
УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«25» мая 2026 г.
Протокол № 2

**Рабочая программа дисциплины (модуля)
«Программирование»**

Направление подготовки: 38.03.05 Бизнес-информатика

Направленность (профиль) подготовки: Дизайн

Квалификация (степень) выпускника: бакалавр

Форма обучения: очная

Срок освоения программы: 4 года

Год набора: 2026

**Москва
2026**

Содержание

1. Краткая характеристика дисциплины (модуля)	3
2. Перечень планируемых результатов обучения.....	5
3. Тематический план.....	8
4. Содержание дисциплины (модуля).....	8
5. Учебно-методическое обеспечение	9
6. Материально-техническое обеспечение	9
7. Методические и оценочные материалы	11

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Программирование» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по специальности 38.03.05 «Бизнес-информатика», профиль Дизайн, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 838 от 29.07.2020 года.

Изучение дисциплины (модуля) «Программирование» позволяет освоить ключевые навыки современной разработки ПО, востребованные в IT-сфере и смежных областях. Полученные компетенции обеспечивают возможность автоматизации процессов, интеграции сервисов и участия в командных проектах с использованием актуальных инструментов разработки.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки бакалавриата по направлению 38.03.05 «Бизнес-информатика», профиль Дизайн и входит в часть, формируемую участниками образовательных отношений, Блока 1, как дисциплина по выбору.

Дисциплина (модуль) изучается на 4 курсе в 7 семестре, доступна для прохождения при условии успешного завершения дисциплины (модуля) «Формальный анализ».

Цель изучения дисциплины (модуля): сформировать у обучающихся фундаментальные знания и практические навыки программирования на языке Python для разработки модульных программ, автоматизации задач, работы с API и создания телеграм-ботов в рамках командной разработки с использованием систем контроля версий.

Задачи изучения дисциплины (модуля):

- изучить основы синтаксиса языка Python, типы данных, коллекции и функции для написания корректного и эффективного кода;
- освоить принципы работы с файлами, обработки исключений и структурирования программного кода;
- научиться выполнять HTTP-запросы к REST API и обрабатывать данные в формате JSON;
- развить навыки разработки и настройки телеграм-ботов с использованием Telegram Bot API и сценариев взаимодействия с пользователями;
- освоить работу с системой контроля версий Git и платформой GitLab для организации командной разработки и управления проектами.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- основы синтаксиса Python, типы данных, функции и коллекции;
- принципы работы с файлами, обработкой ошибок и структурированием кода;
- основы взаимодействия с REST API и обработки JSON-данных;
- базовые принципы разработки телеграм-ботов и работы с Telegram Bot API;
- основы систем контроля версий Git и организации проектов в GitLab.

уметь:

- писать и структурировать программы на Python с использованием функций и коллекций;
- работать с файлами, обрабатывать данные и использовать обработку ошибок;
- выполнять запросы к API и обрабатывать ответы в формате JSON;
- разрабатывать телеграм-ботов с командами, обработчиками и пользовательскими сценариями;
- использовать Git для командной разработки и управления версиями проекта.

владеть:

- навыками разработки программ и автоматизации задач на Python;
- подходами к созданию модульной и поддерживаемой структуры проекта;
- инструментами интеграции API в прикладные решения;
- практиками разработки и поддержки телеграм-ботов для пользовательских задач;
- инструментами командной разработки с использованием Git и GitLab.

2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
УК-1.	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1.	Знает методы поиска и анализа информации в области дизайна, основные принципы критической оценки источников информации, их релевантности и достоверности
		УК-1.2.	Умеет критически оценивать источники информации и синтезировать данные из различных источников для решения задач, применять системный подход к анализу и решению комплексных проблем
		УК-1.3.	Имеет практический опыт работы с современными инструментами и технологиями для обработки информации, формулировании и структурировании задач на основе полученной информации
УК-6.	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни	УК-6.1.	Знает основные принципы самовоспитания и самообразования, профессионального и личностного развития, исходя из этапов карьерного роста и требований рынка труда
		УК-6.2.	Умеет планировать свое рабочее время и время для саморазвития. формулировать цели личностного и профессионального развития и условия их достижения, исходя из тенденций развития области профессиональной деятельности, индивидуально-личностных особенностей
		УК-6.3.	Имеет практический опыт получения дополнительного образования, изучения дополнительных образовательных программ
УК-10.	Способен принимать обоснованные экономические решения в различных областях жизнедеятельности	УК-10.1.	Знает основные экономические теории и принципы экономической деятельности; нормативные и правовые акты,

			регулирующие экономическую деятельность
		УК-10.2.	Умеет анализировать экономическую информацию и данные; разрабатывать и обосновывать экономические стратегии и решения
		УК-10.3.	Имеет практический опыт в применении экономических знаний в реальных ситуациях и проектах
ОПК-3.	Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации	ОПК-3.1.	Знает принципы проектирования пользовательского взаимодействия, этапы жизненного цикла продуктов дизайна и основы разработки алгоритмов функционирования в цифровой среде
		ОПК-3.2.	Умеет управлять процессами создания продуктов дизайна, применять методы эскизирования, прототипирования и цифрового моделирования для практической реализации проектных замыслов
		ОПК-3.3.	Имеет практический опыт разработки и внедрения решений в сфере дизайна и информационно-коммуникационных технологий в рамках учебно-профессиональных и проектных задач
ПК-2.	Способен осуществлять поиск, создание и редактирование информационно-содержательных материалов для цифровых продуктов и коммуникационных решений	ПК-2.1.	Знает методы поиска и систематизации информации, принципы создания и редактирования текстовых и визуальных материалов, требования к контенту в цифровой и физической среде
		ПК-2.2.	Умеет осуществлять подбор информации по тематике проекта, создавать и модифицировать информационные материалы, адаптировать контент для различных цифровых платформ
		ПК-2.3.	Имеет практический опыт разработки информационно-содержательных материалов для веб-ресурсов, социальных сетей и цифровых продуктов в рамках учебно-профессиональных задач

ПК-3.	Способен управлять процессами наполнения, актуализации и нормативного контроля информационного контента в проектной деятельности	ПК-3.1.	Знает принципы организации работы с информационными ресурсами, методы контроля качества контента, требования законодательства к размещению информации в цифровой среде
		ПК-3.2.	Умеет организовывать процессы обновления и модерации информационного контента, осуществлять проверку материалов на соответствие нормативным требованиям, координировать работу с информационными источниками
		ПК-3.3.	Имеет практический опыт управления информационным наполнением цифровых продуктов и коммуникационных платформ в рамках проектной деятельности

3. Тематический план

№ п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Аудиторная работа		Контроль	Самостояте льная работа	
		Лекции	Семинары (практичес кие занятия)			
1	Основы Python		8	4	50	Коллоквиум
2	Инструменты и практика разработки		6	4	68	Проект
	<i>Зачёт с оценкой</i>			4		
<i>Итого:</i>			<i>60</i>	<i>12</i>	<i>118</i>	
<i>Объем дисциплины (модуля) (в ак. ч.)</i>		<i>190</i>				
<i>Объем дисциплины (модуля) (в зач. ед.)</i>		<i>5</i>				

4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Основы Python	Основы синтаксиса и типы данных. Переменные, числа, строки, булевы значения. Динамическая типизация, приведение типов. Ввод/вывод. Условные конструкции. Логические операторы. Вложенные условия. Циклы: for и while. По спискам, строкам, бесконечные циклы. Списки, кортежи и строки. Индексы, срезы, методы списков. Неизменяемость кортежей. Работа со строками. Словари и множества. Ключи и значения, вложенные словари. Множества: уникальность, операции. Упражнение: база данных студентов, подсчёт слов в тексте. Функции и области видимости. Параметры, возврат значений. Локальные и глобальные переменные. Документирование. Работа с файлами и обработка ошибок. Чтение/запись текстовых файлов. Исключения. Логирование ошибок. Коллоквиум. Проверка знаний: тест и практическое задание. Темы: типы данных, циклы, функции, файлы.
2	Инструменты и практика разработки	Git и GitLab. Основы Git: init, add, commit, push, pull. Работа с удалёнными репозиториями (GitLab). Ветки, gitignore, коммит-месседжи. Протокол HTTP и REST API. Что такое HTTP: GET, POST, статусы. Что такое API, JSON, эндпоинты. Примеры: погода, курсы валют, случайные факты. Работа с библиотекой requests и JSON. Установка requests. Отправка GET/POST-запросов. Парсинг JSON. Модульный код и разделение логики. Создание модулей. Структура проекта. Конфигурация: токены, настройки в отдельном файле. Создание бота. Получение токена. Простой echo-бот: "Повторяшка". Инлайн-кнопки и продвинутые обработчики. Клавиатуры: ReplyKeyboard, InlineKeyboard. Обработка callback-запросов. Состояния: ConversationHandler. Самостоятельный проект: бот с функциями. Требования: Git, модульность, обработка ошибок, интерфейс.

5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Чернышев, С. А. Основы программирования на Python : учебник для вузов / С. А. Чернышев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 349 с. — (Высшее образование). — ISBN 978-5-534-17139-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/588667>

2. Гаско, Р. Простой Python просто с нуля / Р. Гаско ; под ред. Н. Ю. Комлева. - Москва : СОЛОН-ПРЕСС, 2023. - 256 с. - (Серия «Программирование»). - ISBN 978-5-91359-334-4. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2185854>

Дополнительная литература:

1. Черткова, Е. А. Компьютерные технологии обучения : учебник для вузов / Е. А. Черткова. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2026. — 245 с. — (Высшее образование). — ISBN 978-5-534-12532-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/584740>

6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;
- механическими калькуляторами;
- специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование»	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное

Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

7. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Программирование» в рамках текущего контроля успеваемости используются такие виды учебной работы, как семинары, коллоквиум, проект, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Семинар — это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где студенты активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к семинару рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

Коллоквиум – устные ответы на вопросы, список которых известен студенту заранее.

В процессе подготовки к коллоквиуму необходимо проанализировать учебные материалы, ознакомившись с лекциями, учебниками и дополнительными источниками, акцентируя внимание на ключевых темах. Рекомендуется создать структурированные конспекты, выделяя основные идеи, термины и формулы.

Проект – исследовательская работа по дисциплине (модулю) и презентация результатов.

Для успешной подготовки к проекту рекомендуется: четко определить цели и задачи проекта; составить план работы, разбив проект на этапы с указанием сроков выполнения каждого из них; использовать разнообразные источники информации и инструменты для исследования темы; регулярно проверять прогресс и вносить коррективы в план, если это необходимо.

Самостоятельная работа – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских

заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Программирование»

Оценивание уровня учебных достижений обучающихся по дисциплине (модулю) осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме **зачета с оценкой**, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом, если иное не указано в силлабусе дисциплины:

Десятибалльная оценка	Пятибалльная оценка	Общая характеристика результата обучения по дисциплине (модулю)
10	Отлично	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину (модуль). Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
9	Отлично	
8	Отлично	
7	Хорошо	Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	
5	Удовлетворительно	Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при
4	Удовлетворительно	

Десятибалльная оценка	Пятибалльная оценка	Общая характеристика результата обучения по дисциплине (модулю)
		самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
3	Не сдан	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	
1	Не сдан	

Дисциплина (модуль) «Программирование» оценивается следующим образом:

Активность	Вес	Количество	Описание
Коллоквиум	30 %	1	Устные ответы на вопросы, список которых известен студенту заранее
Проект	40 %	1	Исследовательская работа по дисциплине (модулю) и презентация результатов
Зачет с оценкой	30 %	1	Письменная или устная работа над заданием, направленным на проверку полученных знаний и навыков по дисциплине (модулю)

Формула расчёта итоговой оценки по дисциплине (модулю) «Программирование»: $0,3 \times \text{коллоквиум} + 0,4 \times \text{проект} + 0,3 \times \text{зачет с оценкой}$.

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные вопросы к коллоквиуму

1. Сколько типов циклов в Python, если учитывать for и while? Ответ укажите в виде числа и слова "цикла".

Ответ: 2 цикла

2. Чему равно количество основных типов коллекций: список, кортеж, словарь, множество? Ответ укажите в виде числа и слова "типа".

Ответ: 4 типа

3. Сколько аргументов может принимать функция open() при работе с файлами, если указать имя файла, режим и кодировку? Ответ укажите в виде числа и слова "аргумента".

Ответ: 3 аргумента

4. Какое количество HTTP-методов используется в REST API, если учитывать GET и POST? Ответ укажите в виде числа и слова "метода".

Ответ: 2 метода

5. Сколько уровней вложенности может быть в словаре Python для хранения данных о студенте: имя, группа, оценки по предметам? Ответ укажите в виде числа и слова "уровня".

Ответ: 2 уровня

6. Сколько команд необходимо для загрузки изменений в GitLab: add, commit, push? Ответ укажите в виде числа и слова "команды".

Ответ: 3 команды

7. Сколько типов клавиатур поддерживает Telegram Bot API: ReplyKeyboard, InlineKeyboard? Ответ укажите в виде числа и слова "типа".

Ответ: 2 типа

8. Сколько компонентов включает простейший обработчик сообщений: получение, анализ, отправка ответа? Ответ укажите в виде числа и слова "компонента".

Ответ: 3 компонента

9. Сколько видов исключений обрабатывается в блоке try-except, если используются FileNotFoundError и ValueError? Ответ укажите в виде числа и слова "исключения".

Ответ: 2 исключения

10. Сколько файлов должно быть в проекте бота для модульности: основной, обработчики, конфиг, API-модуль? Ответ укажите в виде числа и слова "файла".

Ответ: 4 файла

Критерии оценивания участия в коллоквиуме

Точность ответа — корректное и полное формулирование определений, синтаксиса и принципов работы.

Понимание концепций — способность объяснить назначение конструкций (циклы, функции, Git-команды) на примерах.

Логика и структура ответа — последовательность, отсутствие противоречий, уместное использование терминологии.

Связь с практикой — умение привести примеры использования изученных конструкций в реальных задачах.

Реакция на уточняющие вопросы — способность адаптировать ответ при дополнительных вопросах преподавателя.

Примерное описание задания и критерии оценивания к проекту

Цель проекта:

разработать функционального телеграм-бота на языке Python, демонстрирующего навыки программирования, структурирования кода, работы с API, обработки ошибок и использования систем контроля версий.

Задачи / шаги выполнения:

1. Спроектировать сценарии взаимодействия пользователя с ботом (команды, диалоги, клавиатуры).
2. Реализовать бота с использованием библиотеки python-telegram-bot или аналога.
3. Интегрировать минимум одно внешнее API (например, погода, курсы валют, цитаты).
4. Организовать код по модульному принципу: отдельные файлы для обработчиков, конфигурации, API-запросов.
5. Реализовать обработку ошибок и логирование.
6. Разместить проект в репозитории GitLab с использованием Git: инициализация, коммиты, .gitignore, ветки.
7. Подготовить краткую документацию (в README): инструкция по запуску, список команд, описание API.

Критерии оценивания проекта

1. Функциональность бота — корректная работа всех заявленных команд и сценариев, отклик на запросы пользователя.
2. Структура и качество кода — модульность, читаемость, использование функций, документирование (docstrings, комментарии).
3. Интеграция API и обработка данных — успешные запросы к внешнему API, корректная обработка JSON-ответов.
4. Работа с Git и GitLab — наличие репозитория, осмысленные коммиты, использование веток, наличие .gitignore и README.

5. Обработка ошибок и устойчивость — перехват исключений, логирование, поведение бота при сбоях.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1	Какой метод используется для анализа логики программы перед написанием кода?	системный подход / структурирование задачи	УК-1
2	Как называется процесс проверки корректности алгоритма на примерах?	критический анализ / тестирование логики	УК-1
3	Какой принцип помогает объединить код из нескольких модулей в единый проект?	синтез информации / модульная архитектура	УК-1
4	Как называется подход, при котором вы разбиваете задачу на функции и этапы?	системное мышление / декомпозиция	УК-1
5	Какой инструмент помогает распределить время на выполнение лабораторных работ?	тайм-план / планирование	УК-6
6	Как называется подход, при котором вы разбиваете проект на этапы: прототип, тест, релиз?	поэтапная работа / тайм-менеджмент	УК-6
7	Какой навык помогает успеть сдать бота в срок?	самоорганизация / управление временем	УК-6
8	Как называется процесс постановки целей по освоению новых библиотек?	саморазвитие / траектория обучения	УК-6
9	Какой термин описывает ситуацию, когда код работает, но требует много ресурсов?	неэффективность / высокая стоимость выполнения	УК-10
10	Как называется компромисс между скоростью выполнения и сложностью кода?	trade-off / баланс производительности	УК-10
11	Какой показатель используется для оценки затрат на поддержку кода?	технический долг / cost of maintenance	УК-10
12	Как называется практика выбора оптимального алгоритма по времени выполнения?	экономное решение / оптимизация	УК-10
13	Какой оператор в Python используется для повторения блока кода?	for / while	ОПК-3
14	Какой тип данных в Python используется для хранения пар ключ-значение?	словарь / dictionary	ОПК-3
15	Какая команда Git сохраняет изменения в локальном репозитории?	commit / git commit	ОПК-3
16	Как называется функция, обрабатывающая входящие сообщения от пользователя?	обработчик / handler	ОПК-3
17	Какой формат используется для передачи данных в API?	JSON / json	ПК-2
18	Как называется процесс оформления кода с комментариями и документацией?	создание контента / документирование	ПК-2
19	Какой файл используется для описания зависимостей проекта?	requirements.txt / зависимостис	ПК-2
20	Как называется практика создания README-файла для проекта?	разработка описания / техническая документация	ПК-2

21	Как называется процесс обновления кода в репозитории?	актуализация / обновление репозитория	ПК-3
22	Какой термин описывает проверку кода на соответствие стандартам?	нормативный контроль / код-ревью	ПК-3
23	Как называется система регулярного тестирования кода при новых коммитах?	CI / непрерывная интеграция	ПК-3
24	Как называется работа с документацией API при разработке бота?	управление информацией / источниковедение	ПК-3