Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 22.05.2024 18:13:02
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Факультет информационных технологий

УТВЕРЖДАЮ
Декан факультета
«Информационные технологии»
_____ / Д.Г.Демидов /
«15» февраля 2024г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Веб-технологии»

Направление подготовки/специальность
09.03.03 Прикладная информатика

Профиль/специализация
«Информационные технологии управления бизнесом»

Квалификация
бакалавр

Формы обучения
очная

Москва, 2024 г.

Разработчик(и):

к.т.н., доцент / В.Ю. Верещагин /

Согласовано:
Заведующий кафедрой «Инфокогнитивные технологии»,



к.т.н., доцент / Е.А. Пухова /

Содержание

1 Цели, задачи и планируемые результаты обучения по дисциплине 4
2 Место дисциплины в структуре образовательной программы 5
3 Структура и содержание дисциплины 5
3.1 Виды учебной работы и трудоемкость для очной формы обучения 5
3.2 Тематический план изучения дисциплины для очной формы обучения 5
3.3 Содержание дисциплины 6
3.4 Тематика лабораторных занятий 7
4 Учебно-методическое и информационное обеспечение 8
4.1 Нормативные документы и ГОСТы 8
4.2 Основная литература 8
4.3 Дополнительная литература 9
4.4 Электронные образовательные ресурсы 9
4.5 Лицензионное и свободно распространяемое программное обеспечение 9
4.6 Современные профессиональные базы данных и информационные справочные системы 9
5 Материально-техническое обеспечение 9
6 Методические рекомендации 9
6.1 Методические рекомендации для преподавателя по организации обучения 9
6.2 Методические рекомендации для обучающихся по освоению дисциплины 10
7 Фонд оценочных средств 10
7.1 Методы контроля и оценивания результатов обучения 10
7.2 Шкала и критерии оценивания результатов обучения 11
7.3 Оценочные средства 12

1 Цели, задачи и планируемые результаты обучения по дисциплине

Цель дисциплины заключается в формировании практических навыков и целостного представления о методах и подходах, используемых в веб-разработке, работы с веб-приложениями, изучении программных средств при разработке.

Задачи дисциплины: изучение основ языков разметки HTML и CSS, PHP; изучение технологий разработки сайтов, приложений, чат-ботов; изучение

Обучение по дисциплине «Веб-технологии» направлено на формирование у обучающихся следующих компетенций:

Код и наименование компетенций	Индикаторы достижения компетенции
ПК-4. Способен проводить работы по интеграции программных модулей и компонент и проверку работоспособности выпусков программных продуктов	ИПК-4.1. Знает: методы описания алгоритмов, основные абстрактные типы данных и их реализации; основные способы верстки web-страниц, современные языки разметки; современные системы управления контентом сайта (CMS); основы создания интернет-приложений на основе ASP.NET с использование языка программирования C# (в среде Visual Studio); основные команды для платформы 1С. Приемы программирования в 1С; способы интеграции различных модулей. ИПК-4.3. Владеет: навыками использования языка разметки гипертекста с языками программирования; навыками работы в 1С Конфигураторе и программирования на платформе 1С; способностью проводить интеграцию программных модулей, навыками тестирования разработанной программы.
ПК-5. Способен осуществлять концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности	ИПК-5.2. Умеет: формулировать цели, исходя из анализа проблем, потребностей и возможностей; разрабатывать технико-экономическое обоснование; декомпозировать функции на подфункции; алгоритмизировать деятельность; разрабатывать структуры типовых документов; исполнять ручные тесты, проектировать Web-сайты и разрабатывать Internet-приложения; использовать основные приемы web-дизайна. Внедрять графические. Звуковые, анимационные объекты в страницу; формировать интерактивные блоки web-ресурса; разрабатывать модели концептуальной, функциональной и логической архитектуры системы; спроектировать информационную систему для заданного предприятия по заданным

	характеристикам и создать ее на платформе 1С с помощью конфигурирования и программирования. ИПК-5.3. Владеет: навыками концептуального, функционального и логического проектирования; средствами автоматизации проектирования ПО, работы со средствами Internet и Web-технологий для решения задач профессиональной деятельности; навыками проектирования схемы последовательностей, состояний и взаимодействий компонентов системы; навыками работы в 1С Конфигураторе и программирования на платформе 1С.

2 Место дисциплины в структуре образовательной программы

Дисциплина относится к обязательной части Б1 (модули) и междисциплинарно связана с поддерживающими дисциплинами: основы веб-технологий, основы программирования и последующими дисциплинами: веб-разработка.

3 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 2 зачетные единицы (72 часа).

3.1 Виды учебной работы и трудоемкость для очной формы обучения

№ п/п	Вид учебной работы	Количество часов	Семестры	
			2	
1	**Аудиторные занятия**	**36**	36	
	В том числе:			
1.1	Лекции	18	18	
1.2	Семинарские/практические занятия			
1.3	Лабораторные занятия	18	18	
2	**Самостоятельная работа**	**36**	36	
3	**Промежуточная аттестация**			
	Зачет/диф.зачет/Экзамен		Диф.зачет	
	Итого:	**72**	72	

3.2 Тематический план изучения дисциплины для очной формы обучения

№	Разделы/темы дисциплины	Трудоемкость, час		
		Всего	Аудиторная работа	

п/п			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	**Самостоятельная работа**
1	Определение интерфейсов. Виды интерфейсов и методы их построения	8	2		2		4
2	Классификация и критерии качества пользовательского интерфейса	8	2		2		4
3	Основные понятия WEB технологий. Структура HTML документа	8	2		2		4
4	Определения CSS. Каскадирование и классы в CSS.	8	2		2		4
5	Переменные, типы данных, выражения и управляющие конструкции в PHP.	8	2		2		4
6	Функции в PHP.	8	2		2		4
7	Массивы в PHP.	8	2		2		4
8	Суперглобальные массивы.	8	2		2		4
9	Операторы и функции запросов к СУБД MySQL.	8	2		2		4
Итого		**72**	**18**		**18**		**36**

3.3 Содержание дисциплины

1 Определение интерфейсов. Виды интерфейсов и методы их построения. Определение интерфейса. Виды интерфейсов: внутримашинный интерфейс, внешний интерфейс, интерфейс «человек-машина» (интерфейс «человеккомпьютер», пользовательский интерфейс). Машинная и человеческая части пользовательского интерфейса.

2 Классификация и критерии качества пользовательского интерфейса. Классификация пользовательского интерфейса (командный, WIMP – интерфейс и др.). Достоинства и недостатки различных типов интерфейсов. Критерии качества интерфейса: скорость выполнения работы, количество ошибок, скорость обучения, субъективное удовлетворение пользователей.

3 Основные понятия WEB технологий. Структура HTML документа. История web - технологий. Основные этапы развития. Типы браузеров. Верификация (валидация) HTML- сайта. Основные понятия и основы работы web- технологий. Понятия web - сайта, web -сервера, web -адреса, web -страницы, их связь. Структура HTML- страницы. Характеристика и назначение составляющих. Назначение тегов. Блочные и строчные элементы. Атрибуты тегов. Их назначение и примеры использования. Назначение гиперссылок.

4 Определения CSS. Каскадирование и классы в CSS. Определение CSS. Назначение и преимущества каскадных таблиц стилей. Методы связывания. Универсальный селектор. Селекторы типа элемента и класса. Селекторы идентификаторов. Выражения, применяемые в определении селекторов.

Группировка селекторов. Каскадирование и наследование в каскадных таблицах стилей. Отступы и рамки в CSS. Оформление слоев в CSS.

5 Переменные, типы данных, выражения и управляющие конструкции в PHP. Переменные в PHP. Написание имен переменных. Типы переменных. Механизм присвоения типа переменной в PHP. Операторы echo и print. Операции над строковыми переменными. Вывод на экран служебных для языка PHP символов (например, кавычек). Арифметические операции над переменными в PHP. Операция конкатенации строк и переменных.

6 Функции в PHP. Условный оператор языка PHP if- else. Циклы в PHP: с предусловием, с постусловием, со счетчиком.

7 Массивы в PHP. Массивы. Основные понятия. Индексные и ассоциативные массивы. Простые массивы и списки в PHP. Способы записи простых массивов. Вывод содержимого массива с помощью функции print_r(). Ассоциативные массивы. Способы записи ассоциативных массивов. Операции с одномерными массивами. Подсчет количества элементов в массиве. Удаление элемента массива по ключу (индексу), удаление массива целиком. Многомерные массивы. Простой двумерный массив. Способы задания простого двумерного массива. Вывод многомерных массивов.

8 Суперглобальные массивы. Суперглобальные массивы, их описание и содержание. Методы GET и POST передачи параметров сценарию. Суперглобальные массивы $_GET[] и $_POST[]. Суперглобальный массив $_SESSION[]. Открытие сессии. Идентификатор сессии. Передача данных в массив $_SESSION[]. Закрытие (уничтожение) сессии.

9 Операторы и функции запросов к СУБД MySQL. Права и группы привилегий в MySQL. Функция mysqli_query(). Синтаксис, пример применения. Запись результата работы функции mysqli_query() в переменную. Синтаксис оператора SELECT. Синтаксис функции mysqli_fetch_assoc(). Обход массива в цикле foreach. Оператор соединения JOIN. Типы внутреннего соединения таблиц в запросе. Тема раскрывает возможности использования псевдоклассов и псевдоэлементов.

3.4 Тематика лабораторных занятий

Лабораторная работа 1 Разметка главной страницы. Создание шапки сайта.

Лабораторная работа 2 Создание главного (верхнего) меню, разметки для контента и блока айстопперов.

Лабораторная работа 3 Создание блоков навигации, контактов, новостей и информеров.

Лабораторная работа 4 Создание блоков основного контента, авторизации, корзины, выбора по параметрам и подвала (футера).

Лабораторная работа 5 Вёрстка страниц линейного и табличного просмотра каталогов товаров (index-line.html и index-table.html), а также текстовой страницы сайта.

Лабораторная работа 6 Карточка товара.

Лабораторная работа 7 Вёрстка страницы заказа товара (index-zakaz.html)

Лабораторная работа 8 Установка и настройка сервера Open Server. Установка паттерна (шаблона проектирования) MVC

Лабораторная работа 9 Подключение и настройка индексного файла интернет-магазина.

Лабораторная работа 10 Основные (базисные) конструкции языка PHP.

Лабораторная работа 11 Передача параметров контроллеру.

Лабораторная работа 12 Программирование блока категорий товаров.

Лабораторная работа 13 Программирование блока информеров.

Лабораторная работа 14 Программирование блока айстопперов.

Лабораторная работа 15 Программирование блока категорий товаров (табличный вид).

Лабораторная работа 16 Программирование блока категорий товаров (линейный вид).

Лабораторная работа 17 Карточка товара

4 Учебно-методическое и информационное обеспечение

4.1 Нормативные документы и ГОСТы

1. Федеральный закон от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации» (с изменениями и дополнениями);

2. Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.03 Прикладная информатика, утвержденный приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 922.

3. Приказ Министерства образования и науки РФ от 05 апреля 2017 г. № 301 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры;

4. Порядок проведения государственной итоговой аттестации по образовательным программам высшего образования – программам бакалавриата, программам специалитета и программам магистратуры, утвержденный приказом Минобрнауки России от 29 июня 2015 г. № 636;

5. Положение о практической подготовке обучающихся, утвержденное приказом Министерства науки и высшего образования Российской Федерации и Министерства просвещения Российской Федерации от 5 августа 2020 г. № 885/390.

4.2 Основная литература

Верещагин, В.Ю. Создание веб-страниц на стороне клиента: учебно- методическое пособие для студентов бакалавриата, обучающихся по направлению подготовки 09.03.01 «Информатика и вычислительная техника», профиль «Веб-технологии» / В.Ю. Верещагин, М.В. Даньшина. – Москва: Московский Политех, 2023. – 1 CD-R. – Загл. с титул. экрана. – Текст: электронный. ISBN 978-5-2760-2786-9. — URL: https://online.mospolytech.ru/mod/data/view.php?d=127&rid=5281&filter=1 (дата обращения: 18.09.2023)

4.3 Дополнительная литература

Полуэктова, Н. Р. Разработка веб-приложений : учебное пособие для вузов / Н. Р. Полуэктова. — Москва : Издательство Юрайт, 2023. — 204 с. — (Высшее образование). — ISBN 978-5-534-13715-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/519714 (дата обращения: 18.09.2023)

4.4 Электронные образовательные ресурсы

1. https://online.mospolytech.ru/course/view.php?id=10704
https://online.mospolytech.ru/course/view.php?id=1151
https://online.mospolytech.ru/course/view.php?id=7879

4.5 Лицензионное и свободно распространяемое программное обеспечение

1. Visual Studio Code
2. Браузеры Chrome, Edge, Firefox
3. OpenVPN с правами для запуска у студентов
4. FileZilla
5. PuTTy
6. Git
7. Node.js 18
8. Python 3.10
9. Wireshark

4.6 Современные профессиональные базы данных и информационные справочные системы

1. https://doka.guide/
2. https://developer.mozilla.org/ru/
3. https://roadmap.sh/frontend

5 Материально-техническое обеспечение

Для проведения лабораторных работ и самостоятельной работы студентов подходят аудитории, оснащенные компьютерами с программным обеспечением в соответствии со списком в пункте 4.5 и подключенные к интернету.

Число рабочих мест в аудитории должно быть достаточным для обеспечения индивидуальной работы студентов.

Рабочее место преподавателя должно быть оснащено компьютером с подключенным к нему проектором или иным аналогичным по функциональному назначению оборудованием.

6 Методические рекомендации

6.1 Методические рекомендации для преподавателя по организации обучения

1. При подготовке к занятиям следует предварительно проработать материал занятия, предусмотрев его подачу точно в отведенное для этого время занятия. Следует подготовить необходимые материалы – теоретические сведения, задачи и др. При проведении занятия следует контролировать подачу материала и решение заданий с учетом учебного времени, отведенного для занятия.

2. При проверке работ и отчетов следует учитывать не только правильность выполнения заданий, но и оптимальность выбранных методов решения, правильность выполнения всех его шагов.

6.2 Методические рекомендации для обучающихся по освоению дисциплины

Изучение дисциплины осуществляется в строгом соответствии с целевой установкой в тесной взаимосвязи учебным планом. Основой теоретической подготовки студентов являются лекции и самостоятельная работа.

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, готовятся к промежуточной аттестации, а также самостоятельно изучают отдельные темы учебной программы.

На занятиях студентов, в том числе предполагающих практическую деятельность, осуществляется закрепление полученных, в том числе и в процессе самостоятельной работы, знаний. Особое внимание обращается на развитие умений и навыков установления связи положений теории с профессиональной деятельностью будущего специалиста.

Самостоятельная работа осуществляется индивидуально. Контроль самостоятельной работы организуется в двух формах:

- самоконтроль и самооценка студента;
- контроль со стороны преподавателей (текущий и промежуточный).

Текущий контроль осуществляется на аудиторных занятиях.

Критериями оценки результатов самостоятельной работы студента являются:

- уровень освоения студентом учебного материала;
- умения студента использовать теоретические знания при выполнении практических задач;
- сформированность компетенций;
- оформление материала в соответствии с требованиями.

Приветствуется обсуждение самих заданий с другими студентами: можно как давать, так и получать советы по общей стратегии выполнения и изучения материала, давать и получать помощь в отладке. Однако писать код студент должен самостоятельно. Делиться кодом или писать его совместно запрещено.

7 Фонд оценочных средств

7.1 Методы контроля и оценивания результатов обучения

Приведенные ниже правила выставления оценок и опозданий могут быть изменены, если преподаватель сочтет это необходимым. Важно, чтобы студенты регулярно просматривали план курса, выложенный в СДО, на предмет его обновления или изменения.

Достижение компетенций оценивается с помощью лабораторных работ и рубежных контролей. Индикатор ИПК-4.1 заложен в теме 1; ИПК-5.2 в темах 3, 4, 5, 6, 7, 8, 10, 11; ИПК-4.3 в темах 1, 9, 13, ИПК-5.3 в темах 2, 12.

В соответствии с планом на дисциплины студентам выдаются задания на лабораторные работы. Помимо требований и описания функционала в работе указан крайний срок сдачи. Для сдачи лабораторной работы студенту необходимо прислать ссылку на репозиторий в GitHub Classroom и на хостинг, где размещен результат работы с реализованным функционалом, описанным в задании. Работа считается сданной если в ней реализовано 80% и более требований и функционала, описанного в задании.

Каждый студент имеет право на 6 дней опоздания, которые могут быть потрачены на любые задания в течение семестра. Опоздания предназначены для решения особых ситуаций, таких как болезнь или чрезвычайные семейные обстоятельства.

Когда использованы все дни опоздания за каждый день просрочки начисляется штраф в размере 25% от максимального результата за задание. Задания, присланные позже, чем 4 дня, не будут оцениваться. В связи с зависимостью между работами студентам может потребоваться все равно выполнить предыдущие работы, даже если они не оцениваются.

После сдачи лабораторной работы студент должен ее защитить. Во время защиты лабораторной работы преподаватель проверяет репозиторий, хостинг и выполнение критериев и требований задания, а студент отвечает на вопросы преподавателя по его коду, а также теоретических вопросов, приведенных после текста задания лабораторной работы. Если студент отказывается отвечать на вопросы, или дает полностью неверные ответы, или ответы не по теме, то работа может считаться сданной, но при этом она не оценивается.

Работа должна быть выполнена студентом самостоятельно: в репозитории в системе контроля версий студента содержатся коммиты только за его авторством, по этим коммитам можно проследить как велась работа, студент может объяснить свой код и ход выполнения работы, если эти правила не соблюдаются, то работа не считается сданной и не оценивается.

Рубежные контроли пишутся в аудитории индивидуально по варианту задания, выданному преподавателем в назначенные дни. При отсутствии студента в день написания контрольной работы ему дается еще один шанс ее написать на последнем занятии в семестре, но обязательно очно.

Студенты должны заранее сообщать о том, что у них могут возникнуть трудности со своевременной сдачей задания или проекта. При наличии реальных причин задержки студентам следует как можно скорее связаться с преподавателем и обсудить возможные условия.

7.2 Шкала и критерии оценивания результатов обучения

Лабораторная работа оценивается в процентах степени выполнения следующих критериев и для выставления оценки суммируются проценты за каждый из четырех критериев:

1. Полнота выполнения практического задания (30%): соответствует ли функциональность заданным требованиям и целям, насколько точно и без ошибок код выполняет поставленные задачи, насколько эффективно задание отвечает требованиям целевой аудитории и обеспечивает приятное восприятие.
2. Качество и структура кода (10%): качество, читаемость и организация кода, рациональность выполнения задания, последовательность именования и соблюдение лучших практик.
3. Творчество и инновации (10%): творческий подход студентов к выполнению заданий, насколько студенты вышли за рамки основных требований и реализовали дополнительные возможности или использовали уникальные решения.
4. Ответы на вопросы по коду студента и теории (50%):

Дает краткий ответ, содержащий ошибки или неточности. На наводящие вопросы отвечает неправильно (10% из 50%)

Дает развернутый ответ, содержащий ошибки или неточности. На наводящие вопросы отвечает неверно (20% из 50%)

Дает развернутый ответ, содержащий ошибки или неточности. На наводящие вопросы отвечает правильно (30% из 50%)

Дает правильные и развернутые ответы на вопросы (50% из 50%).

R лабораторные рассчитывается как среднее результатов за все лабораторные работы. За полное и безошибочное выполнение всех лабораторных работ в срок и их защиту можно получить максимум 100 баллов (R лабораторные).

Рубежный контроль оценивается по следующим критериям:

Полнота выполнения практического задания: соответствует ли функциональность заданным требованиям и целям, насколько точно и без ошибок код выполняет поставленные задачи.

Качество и структура кода: качество, читаемость и организация кода, рациональность выполнения задания, последовательность именования и соблюдение лучших практик.

Творчество и инновации: творческий подход студентов к выполнению заданий, насколько студенты вышли за рамки основных требований и реализовали дополнительные возможности или использовали уникальные решения.

Пользовательский опыт: отзывчивость, доступность, насколько эффективно задание отвечает требованиям целевой аудитории и обеспечивает приятное восприятие.

Самостоятельность решения: в репозитории студента есть коммиты только за его авторством, по коммитам в репозитории можно проследить как велась работа, студент может объяснить свой код и ход выполнения работы, если эти правила не соблюдаются, то работа не считается сданной.

Более подробное описание критериев дается в тексте задания рубежного контроля.

За полностью выполненные рубежные контроли также можно получить 100 баллов (R контроль).

Также имеется коэффициент сданных работ K сданные, который равен 1 если все работы сданы и 0 если хотя бы одна работа не сдана.

Итоговый балл рассчитывается по формуле: R сем = (0,5 x R лабораторные + 0,5 x R контроль) * K сданные.

Итоговый балл пересчитывается по шкале ниже и на основании полученной оценки фиксируется результат промежуточной аттестации.

Соответствие баллов в 100 балльной рейтинговой системе оценке по 4-бальной шкале:

0-54 - неудовлетворительно

55-69 - удовлетворительно

70-84 - хорошо

85-100 – отлично

7.3 Оценочные средства

7.3.1 Текущий контроль

Примерный список вопросов

1. Что такое HTML?
2. Что такое тег/атрибут/значение?
3. Для чего используется тег *имя тега*?
4. Зачем нужен DOCTYPE?
5. Как отличить кнопку от ссылки?
6. Что такое CSS?
7. Что такое каскад?
8. Как рассчитывается специфичность?
9. Как специфичность влияет на каскад?
10. Что подразумевает наследование и как его использовать на благо?
11. Из чего состоят правила?
12. Какие бывают селекторы?
13. Для чего нужны normalize.css или reseter.css?
14. Какие существуют 3 способа подключения стилей?
15. Как создаются Custom Properties?
16. Классификация шрифтов.
17. На что влияет свойство font-weight?
18. Почему в для свойства font-family лучше указывать несколько шрифтов?
19. Что меняет свойство color?

20. Что такое CSS?
21. Что такое каскад?
22. Как рассчитывается специфичность?
23. Как специфичность влияет на каскад?
24. Что подразумевает наследование и как его использовать на благо?
25. Из чего состоят правила?
26. Какие бывают селекторы?
27. Для чего нужны normalize.css или reseter.css?
28. Какие существуют 3 способа подключения стилей?
29. Как создаются Custom Properties?
30. Классификация шрифтов.
31. На что влияет свойство font-weight?
32. Почему в для свойства font-family лучше указывать несколько шрифтов?
33. Что меняет свойство color?
34. Как работает та или иная единица измерения?
35. Какие варианты задания цветов существуют и чем они отличаются?
36. Что такое флекс-контейнер?
37. Что такое флекс-элемент?
38. Какие свойства применяются на флекс-контейнере?
39. Какие свойства применяются на флекс-элементе?
40. Какие оси есть у флексов и какие свойства меняют их направления?
41. Как выстраиваются флекс-элементы в контейнере?
42. Какие свойства и как они выравнивают флекс-элементы?
43. Что делают свойства flex-grow, flex-shrink и flex-basis?
44. Как работает та или иная единица измерения?
45. Какие варианты задания цветов существуют и чем они отличаются?
46. Что такое флекс-контейнер?
47. Что такое флекс-элемент?
48. Какие свойства применяются на флекс-контейнере?
49. Какие свойства применяются на флекс-элементе?
50. Какие оси есть у флексов и какие свойства меняют их направления?
51. Как выстраиваются флекс-элементы в контейнере?
52. Какие свойства и как они выравнивают флекс-элементы?
53. Что делают свойства flex-grow, flex-shrink и flex-basis?
54. Как работает та или иная единица измерения?
55. Какие варианты задания цветов существуют и чем они отличаются?
56. Что такое флекс-контейнер?
57. Что такое флекс-элемент?
58. Какие свойства применяются на флекс-контейнере?
59. Какие свойства применяются на флекс-элементе?
60. Какие оси есть у флексов и какие свойства меняют их направления?
61. Как выстраиваются флекс-элементы в контейнере?
62. Какие свойства и как они выравнивают флекс-элементы?
63. Что делают свойства flex-grow, flex-shrink и flex-basis?
64. Как работает та или иная единица измерения?
65. Какие варианты задания цветов существуют и чем они отличаются?

66. Что такое флекс-контейнер?
67. Что такое флекс-элемент?
68. Какие свойства применяются на флекс-контейнере?
69. Какие свойства применяются на флекс-элементе?
70. Какие оси есть у флексов и какие свойства меняют их направления?
71. Как выстраиваются флекс-элементы в контейнере?
72. Какие свойства и как они выравнивают флекс-элементы?
73. Что делают свойства flex-grow, flex-shrink и flex-basis?
74. Как работает та или иная единица измерения?
75. Какие варианты задания цветов существуют и чем они отличаются?
76. Что такое флекс-контейнер?
77. Что такое флекс-элемент?
78. Какие свойства применяются на флекс-контейнере?
79. Какие свойства применяются на флекс-элементе?
80. Какие оси есть у флексов и какие свойства меняют их направления?
81. Как выстраиваются флекс-элементы в контейнере?
82. Какие свойства и как они выравнивают флекс-элементы?
83. Что делают свойства flex-grow, flex-shrink и flex-basis?
84. Как работает та или иная единица измерения?
85. Какие варианты задания цветов существуют и чем они отличаются?
86. Что такое флекс-контейнер?
87. Что такое флекс-элемент?
88. Какие свойства применяются на флекс-контейнере?
89. Какие свойства применяются на флекс-элементе?
90. Какие оси есть у флексов и какие свойства меняют их направления?
91. Как выстраиваются флекс-элементы в контейнере?
92. Какие свойства и как они выравнивают флекс-элементы?
93. Что делают свойства flex-grow, flex-shrink и flex-basis?
94. Как работает та или иная единица измерения?
95. Какие варианты задания цветов существуют и чем они отличаются?
96. Что такое флекс-контейнер?
97. Что такое флекс-элемент?
98. Какие свойства применяются на флекс-контейнере?
99. Какие свойства применяются на флекс-элементе?
100. Какие оси есть у флексов и какие свойства меняют их направления?
101. Как выстраиваются флекс-элементы в контейнере?
102. Какие свойства и как они выравнивают флекс-элементы?
103. Что делают свойства flex-grow, flex-shrink и flex-basis?
104. Как работает та или иная единица измерения?
105. Какие варианты задания цветов существуют и чем они отличаются?
106. Что такое флекс-контейнер?
107. Что такое флекс-элемент?
108. Какие свойства применяются на флекс-контейнере?
109. Какие свойства применяются на флекс-элементе?
110. Какие оси есть у флексов и какие свойства меняют их направления?
111. Как выстраиваются флекс-элементы в контейнере?

112. Какие свойства и как они выравнивают флекс-элементы?
113. Что делают свойства flex-grow, flex-shrink и flex-basis?

Пример задания рубежного контроля

Главные требования

1. Работа выполнена самостоятельно.
2. Работа создана во время проведения занятия.
3. Не менее трех коммитов за авторством студента, равномерно распределенных по времени выполнения задания.
4. Работа размещена на удаленном репозитории на Github, который создан через Github Classroom.
5. Работа размещена на хостинге (сервер Мосполитеха).

Полезные ресурсы

Цвета:

- Готовые цветовые палитры: https://colorhunt.co/, https://coolors.co/
- Подобрать самостоятельно https://color.adobe.com/create/color-wheel

Шрифты: https://fonts.google.com/ (Продвинутый способ: https://google-webfonts-helper.herokuapp.com/fonts)

Примеры внешнего вида

- https://dessign.net/web-developer-portfolio-examples/

Задание

Документ index.html содержит:

- Шапка страницы:
    - Элемент ведущий на главную страницу (абсолютный путь до вашего домена): инициалы шрифтом из группы Display или произвольный логотип.
    - Ссылка на ваш профиль на github.
    - Навигационное меню (nav) с якорными ссылками на разделы страницы.
- Основная часть:
    - Крупный блок на весь первый экран (называется Hero Image или Jumbotron). Ваше имя, одно предложение о вас, кнопка с призывом связаться с вами, на фоне крупное изображение (можно взять тут https://unsplash.com/).
    - Список технологий и языков программирования с логотипами: что уже изучаете, что планируете в ближайшее время. Укажите опыт работы с технологией/языком программирования (полгода, год и т.д.). Списки стоит разместить в нескольких колонках.
    - Примеры ваших работ (скриншот, краткое описание, ссылка на сервер если есть) в виде карточек или карточки выполненных лабораторных работ по дисциплине “Основы веб-технологий”, номер, процент выполнения. В начале разделе есть ссылка со стрелкой “смотреть еще”.
    - Отзывы на ваши работы: рейтинг звездочками или другими изображениями, дата, отзыв, имя автора отзыва.
    - Контакты: почта, телефон, telegram и т.д. (можно использовать **ненастоящие** данные). Визуально выделите предпочтительный вариант для связи.
- Подвал:
    - ФИО, группа, дата написания РК.

- Форма с возможностью оставить имя и номер телефона для обратной связи и чекбокс для срочной связи.

Критерии оценки (за невыполнение критерия баллы снимаются полностью или частично)

Общие

- Выполнены главные требования.
- Содержание страницы соответствует заданию.
- Страница выглядит цельно.

Код

- В коде соблюдаются отступы.
- В CSS не используются вложенность селекторов более двух.

HTML

- Правильная структура документа.
- Правильно использованы заголовки разных уровней.
- Использованы семантические теги html5: header, main, footer, nav, section; по необходимости article.
- HTML валиден, нет нарушений вложенности тегов.
- Выбраны правильные форматы изображений.
- Задан правильный язык страницы.
- Ссылки с телефонами, email сделаны корректно.
- Нет лишних оберток div.
- Если нет ссылки в шапке на ваш домен, то работа считается не загруженной на сервер.

CSS

- Корректно подключен и использован шрифт с Google Fonts.
- Использовано не менее двух разных значений для font-weight.
- Использованы пользовательские свойства (custom properties в :root) для цветовой палитры.
- Цвета из палитры применены к тексту, фону, границам.
- Header и footer выделены фоновым цветом.
- Контент страницы имеет ограничение максимальной ширины (1200-1400px) и выровнен по середине (одинаковые поля слева и права).
- Содержимое элементов не переполняет их.
- Стили выдерживаю увеличение или уменьшение количества контента.
- Элементы не наезжают на элементы, следующие за ними.
- Текст и изображения не слипаются друг с другом. Грамотно использованы внутренние и внешние отступы.
- Применено скругление границ.
- Использован **box-sizing: border-box;**.
- Карточки располагаются в ряд, имеют отступы друг от друга, использован flexbox.
- Изменяется оформление элемента при наведении на него курсора.
- К карточкам с проектами применена анимация, которая срабатывает при наведении.
- Использованы переходы.
- Изменены стандартные стили для элементов формы и чекбокса.
- Изменение состояния курсора при наведении на кнопки (cursor: pointer;).
- Классы названы логично, а не являются набором букв и цифр.

GIT и хостинг

1. Клонировать репозиторий, который был создан Github Classroom (**git clone ...**).
2. Установить необходимые пакеты для работы над проектом (**npm ci**).
3. При разработке периодически сохранять результаты в репозиторий git (команды **git add**, **git commit -m "комментарий к версии"**).
4. Перед завершением работы проверьте валидность вашего кода с помощью валидаторов для html и css.
5. Отформатируйте все документы html, css.
6. После проверки на валидаторе не забудьте отправить ваши изменения на github (**git push**).
7. Для загрузки на хостинг соберите проект (**npm run build**).
8. Создайте домен и загрузите в нужную папку с помощью FileZilla файлы из папки build.

7.3.2 Промежуточная аттестация

Оценочные средства для промежуточной аттестации не требуется, так как оценка за промежуточную аттестацию выставляется по балльно-рейтинговой системе, описанной в пункте 7.2.