1. Опыт разработки на Python:  
   - Глубокое понимание языка Python, включая ООП (наследование, полиморфизм, инкапсуляция, абстракция).  
   - Умение работать с типами и аннотациями (модуль `typing`).  
   - Понимание принципов сериализации и десериализации данных (например, с использованием `pydantic`).
2. Веб-фреймворки:  
   - Опыт работы с Flask (создание REST API, работа с middleware, обработка запросов и ответов).  
   - Опыт работы с Django (ORM, миграции, шаблоны, админка, REST Framework).
3. Работа с базами данных:  
   - Уверенное знание SQL (написание запросов, оптимизация, индексы).  
   - Опыт работы с ORM (SQLAlchemy, Django ORM).  
   - Понимание принципов работы с реляционными и нереляционными базами данных.
4. Парсинг данных и автоматизация:  
   - Опыт работы с библиотекой Beautiful Soup для парсинга HTML/XML.  
   - Опыт работы с Playwright для автоматизации тестирования и парсинга.
5. Управление зависимостями и окружением:  
   - Опыт использования Poetry для управления зависимостями и сборки проектов.  
   - Знание Conda и Miniconda для управления виртуальными окружениями.
6. Контейнеризация:  
   - Уверенное владение Docker (создание Dockerfile, docker-compose, работа с контейнерами).
7. Работа с Linux:  
   - Хорошее знание Linux (работа с командной строкой, настройка окружения, управление процессами).
8. Работа с API и сетевыми протоколами:  
   - Понимание принципов работы HTTP (методы, статусы, заголовки, REST).  
   - Умение работать с WebSockets, TCP/IP, UDP.  
   - Умение проектировать и интегрировать API.
9. Разработка Telegram-ботов:  
   - Опыт создания Telegram-ботов с использованием библиотек (например, python-telegram-bot, aiogram, `Pyrogram`).
10. Чтение и анализ чужого кода:  
    - Умение быстро разбираться в чужом коде, находить и исправлять ошибки
11. Асинхронные задачи и очереди:  
    - Опыт работы с Celery и RabbitMQ.  
    - Понимание принципов асинхронного выполнения задач
12. Асинхронное программирование:  
    - Опыт работы с asyncio, aiohttp или другими асинхронными библиотеками.  
    - Понимание работы event loop, корутин и Future-объектов.
13. Многопоточность и параллелизм:  
    - Знание модулей threading и multiprocessing.  
    - Понимание ограничений GIL (Global Interpreter Lock) в Python и способов их обхода
14. Безопасность:  
    - Базовое понимание уязвимостей веб-приложений (SQL-инъекции, XSS, CSRF, JWT-безопасность).  
    - Опыт работы с аутентификацией и авторизацией (OAuth2, JWT, Django Auth, Flask-Security).
15. Кэширование:  
    - Базовое понимание работы с Redis или Memcached для кэширования данных.  
    - Интеграция кэша в веб-приложения (например, через Flask-Caching, `Django-Redis`).
16. Архитектура и паттерны:  
    - Знание принципов SOLID, паттернов проектирования (Фабрика, Наблюдатель, Стратегия и др.).  
    - Опыт построения приложений с чистой архитектурой (Clean Architecture, DDD).
17. Работа с брокерами сообщений:  
    - Опыт работы с Celery и RabbitMQ.  
    - Понимание моделей обмена сообщениями (pub/sub, message queues).
18. Документирование:  
    - Умение описывать API с помощью Swagger/OpenAPI.
19. DevOps-навыки:  
    - Опыт работы с Grafana для создания дашбордов и мониторинга.  
    - Знание Ansible для автоматизации развертывания.  
    - Настройка веб-серверов (Nginx, CaddyServer).
20. Интеграция искусственного интеллекта:  
    - Опыт работы с TensorFlow и YOLO.  
    - Умение интегрировать любые другие модели искусственного интеллекта.
21. Интеграция платежных систем:  
    - Опыт работы с платежными системами (Stripe, PayPal, Яндекс.Касса и др.).  
    - Настройка и обработка платежей, работа с вебхуками.
22. Работа с графиками и визуализацией:  
    - Базовые знания по работе с библиотеками для визуализации данных (Matplotlib, Plotly, Seaborn).  
    - Опыт работы с Grafana для создания дашбордов и мониторинга.

23. Работа по методологии Kanban:  
- Опыт работы с канбан-досками (Trello, Jira, Asana).  
- Умение эффективно управлять задачами и приоритетами в рамках Agile-подходов.