**КИСЛЯКОВ СЕРГЕЙ ВИКТОРОВИЧ**

**Python-разработчик (Junior)**

**[Контакты]**

**Телефон:** +79003008998

**Telegram:** t.me/kislyakow

**Email:** kislyakov84@yandex.ru

**GitHub:** [github.com/kislyakov84](https://github.com/kislyakov84/)

**Местонахождение:** Острогожск, Воронежская область

**Релокация:** Да

**Удаленная работа:** Да

**[Цель]**

Мотивированный начинающий Python-разработчик с опытом стажировки и практическими навыками самостоятельного обучения. Ищу позицию Junior Python Developer для применения и развития знаний в области full-stack разработки. Стремлюсь к профессиональному росту в динамичной команде.

**[Навыки]**

**• Программирование:** Python (ООП, основы, асинхронное программирование)

**• Фреймворки:** Django, FastAPI, Sanic (CRUD, аутентификация, обработка ошибок)

**• Базы данных:**

**- Реляционные:** PostgreSQL/MySQL/SQLite (SELECT, INSERT, UPDATE, JOIN)

**- ORM:** SQLAlchemy (Core, async ORM)

**• Инструменты:**

- Git/GitHub (clone, add, commit, push, pull)

  - Linux/Unix (командная строка)

  - Контейнеризация: Docker/Docker Compose

- Конфигурация: .env файлы

  - Форматирование кода: black, isort

**• Библиотеки:**

- Веб/API:

    - HTTP-клиенты: requests, httpx (асинхронный)

    - Аутентификация: passlib (bcrypt), python-jose (JWT)

  - Telegram: python-telegram-bot

  - Автоматизация: pyautogui, pyperclip

  - Computer Vision: Pillow (Image), Tesseract, Mistral OCR

  - Облачные сервисы: google.cloud

  - Вспомогательные: logging

**• Дополнительно:**

- HTML/CSS (базовые знания)

- REST API (разработка и интеграция)

- Обработка вебхуков (проверка подписи)

- Парсинг веб-данных

- Асинхронное взаимодействие систем

**[Опыт работы]**

DOBROCORP | Стажер Python-разработчик

Март 2025 - настоящее время | Удаленно (Черногория)

Проект: Автоматизация ставок для букмекерских платформ

Стек: Python, FastAPI, Tesseract OCR, Google Vision, Telegram API

- Адаптация парсеров ставок под букмекерские сайты:

  - Анализ структуры сайтов и поиск API эндпоинтов

  - Реализация сбора live-ставок с матчей в реальном времени

- Модернизация Telegram-бота для автоматизации ставок:

  - Интеграция систем распознавания текста (Google Vision → Mistral OCR + Tesseract)

  - Адаптация логики бота под верстку различных брокеров

- Разработка инструмента оптимизации OCR:

  - [Tesseract OCR Preprocessing Tuner](https://github.com/kislyakov84/tesseract\_tuner)

  - Автоматический подбор параметров обработки изображений

  - Тестирование комбинаций фильтров (масштабирование, контраст, резкость)

**[Проекты]**

**[Sanic Payment API (Fullstack: Sanic + PostgreSQL)]** (https://github.com/kislyakov84/sanic\_payment\_api)

*\*Python, Sanic, PostgreSQL, SQLAlchemy (async), Alembic, Docker Compose, JWT\**

- Разработка асинхронного REST API для управления пользователями, счетами и платежами.

- Реализация JWT-аутентификации для пользователей и администраторов.

- Настройка Docker Compose для быстрого развертывания приложения и базы данных.

- Управление миграциями схемы базы данных с помощью Alembic.

- Реализация эндпоинта для обработки вебхука от платежной системы с проверкой SHA256 подписи.

**[Tesseract OCR Preprocessing Tuner] (**https://github.com/kislyakov84/tesseract\_tuner**)**

*\*Python, Tesseract OCR, Pillow (Image)\**

- Инструмент для оптимизации параметров предобработки изображений

- Автоматический перебор комбинаций:

  - Масштабирование изображения

  - Регулировка контраста и резкости

  - Адаптивная бинаризация и размытие

- Оценка результатов OCR по целевым текстам

- Специализирован для обработки скриншотов интерфейсов букмекерских контор

**[Telegram Blog Bot (Fullstack: FastAPI + SQLite)] (**https://github.com/kislyakov84/blogtron**)**

*\*Python, FastAPI, SQLite, SQLAlchemy, python-telegram-bot, JWT, httpx\**

- Разработка full-stack приложения: RESTful API (админка) и Telegram-бота

- Реализация CRUD операций с JWT-аутентификацией

- Интеграция бота с API через асинхронные HTTP-запросы

- Использование SQLAlchemy ORM для работы с БД

- Настройка автоматического форматирования кода (black, isort)

- Обработка ошибок с информативными сообщениями пользователю

**[Образование]**

Воронежский экономико-правовой институт

Специальность: Экономист-Информатик | 2002-2007

Самообразование | 2023 - настоящее время

• Курсы: Python Pro (Stepik), Веб-разработка на Django (Udemy)

• Изучаемые технологии: FastAPI, Docker, асинхронное программирование