



УТВЕРЖДАЮ
Заместитель директора Ур СЭИ (филиал)
ОУП ВО «АТиСО» по УВР

_____ О.В. Зубкова

« 17 » февраля 2026 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ (МОДУЛЯ)
ОПЦ.04 ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ПРОФЕССИОНАЛЬНОЙ
ДЕЯТЕЛЬНОСТИ

СПЕЦИАЛЬНОСТЬ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО
ОБРАЗОВАНИЯ

09.02.11 Разработка и управление программным обеспечением
(код профессии, специальности СПО)

Программист
(наименование квалификации)

Кафедра: Информатики и математики

Разработчик программы:
Зав. кафедрой

Маскайкин Е.П.

17 февраля 2026 г.

Оглавление

1. ОБЛАСТЬ ПРИМЕНЕНИЯ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОГО ПРЕДМЕТА, КУРСА, ДИСЦИПЛИНЫ (МОДУЛЯ)	3
2. ЦЕЛИ И ЗАДАЧИ УЧЕБНОЙ ДИСЦИПЛИНЫ.....	3
3. МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ПРОГРАММЫ ПОДГОТОВКИ СПЕЦИАЛИСТОВ СРЕДНЕГО ЗВЕНА	3
4. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ:.....	4
5. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	5
6. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	10
7. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	11
8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ	12
9. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ СО.02.02. ИНФОРМАТИКА	12

1. ОБЛАСТЬ ПРИМЕНЕНИЯ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОГО ПРЕДМЕТА, КУРСА, ДИСЦИПЛИНЫ (МОДУЛЯ)

Рабочая программа учебной дисциплины «Информационные технологии в профессиональной деятельности» является частью программы подготовки специалистов среднего звена (ППССЗ) в соответствии с ФГОС СПО по специальности 09.02.11 Разработка и управление программным обеспечением, квалификация – программист.

Программа может быть использована в дополнительном профессиональном образовании и профессиональной подготовке специалистов в области информационных технологий.

2. ЦЕЛИ И ЗАДАЧИ УЧЕБНОЙ ДИСЦИПЛИНЫ

Цель изучения: формирование у обучающихся практических навыков применения современных информационных технологий для решения профессиональных задач в области разработки и управления программным обеспечением, включая использование инструментов искусственного интеллекта, систем контроля версий, облачных сервисов, сред разработки и инструментов цифровой безопасности.

Задачи:

- освоить современные цифровые инструменты и сервисы для разработки ПО (IDE, Git, GitHub/GitLab, облачные платформы);
- научиться применять инструменты искусственного интеллекта для генерации, рефакторинга и документирования кода;
- сформировать навыки работы с системами контроля версий и командной разработки;
- освоить основы документирования проектов с использованием Markdown и создания технической документации;
- изучить принципы работы с облачными сервисами (IaaS, PaaS, SaaS) и развёртывания приложений;
- развить навыки использования форматов данных (JSON, YAML, XML) и работы с API;
- сформировать культуру цифровой гигиены и кибербезопасности в профессиональной деятельности.

3. МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ПРОГРАММЫ ПОДГОТОВКИ СПЕЦИАЛИСТОВ СРЕДНЕГО ЗВЕНА

Учебная дисциплина «Информационные технологии в профессиональной деятельности» относится к общепрофессиональному циклу (ОПЦ.04) специальности 09.02.11 Разработка и управление программным обеспечением.

Дисциплина базируется на знаниях, умениях и навыках, полученных при изучении информатики в школе, а также параллельно с дисциплинами «Операционные системы и среды», «Архитектура аппаратных средств».

Знания, умения и навыки, полученные при изучении данной дисциплины, являются основой для освоения профессиональных модулей, связанных с разработкой баз данных, программных модулей, бизнес-приложений, а также для практической деятельности программиста.

4. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ:

В результате освоения дисциплины обучающийся должен овладеть следующими общими компетенциями:

Код	Формулировка компетенции
ПК 1.2	Разрабатывать объекты баз данных в соответствии с результатами анализа предметной области.
ПК 2.2	Разрабатывать модули программного обеспечения.
ПК 3.2	Разрабатывать бизнес-приложения.

Таким образом, в результате освоения дисциплины обучающийся должен:
уметь:

- У1 – определять задачи для поиска информации, планировать процесс поиска, выбирать необходимые источники информации;
- У2 – применять средства информационных технологий для решения профессиональных задач (IDE, Git, облачные сервисы);
- У3 – использовать современное программное обеспечение в профессиональной деятельности, в том числе цифровые средства;
- У4 – разрабатывать объекты баз данных (таблицы, индексы, ограничения) с использованием средств разработки и SQL;
- У5 – оптимизировать запросы к базе данных для повышения производительности;
- У6 – разрабатывать модули программного обеспечения на различных языках программирования с использованием современных инструментов;
- У7 – отлаживать и тестировать разработанные модули;
- У8 – применять паттерны проектирования и структуры данных;
- У9 – разрабатывать клиентскую и серверную части веб-приложений;
- У10 – использовать языки разметки и программирования для веб-разработки;
- У11 – оформлять код в соответствии со стандартами и документацию к проекту;
- У12 – работать с системами контроля версий (Git) и платформами совместной разработки (GitHub/GitLab).

знать:

- 31 – номенклатуру информационных источников, применяемых в профессиональной деятельности;
- 32 – современные средства и устройства информатизации, порядок их применения;
- 33 – программное обеспечение в профессиональной деятельности, в том числе цифровые средства;
- 34 – основы реляционной модели данных;
- 35 – язык SQL и его основные команды;
- 36 – принципы нормализации баз данных;
- 37 – языки программирования и технологии для реализации модулей;
- 38 – паттерны проектирования и структуры данных;
- 39 – методы оптимизации кода и алгоритмов;
- 310 – языки программирования и разметки для веб-разработки;

- 311 – принципы работы объектной модели веб-приложений;
- 312 – технологии клиент-серверного взаимодействия;
- 313 – базовые принципы работы систем контроля версий и CI/CD;
- 314 – основы работы с облачными сервисами и контейнеризацией.

5. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

5.1 Объем учебной дисциплины и виды учебной работы

Вид учебной работы	3 семестр	Итого
Максимальная учебная нагрузка (всего)	32	32
Обязательная аудиторная учебная нагрузка (всего)	30	30
в том числе: лекции	10	10
практические занятия	20	20
Самостоятельная работа обучающегося (всего)	2	2
Итоговая аттестация	Дифференцированный зачет (3 семестр)	Дифф. зачет

5.2. Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала, практические занятия, самостоятельная работа обучающихся	Объем часов	Уровень освоения
Раздел 1. Информационные технологии		32	
Тема 1.1. Искусственный интеллект как инструмент программиста	Лекция: ИИ и LLM: зачем они нужны разработчику. Обзор популярных ИИ-инструментов (GitHub Copilot, ГигаЧат, Codeium). ИИ и написание кода: кейсы и ограничения. Использование ИИ для генерации тестов, SQL-запросов. Промпт-инжиниринг. Ревью кода с ИИ. Генерация документации. Этические аспекты. Практические занятия: Подключение и использование нейросети для генерации кода. Генерация автотестов на Python по описанию задачи. Написание SQL-запросов через Copilot. Рефакторинг кода с объяснением шагов. Генерация комментариев к функциям и классам. Сравнение работы нескольких ИИ-инструментов. Создание readme-файла проекта через ИИ. Превращение баг-репорта в список задач. Разработка промптов для сложных запросов. Самостоятельная работа: изучение документации по API ИИ-сервисов.	8 (2 лекции + 6 практик)	2
Тема 1.2. Git и Markdown в командной разработке	Лекция: Контроль версий: зачем нужен Git. Базовые команды, концепция веток. Ветки, мержи, pull request и конфликты. GitHub/GitLab: интерфейс, CI, багтрекеры. Markdown: синтаксис, структура, назначение. Документирование	8 (2 лекции + 6 практик)	2

Наименование разделов и тем	Содержание учебного материала, практические занятия, самостоятельная работа обучающихся	Объем часов	Уровень освоения
	API в Markdown. README.md как витрина проекта. Использование GitHub Pages и Wiki. Рецензирование кода через pull request. Практические занятия: Создание и клонирование репозитория. Ведение истории коммитов и работа с ветками. Конфликт и его разрешение. Настройка CI в GitHub Actions. Создание красивого README.md. Использование маркдауна для changelog. Описание API-интерфейса в markdown. Работа с pull request и ревью кода. Создание и публикация проекта на GitHub Pages. Создание вики-проекта и структуры документации. Самостоятельная работа: подготовка репозитория для учебного проекта.		
Тема 1.3. Облачные сервисы и инструменты разработчика	Лекция: Основы работы с облаками: IaaS, PaaS, SaaS. Обзор российских облачных платформ (Яндекс Облако, VK Cloud, Selectel). Хранилище, вычисления, базы данных в облаке. Развёртывание приложения на облачном сервере. Terraform / IaC: автоматизация инфраструктуры. GitLab CI/CD + облако. Облачные IDE (Replit, GitHub Codespaces). S3-хранилище и автоматизация бэкапов. Логирование и мониторинг в облаке. Безопасность облачных сред. Практические занятия: Регистрация и запуск виртуальной машины в Яндекс.Облаке. Развёртывание Python-приложения на облачном сервере. Использование S3-хранилища для логов. Настройка CI/CD-пайплайна для загрузки файлов. Подключение к облачной базе данных. Использование облачной IDE для командного проекта. Создание YAML-манифеста Terraform. Настройка доступа к bucket'у. Интеграция с логами и алертами. Аудит безопасности облачного проекта. Самостоятельная работа: изучение документации выбранной облачной платформы.	8 (2 лекции + 6 практик)	2

Наименование разделов и тем	Содержание учебного материала, практические занятия, самостоятельная работа обучающихся	Объем часов	Уровень освоения
Тема 1.4. Цифровые инструменты и экосистема разработчика	Лекция: IDE, расширения, сборщики: VS Code, Bash и командная строка как инструмент. Утилиты curl, wget, ping, telnet. Форматы данных: JSON, XML. Конфигурационные файлы и шаблоны. DevTools в браузере и веб-отладка. Task-менеджеры и трекеры. Работа с docker-образами. Инструменты тестирования API: Postman. Автоматизация повседневных задач. Практические занятия: Работа в VS Code: настройка расширений. Написание bash-скрипта для автоматизации. Отправка API-запроса через curl и Postman. Разбор JSON-структуры и валидация. Написание dockerfile и сборка образа. Использование DevTools для анализа сайта. Создание задачи и доски в Trello. Отладка API на реальном сервисе. Настройка git hooks и lint-автоматизации. Создание шаблона конфига в YAML. Самостоятельная работа: настройка рабочей среды разработчика.	4 (2 лекции + 2 практики)	2
Тема 1.5. Кибербезопасность и цифровая гигиена ИТ-специалиста	Лекция: Угрозы в разработке: инъекции, XSS, MITM. Безопасные пароли, ключи, доступы. Работа с .env-файлами и секретами. Проверка зависимостей: Snyk, Dependabot. Шифрование, хеширование и токены. VPN, SSH и туннелирование. Анонимизация и защита данных. Правила цифровой гигиены и GDPR. Атаки на open-source проекты. Повседневная безопасность в DevOps. Практические занятия: Настройка SSH-ключей и безопасного подключения. Работа с .env-файлом в проекте. Сканирование зависимостей с Snyk. Пример XSS-атаки и защита от неё. Хеширование строки и проверка целостности. Шифрование данных с помощью openssl. Работа с GitHub Secrets и CI. Создание VPN-соединения. Формирование чек-листа цифровой	4 (2 лекции + 2 практики)	2

Наименование разделов и тем	Содержание учебного материала, практические занятия, самостоятельная работа обучающихся	Объем часов	Уровень освоения
	гигиены. Анализ утечек и проверка паролей. Самостоятельная работа: подготовка чек-листа безопасности для личного проекта.		
Всего		32	

6. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

6.1. Требования к минимальному материально-техническому обеспечению

Реализация программы требует наличия лаборатории «Информационных технологий и архитектуры аппаратных средств», оснащённой в соответствии с требованиями ФГОС.

Оборудование лаборатории:

- рабочие места по количеству обучающихся (компьютеры с доступом в Интернет);
- рабочее место преподавателя;
- мультимедийное оборудование (проектор, экран);
- программное обеспечение: современные IDE (VS Code, PyCharm), Git, браузеры, средства для работы с облачными сервисами, Docker, Postman.

Технические средства обучения:

- компьютер, мультимедийный проектор, экран;
- доступ к сети Интернет для использования электронных образовательных ресурсов.

6.2. Информационное обеспечение обучения

Основная литература

1. Информационные технологии : учебник для среднего профессионального образования / В. В. Трофимов, О. П. Ильина, В. И. Кияев, Е. В. Трофимова ; ответственный редактор В. В. Трофимов. — Москва : Издательство Юрайт, 2026. — 546 с. — (Профессиональное образование). — ISBN 978-5-534-18341-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/589573>
2. Шеманаева, Л. И. Информационные технологии в профессиональной деятельности : учебно-методическое пособие / Л. И. Шеманаева. — Москва : Директ-Медиа, 2025. — 156 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=723327>. — Библиогр.: с. 139-143. — ISBN 978-5-4499-5274-5. — DOI 10.23681/723327. — Текст : электронный.
3. Куприянов, Д. В. Информационное обеспечение профессиональной деятельности : учебник и практикум для среднего профессионального образования / Д. В. Куприянов. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 236 с. — (Профессиональное образование). — ISBN 978-5-534-20826-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/584373>

Дополнительная литература

4. Мамонова, Т. Е. Информационные технологии. Лабораторный практикум : учебное пособие для среднего профессионального образования / Т. Е. Мамонова. — Москва : Издательство Юрайт, 2023. — 178 с. — (Профессиональное образование). — ISBN 978-5-534-07791-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/516847>.
5. Гаврилов, М. В. Информатика и информационные технологии : учебник для среднего профессионального образования / М. В. Гаврилов, В. А. Климов. — 7-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 307 с. — (Профессиональное образование).

образование). — ISBN 978-5-534-16997-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/600312>.

6. Муромцев, В. В. Информационные технологии в профессиональной деятельности : учебник и практикум : [16+] / В. В. Муромцев, А. В. Муромцева. — Москва ; Вологда : Инфра-Инженерия, 2023. — 384 с. : ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=725647>. — библиогр. в кн. — ISBN 978-5-9729-1299-5. — Текст : электронный.

Ресурсы информационно-телекоммуникационной сети «Интернет», необходимые для освоения дисциплины

<https://github.com/>
<https://docs.gitlab.com/>
<https://cloud.yandex.ru/docs>
<https://code.visualstudio.com/docs>
<https://www.docker.com/get-started>

7. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Рабочая программа учебной дисциплины обеспечена оценочными материалами для проведения текущего контроля и промежуточной аттестации.

Оценочные материалы включают в себя тестовые задания, практические задания, вопросы к дифференцированному зачёту.

Текущий контроль осуществляется в форме проверки выполнения практических работ, устных опросов, тестирования, защиты отчётов по самостоятельной работе.

Промежуточная аттестация – дифференцированный зачёт в 3 семестре (включает теоретические вопросы и практическое задание).

Примерный перечень вопросов к дифференцированному зачёту:

1. Назначение и возможности ИИ-инструментов для разработчика.
2. Промпт-инжиниринг: основные принципы формулировки запросов.
3. Контроль версий: базовые команды Git, работа с ветками.
4. Pull request и ревью кода: процесс и правила.
5. Markdown: синтаксис, использование для документации.
6. Облачные сервисы: модели IaaS, PaaS, SaaS, примеры.
7. Развёртывание приложения в облаке.
8. CI/CD: основные понятия, настройка пайплайна.
9. Форматы данных: JSON, YAML, XML, сравнение.
10. Инструменты тестирования API (Postman, curl).
11. Docker: контейнеризация, написание Dockerfile.
12. Основы кибербезопасности: угрозы и защита.
13. Работа с секретами в проекте (.env, GitHub Secrets).
14. SSH-ключи и безопасные подключения.
15. Цифровая гигиена: правила безопасной работы разработчика.

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

По выполнению лекционных занятий:

- вести конспект, фиксируя ключевые понятия, инструменты и их применение;
- после лекции прорабатывать материал по учебнику и документации;
- выделять неясные вопросы для обсуждения с преподавателем.

По выполнению практических занятий:

- выполнять лабораторные работы, следуя инструкциям;
- фиксировать результаты, делать выводы;
- использовать документацию инструментов.

По самостоятельной работе:

- выполнять домашние задания, готовиться к практическим работам;
- изучать дополнительную литературу и официальную документацию;
- готовить проекты для портфолио.

По подготовке к дифференцированному зачёту:

- повторить теоретические положения по разделам;
- проработать типовые практические задания;
- использовать вопросы для самопроверки.

9. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Фонд оценочных средств включает задания разных типов, проверяющих как знание теории, так и умение применять математические методы.

Тип задания	Инструкция	Пример задания
Задание закрытого типа на установление соответствия (Знать)	Прочитайте текст и установите соответствие между ИИ-инструментами и их назначением.	Установите соответствие между ИИ-инструментом и его функцией: А) ChatGPT; Б) GitHub Copilot; В) Codeium; Г) Snyk. 1) Генерация кода на основе контекста; 2) Беседа и ответы на вопросы; 3) Проверка безопасности зависимостей; 4) Альтернативный помощник для написания кода.
Задание закрытого типа на установление соответствия (Уметь)	Прочитайте текст и установите соответствие между	Установите соответствие между командой Git и её действием: А) git init; Б) git add; В) git commit; Г) git push. 1) Отправка изменений в удалённый репозиторий; 2) Инициализация репозитория;

Тип задания	Инструкция	Пример задания
	командами Git и их действиями.	3) Фиксация изменений с комментарием; 4) Добавление файлов в индекс.
Задание закрытого типа на установление последовательности (Знать)	Прочитайте текст и установите правильную последовательность этапов работы с Git в командной разработке.	Расположите в правильном порядке этапы работы с Git: 1) Создание ветки; 2) Коммит изменений; 3) Pull request; 4) Ревью и мерж.
Задание закрытого типа на установление последовательности (Уметь)	Прочитайте текст и установите правильную последовательность создания README.md.	Расположите в правильном порядке разделы README.md: 1) Описание проекта; 2) Инструкция по установке; 3) Инструкция по запуску; 4) Примеры использования; 5) Лицензия.
Задание с выбором одного верного ответа (Знать)	Прочитайте текст, выберите правильный ответ и запишите аргументы.	Какой формат данных чаще всего используется для конфигурационных файлов в облачных сервисах? 1) XML; 2) JSON; 3) YAML; 4) CSV.
Задание с выбором одного верного ответа (Уметь)	Прочитайте текст, выберите правильный ответ и запишите аргументы.	Какая команда используется для создания нового репозитория в Git? 1) git clone; 2) git init; 3) git create; 4) git new.
Задание с выбором нескольких вариантов (Знать)	Прочитайте текст, выберите все правильные ответы и запишите обоснование.	Какие из перечисленных сервисов относятся к облачным платформам IaaS? 1) Яндекс Облако; 2) VK Cloud; 3) ChatGPT; 4) AWS.
Задание с выбором нескольких вариантов (Уметь)	Прочитайте текст, выберите все правильные ответы и запишите обоснование.	Какие инструменты можно использовать для тестирования API? 1) Postman; 2) curl; 3) Git; 4) Docker.

Тип задания	Инструкция	Пример задания
Задание комбинированное с выбором одного верного ответа и обоснованием (Знать)	Прочитайте текст, выберите правильный ответ и запишите развёрнутое обоснование.	Что такое CI/CD? 1) Непрерывная интеграция и доставка; 2) Центральное хранилище данных; 3) Система контроля версий; 4) Облачная платформа.
Задание комбинированное с выбором одного верного ответа и обоснованием (Уметь)	Прочитайте текст, выберите правильный ответ и запишите развёрнутое обоснование.	Какой способ подключения к удалённому серверу является наиболее безопасным? 1) Telnet; 2) SSH; 3) HTTP; 4) FTP.
Задание комбинированное с выбором нескольких вариантов и обоснованием (Знать)	Прочитайте текст, выберите все правильные ответы и запишите развёрнутое обоснование.	Какие из перечисленных технологий используются для контейнеризации приложений? 1) Docker; 2) Kubernetes; 3) Git; 4) Jira.
Задание комбинированное с выбором нескольких вариантов и обоснованием (Уметь)	Прочитайте текст, выберите все правильные ответы и запишите развёрнутое обоснование.	Какие меры относятся к цифровой гигиене разработчика? 1) Использование сложных паролей; 2) Хранение секретов в .env; 3) Работа без VPN; 4) Регулярное обновление зависимостей.
Задание открытого типа с развёрнутым ответом (Знать)	Прочитайте текст и запишите развёрнутый обоснованный ответ.	Объясните принцип работы Git и назовите основные преимущества использования системы контроля версий.
Задание открытого типа с развёрнутым ответом (Уметь)	Прочитайте текст и запишите развёрнутый обоснованный ответ.	Опишите процесс создания и публикации проекта на GitHub Pages с использованием Markdown.
Задание открытого типа с	Прочитайте текст и запишите развёрнутый	Сравните форматы JSON и YAML: преимущества и недостатки.

Тип задания	Инструкция	Пример задания
развёрнутым ответом (Знать)	обоснованный ответ.	
Задание открытого типа с развёрнутым ответом (Уметь)	Прочитайте текст и запишите развёрнутый обоснованный ответ.	Как использовать ИИ-инструменты для генерации тестов? Приведите пример промпта.
Практическое задание (Уметь)	Выполните задание и запишите результат.	Создайте репозиторий на GitHub, добавьте файл README.md с описанием проекта, выполните коммит и отправьте изменения.
Практическое задание (Уметь)	Выполните задание и запишите результат.	Напишите Dockerfile для простого Python- приложения, соберите образ и запустите контейнер.
Практическое задание (Уметь)	Выполните задание и запишите результат.	Используя Postman, отправьте GET-запрос к открытому API (например, https://jsonplaceholder.typicode.co m/posts) и проанализируйте ответ.
Практическое задание (Уметь)	Выполните задание и запишите результат.	Настройте CI/CD пайплайн в GitHub Actions для автоматической проверки кода при пуше в репозиторий.