Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 03.11.2023 11:24:07
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/
«10» мая 2022

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Технологии программирования

Направление подготовки/специальность
09.03.02 Информационные системы и технологии

Профиль/специализация
Цифровая трансформация

Квалификация
Бакалавр

Формы обучения
Очная

Москва, 2022 г.

Разработчик(и):

ст. преподаватель  / М.В. Алпатова /

Согласовано:
Заведующий кафедрой
«Информатики и информационных технологий»,
к.т.н.  / Е.В. Булатников/

Содержание

1 Цели, задачи и планируемые результаты обучения по дисциплине 4
2 Место дисциплины в структуре образовательной программы 6
3 Структура и содержание дисциплины 6
3.1 Виды учебной работы и трудоемкость 6
3.2 Тематический план изучения дисциплины 7
3.3 Содержание дисциплины 7
3.4 Тематика семинарских/практических и лабораторных занятий 9
3.5 Курсовое проектирование 11
4 Учебно-методическое и информационное обеспечение 12
4.1 Нормативные документы и ГОСТы 12
4.2 Основная литература 12
4.3 Дополнительная литература 13
4.4 Электронные образовательные ресурсы ....**Ошибка! Закладка не определена.**
4.5 Электронные образовательные ресурсы 14
4.6 Лицензионное и свободно распространяемое программное обеспечение 14
5 Материально-техническое обеспечение 14
6 Методические рекомендации 14
6.1 Методические рекомендации для преподавателя по организации обучения 14
6.2 Методические указания для обучающихся по освоению дисциплины 14
7 Фонд оценочных средств 15
7.1 Методы контроля и оценивания результатов обучения 15
7.2 Шкала и критерии оценивания результатов обучения 15
7.3 Оценочные средства 16

1 Цели, задачи и планируемые результаты обучения по дисциплине

Целью изучения дисциплины «Технологии программирования» является Подготовка специалистов, способных к разработке, тестированию и внедрению программного обеспечения на основе современных методологий и технологий программирования.

В рамках дисциплины ставятся следующие задачи, соответствующие задачам профессиональной деятельности, указанным в ФГОС ВО:

- Ознакомление с основами языков программирования, в частности Python и JavaScript.
- Изучение основной структуры и элементов веб-разработки, включая HTML и CSS.
- Формирование навыков по работе с современными инструментальными средствами проектирования ПО.
- Развитие практических навыков в создании, отладке и тестировании программного обеспечения.
- Подготовка к профессиональной деятельности в сфере разработки информационных систем и автоматизированных систем обработки информации

.

По завершении изучения дисциплины студенты:

- Смогут разрабатывать требования и программное обеспечение для автоматизации информации и управления системами на основе изученных языков программирования;
- Ознакомятся со средами разработки, что позволит им в дальнейшем интегрировать и внедрять современные информационные системы;
- Смогут использовать современные информационные технологии для создания и настройки информационных систем;
- Будут владеть знаниями основных языков программирования, а также сред разработки информационных систем
- Смогут применять языки программирования для автоматизации бизнес-процессов и ведения баз данных

Обучение по дисциплине «Технологии программирования» направлено на формирование у обучающихся следующих компетенций:

Код и наименование компетенций	**Индикаторы достижения компетенции**
ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий.	ИОПК-6.1 знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий ИОПК-6.2 умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ ИОПК-6.3 имеет навыки

	программирования, отладки и тестирования прототипов программно-технических комплексов задач
ОПК-2. Способен принимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности.	ИОПК-2.1. знает современные информационные технологии и программные средства, основные виды и принципы работы информационных систем и информационных технологий; способы внедрения и интеграции современных информационных систем, способы оценки необходимости использования программных средств ИОПК-2.2. умеет использовать современные информационные технологии и программные средства, как в рамках отдельного предприятия, так и в рамках корпораций, государственных систем; внедрять и настраивать современные информационные системы, проводить интеграцию различных информационных систем и программных средств, оценивать необходимость использования программного средства для решения задач ИОПК-2.3. владеет навыками применения современных информационных технологий и программных средств, при решении задач в различных отраслях, внедрения и настройки современных информационных систем, оценки необходимости использования программных средств и информационных систем для решения задач
ПК-1. Способен разрабатывать требования и проектировать программное обеспечение.	ИПК-1.1. Знает способы разработки требований и проектирования программного обеспечения для обработки и автоматизации информации и систем управления ИПК-1.2. Умеет проектировать программное обеспечение и системы управления с применением современных инструментальных средств ИПК-1.3. Имеет навыки разработки требований и проектирования информационных систем управления и автоматизированных систем обработки информационных потоков и систем

2 Место дисциплины в структуре образовательной программы

Дисциплина относится к модулю «Базовые информационные технологии» обязательной части блока Б1 «Дисциплины (модули)».

Дисциплина взаимосвязана логически и содержательно-методически со следующими дисциплинами и практиками ОПОП:

- Основы цифровой трансформации
- Программирование для мобильных устройств
- Проектирование интерфейсов информационных систем
- Сети и телекоммуникации
- Аппаратное обеспечение информационных систем
- Базы данных
- Распределенные компьютерные системы сбора и обработки данных
- Методы искусственного интеллекта
- Архитектура информационных систем
- Информационная безопасность и защита информации
- Анализ данных
- Управление программными проектами
- Введение в проектную деятельность
- Проектная деятельность
- Инженерия требований
- Системный анализ
- Тестирование ПО
- Корпоративные системы обработки данных
- Операционные системы
- Администрирование информационных систем

3 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 8 зачетных единиц, т.е. 288 академических часов (из них 102 часа – аудиторные занятия и 186 часов – самостоятельная работа студентов).

Разделы дисциплины изучаются на 1 курсе в 1 и 2 семестре, форма промежуточной аттестации – зачет (1 семестр), экзамен (2 семестр). Во втором семестре предусмотрено курсовое проектирование.

3.1 Виды учебной работы и трудоемкость (по формам обучения)

3.1.1 Очная форма обучения

№ п/п	Вид учебной работы	Количество часов	Семестры	
			Семестр	Неделя семестра
1	**Аудиторные занятия**	**102**	1, 2	1-17, 23-40
	В том числе:			
1.1	Лекции	34		
1.2	Семинарские/практические занятия	-		
1.3	Лабораторные занятия	68		
2	**Самостоятельная работа**	**186**	1, 2	1-17, 23-40
3	**Курсовое проектирование**		**-, КП**	

4	**Промежуточная аттестация**		1, 2	19-21, 41-43
	Экзамен/зачет		зачет, экзамен	
	Итого:	**288**		

3.2 Тематический план изучения дисциплины (по формам обучения)

3.2.1 Очная форма обучения

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	Аудиторная работа				Самостоятельная работа
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1 семестр							
1.1	Введение в Python	4	2		2		14
1.2	Условия и циклы	4	2		2		16
1.3	Массивы данных	8	2		6		14
1.4	Функции	8	2		6		16
1.5	Объектно-ориентированное программирование	16	4		12		16
1.6	Введение в анализ данных	8	4		4		14
2 семестр							
2.1	HTML5	12	6		6		24
2.2	CSS3	12	4		8		22
2.3	JavaScript	14	6		8		24
2.4	Курсовое проектирование	14			14		23

3.3 Содержание дисциплины

3.3.1 Семестр 1

Тема 1. Введение в Python:

Основы языка Python, его история и особенности. Презентация возможностей и сфер применения Python в современном мире.

Тема 2. Первая программа на Python:

Создание и запуск простой программы на Python. Обзор основных элементов языка и синтаксиса.

Тема 3. Условия:

Изучение условных операторов в Python, их синтаксис и применение для создания ветвлений в коде.

Тема 4. Циклы:

Основы работы с циклами в Python: for и while. Применение циклов для повторяющихся действий.

Тема 5. Списки:

Введение в структуры данных Python. Создание, манипуляция и применение списков.

Тема 6. Строки:

Работа со строками в Python, методы и операции над строками.

Тема 7. Создание функций:

Основы создания пользовательских функций, передача аргументов и возврат значений.

Тема 8. Работа с файлами:

Основы чтения и записи файлов в Python, работа с разными форматами данных.

Тема 9. Введение в ООП:

Основные концепции объектно-ориентированного программирования и их реализация в Python.

Тема 10. Классы и объекты:

Создание и использование классов и объектов в Python, инкапсуляция, наследование и полиморфизм.

Тема 11. Введение в анализ данных:

Обзор основных методик и инструментов для работы с данными на Python.

Тема 12. Манипуляции с данными:

Применение Python для обработки, анализа и визуализации данных.

3.3.2 Семестр 2

Тема 1. Введение в HTML:

Обзор основ HTML, его структура и принципы создания веб-страниц.

Тема 2. Основы структуры HTML документа:

Изучение основных тегов, атрибутов и структурирование содержимого веб-страницы.

Тема 3. Категории HTML контента:

Разделение контента на текстовый, мета-данные, встроенный и т.д. Применение соответствующих тегов.

Тема 4. Основы правил CSS:

Введение в каскадные таблицы стилей, основы стилизации и форматирования веб-страниц.

Тема 5. Селекторы CSS:

Использование различных типов селекторов для применения стилей к элементам HTML.

Тема 6. Позиционирование элементов в CSS:

Методы и стратегии позиционирования элементов на веб-странице с использованием CSS.

Тема 7. Знакомство с JavaScript:

Введение в язык программирования JavaScript, его возможности и сферы применения.

Тема 8. Объявление переменных и функций в JS:

Основы создания переменных и функций в JavaScript, область видимости и жизненный цикл.

Тема 9. Типы данных в JS:

Обзор основных типов данных в JavaScript, их особенности и способы применения.

Тема 10. Основные конструкции JS:

Изучение основных операторов, условий и циклов в JavaScript.

Тема 11. Создание объектов в JS:

Основы работы с объектами в JavaScript, создание, манипуляция и использование.

Тема 12. Объяснение функций в JS:

Глубокое погружение в механику функций в JavaScript, замыкания, контекст вызова и другие особенности.

3.4 Тематика семинарских/практических и лабораторных занятий

Семестр 1

«IF Hello, World»:

Цель: Знакомство с базовыми операторами условия в Python.

Задачи:

- Написать программу, выводящую "Hello, World" при определенном условии.
- Экспериментировать с разными условиями для понимания их работы.

«Циклы For и While»:

Цель: Изучить основные типы циклов в Python.

Задачи:

- Создать цикл, перебирающий последовательность чисел.
- Реализовать циклическое повторение операции с использованием условий.

«Работа со списками»:

Цель: Освоить создание и манипулирование списками в Python.

Задачи:

- Создать список с разными типами данных.
- Применить основные методы манипуляции списком.

«Манипуляции со строками»:

Цель: Понять основы работы со строками и методами строк.

Задачи:

- Создание и изменение строк.
- Использование методов строк для манипуляции и поиска.

«Создание функций»:

Цель: Освоить создание и использование функций в Python.

Задачи:

- Разработать функцию для выполнения конкретной задачи.
- Реализовать передачу аргументов и возврат значения.

«Сохранение и загрузка файлов»:
Цель: Изучить чтение и запись данных в файлы.
Задачи:

- Создать файл и записать в него информацию.
- Прочитать данные из файла и проанализировать их.

«Контрольная работа по темам 1-8»:
Цель: Проверить усвоение материала по изученным темам.
Задачи:
Выполнить задания, основанные на темах 1-8.

«Классы и объекты»:
Цель: Освоить создание классов и объектов в Python.
Задачи:

- Разработать класс с атрибутами и методами.
- Создать объекты этого класса и работать с ними.

«Контрольная работа по темам 9, 10»:
Цель: Проверить усвоение материала по объектно-ориентированному программированию.
Задачи:

- Выполнить задания, связанные с классами, объектами и их взаимодействием.

«Работа с данными»:
Цель: Практика работы с данными: их анализ и обработка.
Задачи:

- Загрузить набор данных, провести его анализ.
- Применить методы обработки и визуализации данных.

«Контрольная по теме 11,12»:
Цель: Проверить усвоение материала по анализу данных.
Задачи:

- Выполнить задания, связанные с анализом и обработкой данных.

Семестр 2:
«Hello, World»:
Цель: Знакомство с базовой структурой HTML.
Задачи:

- Создать простую веб-страницу, выводящую "Hello, World".

«HTML тэги»:
Цель: Изучить основные HTML теги и их использование.
Задачи:

- Создать структурированную веб-страницу с использованием различных тегов.

«Работа со стилями»:
Цель: Освоить применение базовых стилей CSS.
Задачи:

- Стилизовать элементы веб-страницы с использованием CSS.

«Задачи по JS №1»:
Цель: Знакомство с основами JavaScript.
Задачи:

- Решить базовые задачи на JavaScript, такие как объявление переменных и простые операции.

«Задачи по JS №2»:
Цель: Глубже погрузиться в особенности JavaScript.
Задачи:

- Решить задачи, связанные с использованием функций и объектов в JS.

«Задачи по JS №3»:
Цель: Продолжить изучение JavaScript, углубившись в сложные конструкции.
Задачи:

- Решить задачи на применение циклов, условных операторов и работы с данными в JS.

3.5 Курсовое проектирование

Тема: Разработка многостраничного сайта с использованием технологий HTML, CSS и JS.

Цели и задачи:

Основная цель: Создание функционального многостраничного сайта, демонстрирующего полученные студентами навыки веб-разработки.

Задачи:

Разработать структуру сайта, состоящего из примерно 10 статичных веб-страниц.

1. Применить базовые технологии HTML для структурирования содержимого: заголовки, блоки, изображения, таблицы и пр.
2. Использовать CSS для стилизации элементов сайта.
3. Добавить интерактивные элементы с использованием JavaScript.
4. При желании и наличии соответствующих навыков, интегрировать продвинутые веб-технологии, такие как React.js, THREE.js и другие.

Требования к содержанию сайта:

- Сайт должен быть разработан с соблюдением норм и стандартов веб-разработки.
- Тематика сайта может быть любой, интересной самому студенту, но содержание сайта не должно противоречить нормативным кодексам и законам РФ.

- Сайт должен содержать разнообразные элементы (текст, изображения, мультимедиа, интерактивные элементы, таблицы).

Требования к предоставлению проекта:

- В случае использования серверных технологий, требующих специфического хостинга или настройки, студент обязан предоставить преподавателю доступ к размещенному в сети Интернет сайту. Ссылка на сайт и исходные коды должны быть предоставлены для проверки.
- Преподаватель не должен производить дополнительные действия по установке и настройке сервера, загрузке зависимостей и прочее. Все необходимые компоненты для функционирования сайта должны быть интегрированы или хоститься на сторонних ресурсах.

Оценка:

Оценка проекта будет производиться на основе соответствия требованиям, функциональности сайта, его дизайна, а также правильного использования технологий HTML, CSS и JS.

4 Учебно-методическое и информационное обеспечение

4.1 Нормативные документы и ГОСТы

1. Федеральный закон от 29 декабря 2012 года No 273-ФЗ «Об образовании в Российской Федерации» (с изменениями и дополнениями);
2. Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.02 Информационные системы и технологии, утвержденный Приказом Министерства образования и науки РФ от 19 сентября 2017 г. N 929 "Об утверждении федерального... Редакция с изменениями N 1456 от 26.11.2020;
3. Приказ Министерства образования и науки РФ от 05 апреля 2017 г. No 301 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры.

4.2 Основная литература

1. Буйначев, С. К. Основы программирования на языке Python : учебное пособие / С. К. Буйначев, Н. Ю. Боклаг ; под редакцией Ю. В. Песин. — Екатеринбург : Уральский федеральный университет, ЭБС АСВ, 2014. — 92 с. — ISBN 978-5-7996-1198-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/66183.html (дата обращения: 23.08.2023). — Режим доступа: для авторизир. пользователей

2. Маккинли, Уэс Python и анализ данных / Уэс Маккинли ; перевод А. Слинкина. — 2-е изд. — Саратов : Профобразование, 2019. — 482 с. — ISBN 978-5-4488-0046-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/88752.html (дата обращения: 23.08.2022). — Режим доступа: для авторизир. пользователей

3. Шелудько, В. М. Основы программирования на языке высокого уровня Python : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного

федерального университета, 2017. — 146 с. — ISBN 978-5-9275-2649-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/87461.html (дата обращения: 23.08.2023). — Режим доступа: для авторизир. пользователей

4. Шелудько, В. М. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 107 с. — ISBN 978-5-9275-2648-2. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/87530.html (дата обращения: 10.07.2023). — Режим доступа: для авторизир. пользователей

5. Титов, В. А. Разработка WEB-сайта средствами языка HTML : учебное пособие / В. А. Титов, Г. И. Пещеров. — Москва : Институт мировых цивилизаций, 2018. — 184 с. — ISBN 978-5-9500469-3-3. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/80643.html (дата обращения: 23.08.2023). — Режим доступа: для авторизир. пользователей

6. Савельев, А. О. HTML5. Основы клиентской разработки : учебное пособие / А. О. Савельев, А. А. Алексеев. — 3-е изд. — Москва, Саратов : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 270 с. — ISBN 978-5-4497-0296-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/89407.html (дата обращения: 23.08.2023). — Режим доступа: для авторизир. пользователей

7. Кулькова, Л. И. Задачи и упражнения по JavaScript : учебное пособие / Л. И. Кулькова, С. И. Салпагаров. — Москва : Российский университет дружбы народов, 2018. — 102 с. — ISBN 978-5-209-08646-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/104199.html (дата обращения: 23.08.2023). — Режим доступа: для авторизир. пользователей

8. Рындин, Н. А. Технологии разработки клиентских WEB-приложений на языке JavaScript : учебное пособие / Н. А. Рындин. — Воронеж : Воронежский государственный технический университет, ЭБС АСВ, 2020. — 54 с. — ISBN 978-5-7731-0888-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/108188.html (дата обращения: 23.08.2023). — Режим доступа: для авторизир. пользователей

4.3 Дополнительная литература

1. Мусин Д. Самоучитель Python. 2017.
2. Любанович Б. Простой Python. Современный стиль программирования. - 2 изд. Питер, 2021.
3. Мэтиз Э. Изучаем Python: программирование игр, визуализация данных, веб-приложения. - 3 изд. Питер, 2021.
4. Маккинни У. Python и анализ данных. ДМК Пресс, 2013.
5. Зиаде Т., Яворски М. Python. Лучшие практики и инструменты. - 3 изд. Питер, 2019.
6. Роббинс, Д.Н., 2021. Веб-дизайн для начинающих. HTML, CSS, JavaScript и веб-графика. БХВ.
7. Дакетт, Д., 2020. Javascript и jQuery. Интерактивная веб-разработка. Эксмо.
8. Симпсон, К., 2022. Вы пока еще не знаете JS. Познакомьтесь, JavaScript. Питер.
9. Заяц, А.М., 2021. Проектирование и разработка WEB-приложений. Введение в frontend и backend разработку на JavaScript и node.js. Лань.
10. Форсье, Д. and П. Биссекс, 2020. Django. Разработка веб-приложений на Python. Символ.

4.4 Электронные образовательные ресурсы

1. Технологии программирования. LMS Московского политеха. URL: https://online.mospolytech.ru/course/view.php?id=8629
2. Технологии программирования: Web. LMS Московского политеха. URL: https://online.mospolytech.ru/course/view.php?id=9944

4.5 Лицензионное и свободно распространяемое программное обеспечение

1. VS Code
2. PyCharm Community
3. Anaconda, Spyder

4.6 Современные профессиональные базы данных и информационные справочные системы:

1. Образовательная платформа Юрайт
2. Электронно-библиотечная система Лань
3. Цифровой образовательный ресурс IPR SMART

5 Материально-техническое обеспечение

Лабораторные работы и самостоятельная работа студентов должны проводиться в специализированной аудитории, оснащенной современной оргтехникой и персональными компьютерами с программным обеспечением в соответствии с тематикой изучаемого материала. Число рабочих мест в аудитории должно быть достаточным для обеспечения индивидуальной работы студентов. Рабочее место преподавателя должно быть оснащены современным компьютером с подключенным к нему проектором на настенный экран, 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 указания для обучающихся по освоению дисциплины

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, дорабатывают конспекты и записи, 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, промежуточный контроль осуществляется на зачете в письменной (устной) форме.

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.

7 Фонд оценочных средств

7.1 Методы контроля и оценивания результатов обучения

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:

- Выполнение лабораторных работ
- Промежуточное тестирование (посредством изучения теоретических материалов в системе LMS)
- Итоговое тестирование

7.2 Шкала и критерии оценивания результатов обучения

Итоговая оценка по дисциплине рассчитывается как среднее взвешенное всех оценок в соответствующем курсе LMS Московского политеха с применением весовых коэффициентов, представленных ниже:

1 семестр

- Лабораторные работы → 0,2
- Контрольные работы → 0,4
- Теория → 0,1
- Итоговый тест (экзамен) → 0,3

2 семестр

- Лабораторные работы → 0.3
- Семестровый проект → 0.5 (за проект)+0.05 (за лист задания)
- Теория → 0.05
- Итоговый тест (экзамен) → 0.1

Оценка за каждую лабораторную работу выставляется исходя из фактического выполнения всех поставленных задач с учётом сроков исполнения: за каждую 1 неделю просрочки задания из оценки вычитается 10 баллов.

Для получения положительной экзаменационной оценки студенту необходимо набрать всего минимально 55 баллов по дисциплине и завершить итоговый тест с результатом не менее 55%.

Шкала оценивания	Диапазон баллов	Описание
Неудовлетворительно	0-54	Не достигнуто пороговое значение хотя бы для одного уровня формируемых на момент проведения аттестации компетенций. Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.
Удовлетворительно	55-69	Среднее значение для всех формируемых на момент проведения аттестации уровней компетенций – 3. Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, в котором освещена основная, наиболее важная часть материала, но при этом допущена одна значительная ошибка или неточность.
Хорошо	70-84	Среднее значение для всех формируемых на момент проведения аттестации уровней компетенций – 4. Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует неполное, правильное соответствие знаний, умений, навыков приведенным в таблицах показателей, либо если при этом были допущены 2-3 несущественные ошибки.
Отлично	85-100	Среднее значение для всех формируемых на момент проведения аттестации уровней компетенций – 5. Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.

7.3 Оценочные средства

7.3.1 Вопросы к зачету

1. Что такое Python и для чего он используется?
2. Какие основные особенности Python отличают его от других языков программирования?
3. Какой синтаксис используется для написания первой программы "Hello, World" на Python?

4. Какие условные операторы существуют в Python и как они используются?
5. Объясните разницу между циклами for и while в Python.
6. Как создать и манипулировать списками в Python?
7. Какие операции можно выполнять со строками в Python?
8. Как определить функцию в Python и какие типы аргументов функции можно использовать?
9. Как в Python осуществляется чтение и запись файлов?
10. Что такое ООП и как Python поддерживает его основные принципы?
11. Что такое классы и объекты в контексте Python? Как определить класс и создать объект этого класса?
12. Какие инструменты Python предоставляются для анализа данных?
13. Какие методы существуют для манипулирования данными в Python?
14. В чем разница между списками и кортежами в Python?
15. Как можно обработать исключения в Python?
16. Что такое лямбда-функция в Python и как она используется?
17. Какие библиотеки Python рекомендуются для работы с данными и их анализа?
18. Что такое наследование в контексте ООП в Python?
19. Как работает механизм инкапсуляции в Python?
20. Каким образом в Python реализуется полиморфизм?

7.3.2 Вопросы к экзамену

1. Что такое HTML и для чего он используется в веб-разработке?
2. Какие основные элементы входят в структуру HTML документа?
3. Что такое блочные и строчные элементы в HTML? Приведите примеры.
4. Что такое CSS и каким образом он используется для стилизации веб-страниц?
5. Как работают селекторы в CSS? Приведите примеры разных типов селекторов.
6. Объясните основные концепции позиционирования элементов с помощью CSS.
7. Что такое JavaScript и какие задачи он решает в контексте веб-разработки?
8. Как объявляются переменные и функции в JS? Какие особенности есть у let, const и var?
9. Опишите основные типы данных в JavaScript.
10. Какие основные управляющие конструкции существуют в JS для реализации циклов и условных операторов?
11. Что такое объекты в JavaScript? Как создать объект и обратиться к его свойствам?
12. Какова роль и синтаксис функций в JavaScript?
13. Как взаимодействовать с DOM с помощью JavaScript?
14. Как работает событийная модель в JS и каким образом можно обрабатывать события пользовательского интерфейса?
15. Что такое CSS Flexbox и Grid? В чем их основные принципы работы?
16. Какие методы оптимизации загрузки веб-страниц вы знаете?
17. Что такое адаптивный и отзывчивый дизайн веб-страниц? В чем их основное отличие?
18. Какие инструменты и библиотеки могут помочь в создании интерактивных веб-приложений?

19. Что такое кросс-браузерная совместимость и какие основные проблемы могут возникнуть при ее обеспечении?
20. Опишите процесс разработки веб-приложения с нуля: от идеи до развертывания на сервере.