67	68-73	74-83	84-90	91-100
	F	Fx	E	D	C	B	A
Уровень сформированности компетенций	Не сформированы		Пороговый		Высокий		Повышенный

Шкала оценивания учебных достижений по дисциплине, завершающейся экзаменом/зачетом с оценкой

Баллы по дисциплине	60 и менее		61-73		74-90		91-100
Итоговая оценка по дисциплине	Неудовлетворительно		Удовлетворительно		Хорошо		Отлично
Баллы в международной шкале ECTS с буквенным обозначением уровня	<50	51-60	61-67	68-73	74-83	84-90	91-100
	F	Fx	E	D	C	B	A
Уровень сформированности компетенций	Не сформированы		Пороговый		Высокий		Повышенный

9.1. Типовые контрольные задания для текущего контроля

Задача №1

Решение задач на основы верстки на языке HTML

1. Создайте html-документ с текстом согласно образцу:

Конфуций

Конфуций — древнекитайский мыслитель, этико-философское учение которого оказало огромное влияние на жизнь и культурное развитие Дальнего Востока. Его философия строится на морали, этике и жизненных принципах взаимоотношений между людьми. На ее основе появилась концепция, господствовавшая в Поднебесной более двух тысяч лет (*и продолжающая оказывать свое влияние и в наши дни*).

Главной книгой конфуцианства является «*Лунь Юй*» («Суждения и беседы») — записки и краткие заметки учеников Конфуция, в которых зафиксированы высказывания, поступки и диалоги их учителя.

Афоризмы великого мудреца

- Драгоценный камень нельзя отполировать без трения. Также и человек не может стать успешным без достаточного количества трудных попыток.
- Мудрый человек не делает другим того, чего он не желает, чтобы ему сделали.
- Не беспокойся о том, что тебя не знают. Беспокойся о том, достоин ли ты того, чтобы тебя знали.
- Порой мы видим многое, но не замечаем главного.
Учитесь так, словно вы постоянно ощущаете нехватку своих знаний, и так, словно вы постоянно боитесь растерять свои знания.
- Когда, совершив ошибку, не исправил ее, это и называется совершить ошибку.
- Учение без размышления бесполезно, но и размышление без учения опасно.

2. Создайте html-документ с текстом согласно образцу:

Основные цвета радуги

- Красный
- Оранжевый
- Желтый
- Зеленый
- Голубой
- Синий
- Фиолетовый

Три чуда по Геродоту

1. Акведук в виде тоннеля
2. Дамба в порту на острове
3. Храм богини Геры

Семь чудес света античного мира

1. Пирамида Хеопса
 - Гиза
 - 2550 г. до н. э.
2. Висячие сады Семирамиды
 - Вавилон
 - 600 г. до н. э.
3. Статуя Зевса в Олимпии
 - Олимпия
 - 435 г. до н. э.
4. Храм Артемиды Эфесской
 - Эфес
 - 550 г. до н. э.
5. Мавзолей в Галикарнасе
 - Галикарнас
 - 351 г. до н. э.
6. Колосс Родосский
 - Родос
 - между 292 и 280 гг. до н. э.
7. Александрийский маяк
 - Александрия
 - III век до н. э.

3. Создайте html-документ с таблицей согласно образцу:

Заглавие таблицы							
Заголовок 1	Заголовок 2		Заголовок 3				
Текст 1	<table border="1"><tr><td>1</td><td>2</td></tr><tr><td>3</td><td>4</td></tr></table>		1	2	3	4	Текст 2
	1	2					
	3	4					
		Текст 3					
Текст 4	Текст 5	Текст 6	Текст 7				

4. Создайте три html-страницы и свяжите их ссылками друг с другом:

Страница №1

[Ссылка на вторую страницу](#)

Страница №2

[Ссылка на третью страницу](#)

Страница №3

[Ссылка на первую страницу](#)

Задача №2

Решение задач на основы форматирования с помощью CSS

1. Создайте с использованием CSS html-документ, содержащий три абзаца с произвольным текстом:

- Первый абзац: цвет текста - синий, шрифт - Arial, отступ снизу - 20px; CSS код находится в отдельном файле и подключен к html-странице через элемент link.
- Второй абзац: цвет текста - черный, фон - светло-серый, шрифт - Times New Roman, размер текста на 20% больше, чем в первом абзаце; CSS код размещен в теге .
- Третий абзац: цвет текста - зеленый, стиль текста - жирный курсив, отступ слева - 30px; CSS

код размещен внутри атрибута style тега "p".

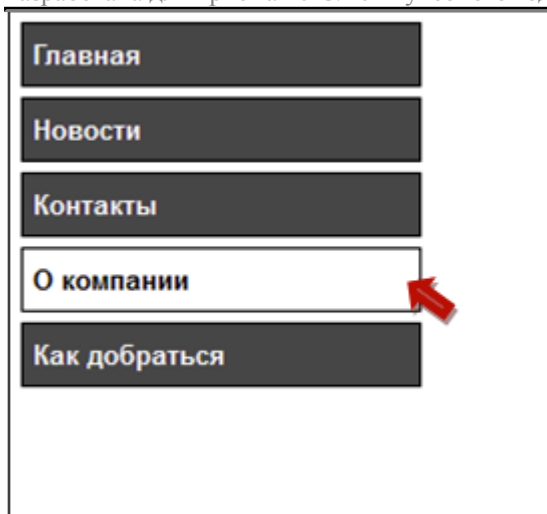
2. Подготовьте файл стилей CSS для определения следующих свойств оформления html-страницы (затем создайте html-страницу, к которой подключите разработанный файл стилей, и продемонстрируйте работу с ним на произвольном тексте):

- Сделайте все абзацы "p" красного цвета.
- Сделайте все "h1" голубого цвета.
- Сделайте все абзацы "h2" шириной 300px.
- Сделайте все таблицы шириной 400px, высотой 200px.
- Поставьте все "h1" по центру.
- Поставьте все "td" по центру.
- Сделайте все "td" жирным.
- Сделайте одновременно "th", "h1" и "h2" нежирным.
- Сделайте размер текста 15px для всех абзацев "p".
- Сделайте для "h3" любой шрифт без засечек.
- Сделайте межстрочный интервал для абзацев "p" в 30px.
- Для "h2" сделайте следующий шрифт: нежирный, 20 пикселей, Verdana.
- Сделайте красную строку в абзацах 30px.
- Поставьте текст в таблице по верхнему краю по вертикали.
- Поставьте текст в таблице по центру по вертикали.

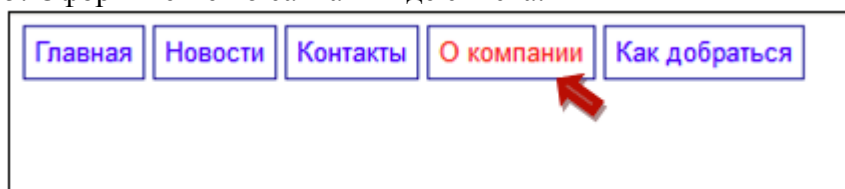
3. Придумайте селектор для определения элемента, к которому требуется применить CSS-правило:

- все абзацы "p" внутри дивов "div".
- все "h2" внутри дивов "div".
- все абзацы "p" из элемента с id="type1".
- все "h2" из элемента с id="type1".
- все элементы с классом xxx.
- все абзацы "p" с классом xxx.
- все "h2" с классом xxx.
- все элементы с классом xxx из элемента с id="type1".
- все абзацы "p" с классом xxx из элемента с id="type1".
- все элементы с классом xxx и элементы с классом ууу одновременно.
- все абзацы "p" с классом xxx и "h2" с классом ууу одновременно.
- все абзацы "p" с xxx из id="type1" и все абзацы "p" с классом ууу из id="type1" одновременно.
- все элементы из класса hhh.
- все абзацы "p" из класса hhh.
- все элементы с классом xxx из класса hhh.
- все "h2" с классом xxx из класса hhh.

4. Оформите меню сайта в виде списка :



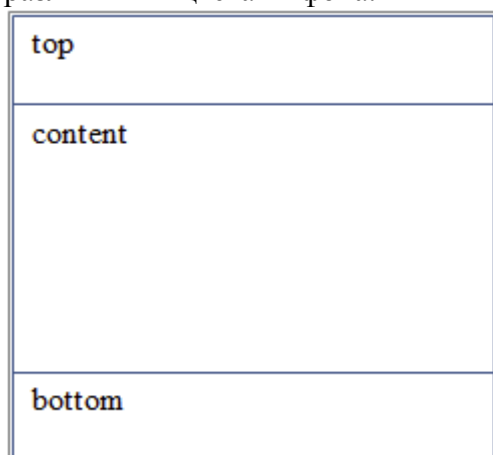
5. Оформите меню сайта в виде списка:



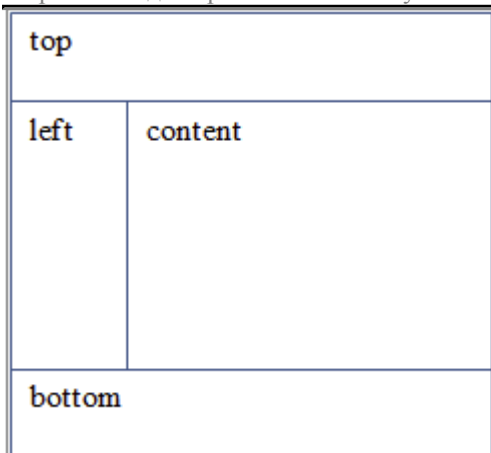
Задача №3

Решение задач на основы верстки с помощью CSS

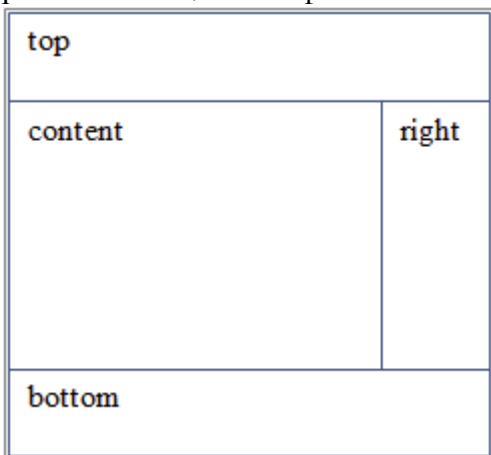
1. Создайте макет web-страницы, расположив блоки согласно образцу. Блоки выделите различными цветами фона.



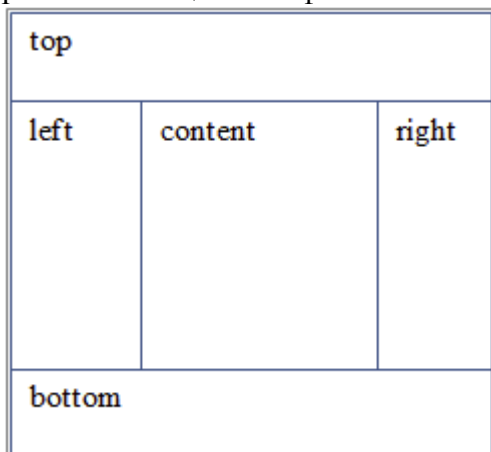
2. Создайте макет web-страницы, расположив блоки согласно образцу. Блоки выделите различными цветами фона.



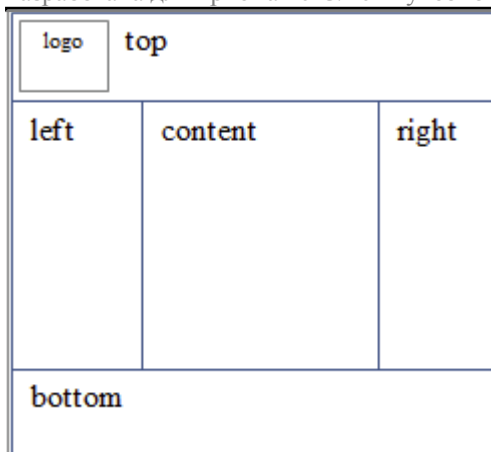
3. Создайте макет web-страницы, расположив блоки согласно образцу. Блоки выделите различными цветами фона.



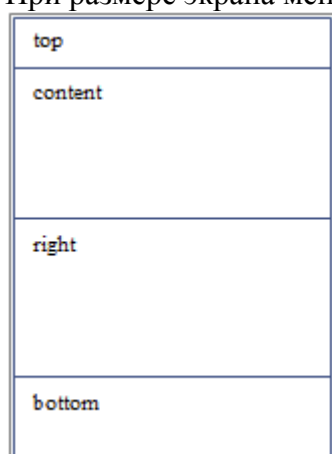
4. Создайте макет web-страницы, расположив блоки согласно образцу. Блоки выделите различными цветами фона.



5. Создайте макет web-страницы, расположив блоки согласно образцу. Блоки выделите различными цветами фона.
При размере экрана больше 800px:



При размере экрана менее 800px:



5. Выполните проверку предложенной web-страницы на соответствие стандарту (в соответствии с записью !DOCTYPE). Служба валидации W3C доступна по адресу: <http://validator.w3.org>. Внесите исправления, чтобы код соответствовал стандарту.

Задача №4

Решение задач на основы языка JavaScript

1. Напишите скрипт, который считает количество секунд в сутках.
2. Создайте переменные `str1='Привет, '` и `str2='Мир!'`. С помощью этих переменных и операции сложения строк выведите на экран фразу 'Привет, Мир!'
3. Спросите имя пользователя с помощью метода `prompt`. Выведите с помощью `alert` сообщение 'Ваше имя %имя%'
4. Дан массив `['Привет, ', 'мир', '!']`. Запишите в нулевой элемент этого массива слово 'Здравствуй, '. Запишите в переменную `text` фразу, составленную из элементов массива, затем выведите на экран содержимое этой переменной.
5. Создайте объект с днями недели. Ключами в нем должны служить номера дней от начала недели (понедельник - первый и т.д.). Выведите на экран текущий день недели.
6. Создайте двухмерный массив. Первые два ключа - это 'ru' и 'en'. Пусть первый ключ содержит элемент, являющийся массивом названий дней недели "по-русски", а второй - "по-английски". Выведите с помощью этого массива понедельник по-русски и среду по английски (пусть понедельник - это нулевой день).
7. Дана строка из 6-ти цифр. Проверьте, что сумма первых трех цифр равняется сумме вторых трех цифр. Если это так - выведите 'да', в противном случае выведите 'нет'.
8. Составьте массив дней недели. С помощью цикла `for` выведите все дни недели, а текущий день выведите курсивом. Текущий день должен храниться в переменной `day`.
9. Дан объект `obj` с ключами 'Париж', 'Москва', 'Пекин' с элементами 'Франция', 'Россия', 'Китай'.

С помощью цикла `for-in` выведите на экран строки такого формата: 'Париж - столица Франции'.

10. Преобразуйте строку `'var_test_text'` в `'varTestText'`. Скрипт должен работать с любыми аналогичными строками.

11. Имеются кнопка и два инпута, в которых может быть размещен произвольный текст. По нажатию на кнопку требуется выполнить обмен содержимым между двумя инпутами.

12. Имеются кнопка и два инпута. По нажатию на кнопку во втором инпуте должен появиться квадрат числа из первого инпута. Если введено не число - должно быть выведено сообщение об ошибке.

13. Повторите страницу по образцу:

+ = ?

Нажмите чтобы сложить

14. Создайте тикающие часы:

13:24:16

15. Даны дивы. По первому нажатию на каждый див он красится красным фоном, по второму красится обратно и так далее каждый клик происходит чередование фона. Сделайте так, чтобы было две функции: одна красит в красный цвет, другая в зеленый и они сменяли друг друга через `removeEventListener`.

16. Дан элемент `ul`, а в нем `li #elem`. Вставьте перед элементом `#elem` новую `li` с текстом '!!!'.

17. Дан див красного цвета. По нажатию на красный див ниже должен появиться еще один див зеленого цвета, а исходный див должен поменять цвет на синий. По нажатию на синий див зеленый должен исчезнуть с экрана.

18. Дан селект, инпут и кнопка. Сделайте так, чтобы в инпут можно было ввести число, нажать на кнопку и в селекте становился выбранным пункт с этим номером.

19. Дан элемент `#elem` с границами. По нажатию на кнопку выведите его полную ширину с учетом границы и `padding`.

20. По нажатию на кнопку прокрутите страницу на 400px от текущего положения.

Задача №5

Решение задач на основы языка PHP

1. Создайте программу на PHP, позволяющую выполнить следующие задания:

- печать таблицы умножения (для чисел от 1 до 9);

- печать таблицы квадратов (для чисел от 1 до 20).

2. Напишите программу обработки заявки на обучение выбранным курсам. Программа должна реализовать следующие функции:

- прием данных из формы;

- формирование текста: "Уважаемый(ая) <Фамилия> <Имя>. Вы зачислены на курсы по изучению <наименование(я) курса>." Выведите сформированный текст ниже заявки.

3. Напишите программу на PHP, которая принимает из формы HTML-код и выводит на экран:

- встречающиеся теги;

- текст без указания тегов.

4. Напишите на PHP программу обработки массива. Организуйте ввод массива через форму. Напишите сценарий, удаляющий из упорядоченного массива элемент, значение которого совпадает с заданным. Если таких элементов несколько, то требуется удалить элемент с наименьшим индексом.

5. Напишите на PHP программу создания календаря. По заданному году и месяцу на экран выводится календарь этого месяца.

6. В тексте идет описание единиц измерений. Единица измерения и пояснение к ней разделяются символом «-». Преобразовать этот текст в таблицу из двух столбцов: «единица измерения» и «пояснение».

Доклад, сообщение/Реферат №1

1. Обзор CMS для интернет-магазина
2. Обзор бесплатных CMS
3. CMS WordPress
4. CMS Joomla
5. CMS OpenCart
6. CMS Drupal
7. CMS 1С-Битрикс
8. CMS Magento
9. CMS TYPO3
10. CMS Made Simple

Собеседование, опрос/Контрольная работа №1

- Принципы гипертекстовой разметки.
- Структура документов. Примеры простой гипертекстовой разметки.
- Контейнеры заголовка документа.
- Контейнер тела документа: заголовки, параграфы, переводы строк, отчеркивания, управлением форматированием. Атрибуты.
- Гипертекстовые ссылки. Ссылки на графику и почту. Закладки.
- Специальные символы HTML-документа.
- Графика в HTML. Форматы графических файлов. Активные изображения. Изображения в миниатюре.
- Создание списков: нумерованные, маркированные, смешанные и вложенные списки.
- HTML-таблицы. Атрибуты строк и ячеек.
- HTML-формы: текст, пароль, переключатель, радиокнопка, отсылка, перезагрузка, файл, скрытый, графический, выбор элемента.
- Назначение CSS. Способы применения. Переопределение стиля.
- Свойства элементов, управляемых с помощью CSS. Ссылка на внешнее описание. Импорт описания стилей.
- Селектор-элемент разметки, селектор-класс, селектор-идентификатор объекта. Наследование и переопределение.
- Блочные и строковые элементы. Набивка, отступ, границы, обтекание.
- Форматирование текста. Управление цветом. Позиционирование. Координаты. Слои.
- Современные стандарты языка HTML.
- Консорциум Всемирной паутины (W3C).
- Понятие кроссбраузерной верстки.
- Понятие адаптивной верстки.

Собеседование, опрос/Контрольная работа №2

- JavaScript: объектная модель, задание сценария, примеры реализации.
- Базовые основы языка JavaScript.
- Типы данных JavaScript. Область действия переменных.
- Управление потоком вычислений в JavaScript. Функции. Объекты.
- PHP: история развития, возможности, области применения, PHP-скрипты, встраивание в HTML.
- Основы синтаксиса языка PHP. Структура простейшей программы. Переменные. Константы. Предопределенные константы.

- PHP. Элементы структурного программирования (ветвление, организация циклов, передача управления).
- Использование массивов в PHP. Способы создания/удаления массива. Индексные массивы (списки), обращение к элементу массива. Ассоциативные массивы, обращение к элементу массива. Цикл `foreach`.
- Функции в PHP. Синтаксис описания функций. Порядок определения и вызова. Способы передачи аргументов, значения по умолчанию.
- Схема работы Web-приложения, архитектура клиент-сервер. Протокол HTTP. Типы HTTP-соединений. Общая структура HTTP-сообщений (запрос, ответ). Выбор метода: GET или POST.
- Обработка данных HTML-форм. Способы передачи параметров сценарию PHP. Обращение к параметру, переданному из формы.

Расчетно-графическая работа №1

Разработайте сайт компании, занимающейся разработкой садовых инструментов (изданием книг, печатью постеров и т.п.)

Требования:

1. Сайт должен быть разработан на базе CMS.
2. Интегрируйте в CMS один из предложенных готовых шаблонов. Используйте минимум два различных макета страниц.
3. Реализуйте форму обратной связи.

9.2. Примерный перечень тем курсовой работы

Не предусмотрено учебным планом

9.3. Типовые контрольные задания для промежуточной аттестации: экзамен

Примерный перечень теоретических вопросов к экзамену: Вопрос № 1

- Принципы гипертекстовой разметки.
- Структура документов. Примеры простой гипертекстовой разметки.
- Контейнеры заголовка документа.
- Контейнер тела документа: заголовки, параграфы, переводы строк, отчеркивания, управлением форматированием. Атрибуты.
- Гипертекстовые ссылки. Ссылки на графику и почту. Закладки.
- Специальные символы HTML-документа.
- Графика в HTML. Форматы графических файлов. Активные изображения. Изображения в миниатюре.
- Создание списков: нумерованные, маркированные, смешанные и вложенные списки.
- HTML-таблицы. Атрибуты строк и ячеек.
- HTML-формы: текст, пароль, переключатель, радиокнопка, отсылка, перезагрузка, файл, скрытый, графический, выбор элемента.
- Назначение CSS. Способы применения. Переопределение стиля.
- Свойства элементов, управляемых с помощью CSS. Ссылка на внешнее описание. Импорт описания стилей.
- Селектор-элемент разметки, селектор-класс, селектор-идентификатор объекта. Наследование и переопределение.
- Блочные и строковые элементы. Набивка, отступ, границы, обтекание.
- Форматирование текста. Управление цветом. Позиционирование. Координаты. Слои.
- Современные стандарты языка HTML.
- Консорциум Всемирной паутины (W3C).
- Понятие кроссбраузерной верстки.

- Понятие адаптивной верстки.
- JavaScript: объектная модель, задание сценария, примеры реализации.
- Базовые основы языка JavaScript.
- Типы данных JavaScript. Область действия переменных.
- Управление потоком вычислений в JavaScript. Функции. Объекты.
- PHP: история развития, возможности, области применения, PHP-скрипты, встраивание в HTML.
- Основы синтаксиса языка PHP. Структура простейшей программы. Переменные. Константы. Предопределенные константы.
- PHP. Элементы структурного программирования (ветвление, организация циклов, передача управления).
- Использование массивов в PHP. Способы создания/удаления массива. Индексные массивы (списки), обращение к элементу массива. Ассоциативные массивы, обращение к элементу массива. Цикл foreach.
- Функции в PHP. Синтаксис описания функций. Порядок определения и вызова. Способы передачи аргументов, значения по умолчанию.
- Схема работы Web-приложения, архитектура клиент-сервер. Протокол HTTP. Типы HTTP-соединений. Общая структура HTTP-сообщений (запрос, ответ). Выбор метода: GET или POST.
- Обработка данных HTML-форм. Способы передачи параметров сценарию PHP. Обращение к параметру, переданному из формы.
- Системы управления контентом: понятие системы управления контентом, общая структура систем управления контентом.
- Обзор систем управления контентом.

Примерный перечень практических заданий к экзамену: Вопрос № 2

1. Создайте html-документ с текстом согласно образцу:

Конфуций

Конфуций — древнекитайский мыслитель, этико-философское учение которого оказало огромное влияние на жизнь и культурное развитие Дальнего Востока. Его философия строится на морали, этике и жизненных принципах взаимоотношений между людьми. На ее основе появилась концепция, господствовавшая в Поднебесной более двух тысяч лет (*и продолжающая оказывать свое влияние и в наши дни*).

Главной книгой конфуцианства является «*Лунь Юй*» («Суждения и беседы») — записки и краткие заметки учеников Конфуция, в которых зафиксированы высказывания, поступки и диалоги их учителя.

Афоризмы великого мудреца

- Драгоценный камень нельзя отполировать без трения. Также и человек не может стать успешным без достаточного количества трудных попыток.
- Мудрый человек не делает другим того, чего он не желает, чтобы ему сделали.
- Не беспокойся о том, что тебя не знают. Беспокойся о том, достоин ли ты того, чтобы тебя знали.
- Порой мы видим многое, но не замечаем главного.
Учитесь так, словно вы постоянно ощущаете нехватку своих знаний, и так, словно вы постоянно боитесь растерять свои знания.
- Когда, совершив ошибку, не исправил ее, это и называется совершить ошибку.
- Учение без размышления бесполезно, но и размышление без учения опасно.

2. Создайте html-документ с текстом согласно образцу:

Основные цвета радуги

- Красный
- Оранжевый
- Желтый
- Зеленый
- Голубой
- Синий
- Фиолетовый

Три чуда по Геродоту

1. Акведук в виде тоннеля
2. Дамба в порту на острове
3. Храм богини Геры

Семь чудес света античного мира

1. Пирамида Хеопса
 - Гиза
 - 2550 г. до н. э.
2. Висячие сады Семирамиды
 - Вавилон
 - 600 г. до н. э.
3. Статуя Зевса в Олимпии
 - Олимпия
 - 435 г. до н. э.
4. Храм Артемиды Эфесской
 - Эфес
 - 550 г. до н. э.
5. Мавзолей в Галикарнасе
 - Галикарнас
 - 351 г. до н. э.
6. Колосс Родосский
 - Родос
 - между 292 и 280 гг. до н. э.
7. Александрийский маяк
 - Александрия
 - III век до н. э.

3. Создайте html-документ с таблицей согласно образцу:

Заглавие таблицы							
Заголовок 1	Заголовок 2		Заголовок 3				
Текст 1	<table border="1"><tr><td>1</td><td>2</td></tr><tr><td>3</td><td>4</td></tr></table>		1	2	3	4	Текст 2
	1	2					
	3	4					
		Текст 3					
Текст 4	Текст 5	Текст 6	Текст 7				

4. Создайте три html-страницы и свяжите их ссылками друг с другом:

Страница №1

[Ссылка на вторую страницу](#)

Страница №2

[Ссылка на третью страницу](#)

Страница №3

[Ссылка на первую страницу](#)

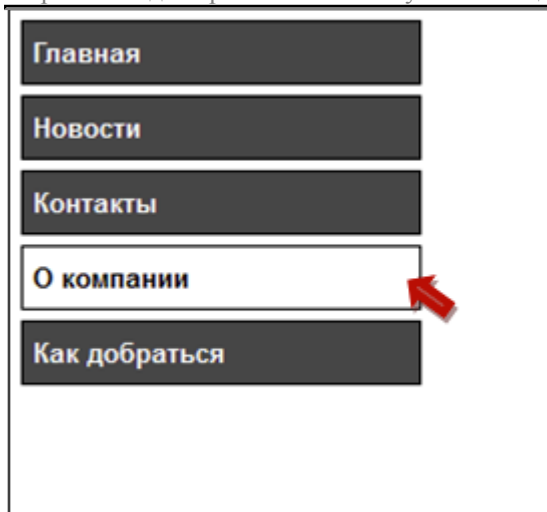
5. Создайте с использованием CSS html-документ, содержащий три абзаца с произвольным текстом:

- Первый абзац: цвет текста - синий, шрифт - Arial, отступ снизу - 20px; CSS код находится в отдельном файле и подключен к html-странице через элемент link.

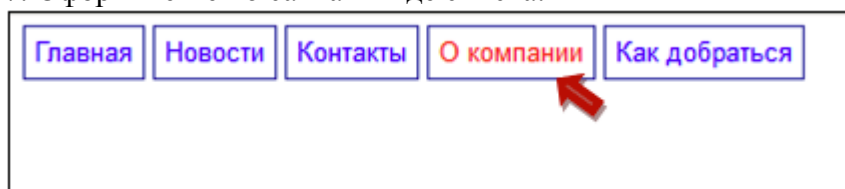
- Второй абзац: цвет текста - черный, фон - светло-серый, шрифт - Times New Roman, размер текста на 20% больше, чем в первом абзаце; CSS код размещен в теге .

- Третий абзац: цвет текста - зеленый, стиль текста - жирный курсив, отступ слева - 30px; CSS код размещен внутри атрибута style тега "p".

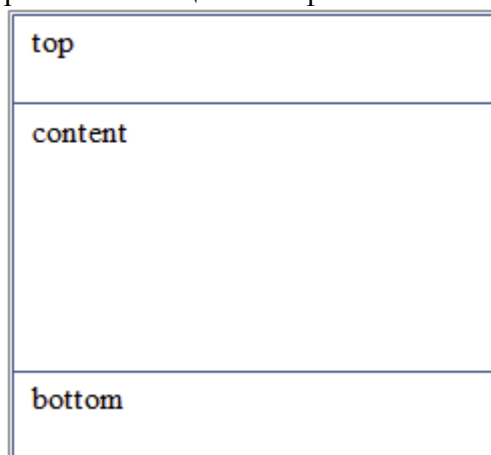
6. Оформите меню сайта в виде списка :



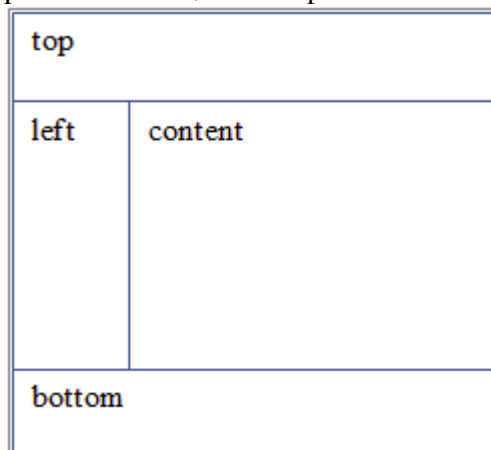
7. Оформите меню сайта в виде списка:



8. Создайте макет web-страницы, расположив блоки согласно образцу. Блоки выделите различными цветами фона.

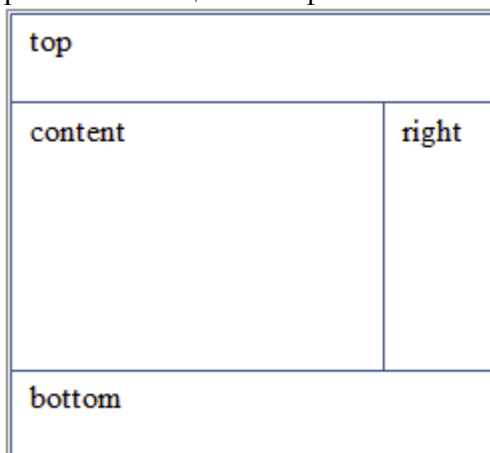


9. Создайте макет web-страницы, расположив блоки согласно образцу. Блоки выделите различными цветами фона.

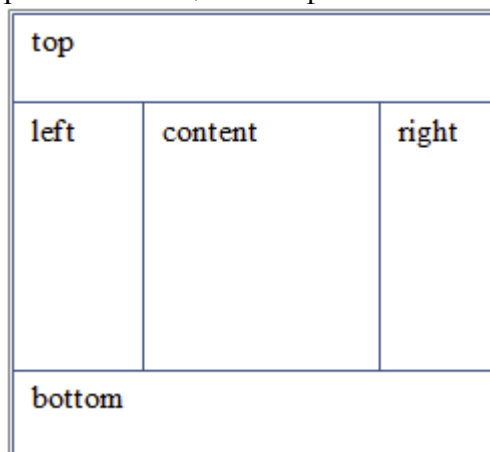


10. Создайте макет web-страницы, расположив блоки согласно образцу. Блоки выделите

различными цветами фона.

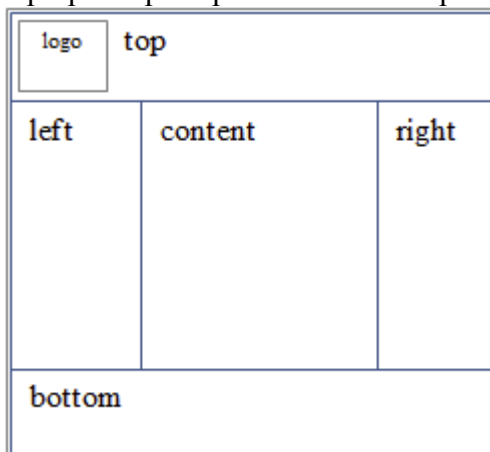


11. Создайте макет web-страницы, расположив блоки согласно образцу. Блоки выделите различными цветами фона.



12. Создайте макет web-страницы, расположив блоки согласно образцу. Блоки выделите различными цветами фона.

При размере экрана больше 800px:



При размере экрана менее 800px:

top
content
right
bottom

13. Напишите скрипт, который считает количество секунд в сутках.
14. Создайте переменные `str1='Привет, '` и `str2='Мир!'`. С помощью этих переменных и операции сложения строк выведите на экран фразу 'Привет, Мир!'.
15. Спросите имя пользователя с помощью метода `prompt`. Выведите с помощью `alert` сообщение 'Ваше имя %имя%'.
16. Дан массив ['Привет, ', 'мир', '!']. Запишите в нулевой элемент этого массива слово 'Здравствуй, '. Запишите в переменную `text` фразу, составленную из элементов массива, затем выведите на экран содержимое этой переменной.
17. Создайте объект с днями недели. Ключами в нем должны служить номера дней от начала недели (понедельник - первый и т.д.). Выведите на экран текущий день недели.
18. Создайте двухмерный массив. Первые два ключа - это 'ru' и 'en'. Пусть первый ключ содержит элемент, являющийся массивом названий дней недели "по-русски", а второй - "по-английски". Выведите с помощью этого массива понедельник по-русски и среду по английски (пусть понедельник - это нулевой день).
19. Дана строка из 6-ти цифр. Проверьте, что сумма первых трех цифр равняется сумме вторых трех цифр. Если это так - выведите 'да', в противном случае выведите 'нет'.
20. Составьте массив дней недели. С помощью цикла `for` выведите все дни недели, а текущий день выведите курсивом. Текущий день должен храниться в переменной `day`.
21. Дан объект `obj` с ключами 'Париж', 'Москва', 'Пекин' с элементами 'Франция', 'Россия', 'Китай'. С помощью цикла `for-in` выведите на экран строки такого формата: 'Париж - столица Франции'.
22. Преобразуйте строку `'var_test_text'` в `'varTestText'`. Скрипт должен работать с любыми аналогичными строками.
23. Имеются кнопка и два инпута, в которых может быть размещен произвольный текст. По нажатию на кнопку требуется выполнить обмен содержимым между двумя инпутами.
24. Имеются кнопка и два инпута. По нажатию на кнопку во втором инпуте должен появиться квадрат числа из первого инпута. Если введено не число - должно быть выведено сообщение об ошибке.
25. Повторите страницу по образцу:

+ = ?

26. Даны дивы. По первому нажатию на каждый див он красится красным фоном, по второму красится обратно и так далее каждый клик происходит чередование фона. Сделайте так, чтобы было две функции: одна красит в красный цвет, другая в зеленый и они сменяли друг друга через `removeEventListener`.
27. Дан див красного цвета. По нажатию на красный див ниже должен появиться еще один див зеленого цвета, а исходный див должен поменять цвет на синий. По нажатию на синий див

зеленый должен исчезнуть с экрана.

28. Дан селект. Дан инпут. Дана кнопка. Сделайте так, чтобы в инпут можно было ввести число, нажать на кнопку и в селекте становился выбранным пункт с этим номером.

29. Дан элемент #elem с границами. По нажатию на кнопку выведите его полную ширину с учетом границы и padding.

30. Напишите программу на PHP обработки заявки на обучение выбранным курсам. Программа должна реализовать следующие функции:

- прием данных из формы;

- формирование текста: "Уважаемый(ая) <Фамилия> <Имя>. Вы зачислены на курсы по изучению <наименование(я) курса>." Выведите сформированный текст ниже заявки.

31. Напишите программу на PHP, которая принимает из формы HTML-код и выводит на экран:

- встречающиеся теги;

- текст без указания тегов.

32. Напишите на PHP программу обработки массива. Организуйте ввод массива через форму. Напишите сценарий, удаляющий из упорядоченного массива элемент, значение которого совпадает с заданным. Если таких элементов несколько, то требуется удалить элемент с наименьшим индексом.

33. Напишите на PHP программу создания календаря. По заданному году и месяцу на экран выводится календарь этого месяца.

34. В тексте идет описание единиц измерений. Единица измерения и пояснение к ней разделяются символом «-». Преобразовать этот текст в таблицу из двух столбцов: «единица измерения» и «пояснение».

Раздел билета	Компетенции	Планируемые результаты обучения по дисциплине	Количество баллов
Вопрос №1 Теоретический вопрос (проверяет знания («знать»), сформированные дисциплиной)	ПК-2	- основы языка гипертекстовой разметки HTML - основы форматирования web-страниц с помощью CSS - принципы валидации web-страниц - основы программирования на стороне клиента на языке JavaScript - основы программирования на стороне сервера на языке PHP	40
Вопрос №2 Практическое задание (проверяет умения («уметь»), проверяет практические навыки («владеть»), сформированные дисциплиной)	ПК-2	- ориентироваться в современных информационных технологиях - выполнять верстку web-страниц на языке HTML с применением CSS - применять основы серверных и клиентских скриптовых языков (на примере JavaScript и PHP) для создания интерактивных web-страниц - навыками разработки прототипа ИС в соответствии с требованиями средствами Web-дизайна и Internet-программирования.	60