Теоретические вопросы:

1. Как языком программирования Python является: компилируемым или интерпретируемым? Как выполняется исходный код Python? В чем разница между высокоуровневыми и низкоуровневыми языками?

2. Какие типы данных есть в Python для хранения коллекции элементов? Расскажите про их отличия между друг другом.

3. Что такое парадигмы в программировании? Расскажите про императивную парадигму.

4. Что такое парадигмы в программировании? Расскажите про декларативную парадигму.

5. Что такое парадигмы в программировании? Расскажите про объектно-ориентированное программирование (ООП).

6. Опишите структуру ООП. Назовите основные принципы ООП.

7. Назовите ключи модификаторов доступа для работы с файлами, опишите, какой доступ каждый из ключей предоставляет. Какая функция используется для работы с файлом и какие обязательные аргументы необходимо в нее передать для работы с файлом?

8. Что такое менеджер контекста в Python? Для чего он используется? Опишите структуру использования менеджера контекста.

9. Что такое json? Расскажите про сериализацию и десериализацию.

10. Какие виды программирования по выполнению кода существуют? Расскажите про разницу между ними.

11. Что такое многопоточность? Что такое поток?

12. Что такое интерпретатор? Как он работает?

13. Для чего используется механизм GIL в интерпретаторе CPython? Как он мешает, при работе с многопоточностью?

14. Что такое состояние гонки? Опишите пример проявления данного состояния.

15. Что такое взаимоблокировка? Опишите пример ее проявления.

16. Какие 2 вида нагрузок на программу существуют? Назовите, где подойдет многопоточность, для ускорения работы программы, а где многопроцессорность.

17. Какие виды блокировок существуют в Python? Для чего они используются?

18. Назовите инструменты для синхронизации и управления потоками. Опишите ситуации, в которых можно использовать каждый из инструментов.

19. Что такое потоки демоны. Когда используются и как создаются?

20. Что такое мультипроцессинг? Приведите примеры использования мультипроцессинга.

21. Какие три модели запуска процессов поддерживает Python? В чем разница между ними?

22. Назовите инструменты управления процессами. В каких ситуациях каждый инструмент необходимо использовать?

23. Как производить обмен данными между процессами? Перечислите преимущества и недостатки каждого инструмента.

24. Что такое асинхронность? Как работать с асинхронностью в Python.

25. Что такое асинхронность? Что такое событийный цикл?

26. Как работает асинхронность? Приведите примеры использования асинхронности вместо многопоточности и многопоточности.

27. Что такое корутина? Что такое объект-корутины и для чего используется?

28. Что такое API? Расскажите про REST API.

29. Что такое API? Опишите из чего состоит запрос и ответ от API.

30. Для чего используется библиотека SQLAlchemy? В чем разница между CORE и ORM уровнями?

31. Для чего нужны драйверы, при работе с БД через SQLAlchemy? Что такое строка подключения к БД и из чего она состоит?

32. Что такое Query Builder? В чем его разница от сырых SQL-запросов?

33. Что такое SQL-инъекция? Какой вид запросов используется для избежания ее допущения?

34. Что такое .env файл? Для чего он используется?

35. Что такое виртуальное окружение? Расскажите, как его активировать и как установить в него внешние библиотеки.