

Отчёт по 2 этапу проекта

Сайт научного работника

Вишняков Родион Сергеевич

Содержание

1	Цель работы	5
2	Выполнение работы	6
3	Выводы	10

Список иллюстраций

2.1	Файл об авторе	7
2.2	Файл для поста	8
2.3	Файл для публикации	9

Список таблиц

1 Цель работы

Добавить к сайту данные о себе.

2 Выполнение работы

Заполняю файл с данными о владельце сайта.

О себе

👋 Привет! Меня зовут Родион Вишняков. Я учусь на 1 курсе РУДН по направлению «Прикладная информатика».

Что меня вдохновляет?

- 💡 Идеи цифровой трансформации
- 📊 Анализ данных и визуализация
- 🌐 Разработка веб-приложений
- 🔧 Практическое применение IT в бизнесе

Чем я занимаюсь?

- 📖 Осваиваю языки программирования и базы данных
- ⚙️ Учусь автоматизировать процессы
- 📝 Разрабатываю учебные проекты и делюсь знаниями

Готов к новым идеям, сотрудничеству и интересным задачам! 🚀

|

Рис. 2.1: Файл об авторе

Заполняю файл с текстом поста.

#17 Неделя с 20 по 26 мая

Эта неделя прошла под знаком активной подготовки к сессии.

Вот что удалось сделать:





-  **Программирование**: разобрали рекурсию и написали свои первые реализации сортировки слиянием. Получилось не сразу, но после пар с практикой стало яснее.
-  **Математический анализ**: закончили тему пределов и перешли к производным. Решали задачи с параметрами и делали упор на графическое представление функций.
-  **Математическая логика**: строили логические выражения, познакомились с нормальными формами и сделали первое домашнее по преобразованию формул.
-  **Физкультура**: сдавали бег 1000 м и делали замеры на гибкость и силу. Участвовал в командной эстафете – неплохой результат!

Рис. 2.2: Файл для поста

Заполняю файл с текстом публикации.

🚧 Что такое CI/CD?

CI/CD – это два ключевых компонента современной DevOps-культуры:

- **CI (Continuous Integration)** – непрерывная интеграция: автоматическая сборка и тестирование кода каждый раз, когда в репозиторий вносятся изменения.
- **CD (Continuous Delivery / Deployment)** – непрерывная доставка или развёртывание: автоматическая доставка обновлений на тестовые или продуктивные серверы без участия разработчика.

Идея в том, чтобы **ускорить цикл поставки программного обеспечения**, минимизировать ошибки и сделать процесс выпуска новых версий максимально надёжным и предсказуемым.

🎯 Зачем это нужно?

CI/CD помогает:

- 🚩 **Обнаруживать ошибки на ранних этапах** – за счёт автоматического тестирования после каждого коммита.
- 🧠 **Сократить ручную работу** – развёртывание, сборка и публикация происходят автоматически.
- 🔄 **Обеспечить повторяемость процессов** – одинаковый результат на всех этапах пайплайна.
- ⚡ **Ускорить вывод продукта на рынок** – за счёт быстрого цикла изменений и автоматизации.
- 🌱 **Упростить командную работу** – каждый разработчик видит, работает ли его код, и не ломает ли чужой.

📦 Основные компоненты CI/CD

Процесс CI/CD строится из следующих этапов:

1. **Push/Commit** – разработчик отправляет изменения в репозиторий.
2. **Build (CI)** – система собирает проект, устанавливает зависимости.
3. **Test (CI)** – запускаются юнит-тесты, интеграционные или end-to-end тесты.
4. **Package** – при успешной сборке и тестировании создаётся артефакт (например, Docker-образ).

Рис. 2.3: Файл для публикации

Перекомпилирую сайт

3 Выводы

Добавили к сайту данные о себе.