

Задание 01

1. Установите Node.js.
2. Проверьте работоспособность Node.js.

Задание 02

3. Разработайте серверное приложение 01-02, возвращающее ответ с разметкой **<h1>Hello World</h1>**.
4. Проверьте работоспособность приложения с помощью браузера.
5. Проверьте работоспособность приложения с помощью приложения **POSTMAN**.

Задание 03

6. Разработайте серверное приложение 01-03, на основе разработанных в задании 2, которое в ответе сервера пересылает html-страницу с содержимым запроса (метод, uri, ...).
7. Выполните GET-запрос к серверу с помощью браузера и убедитесь в работоспособности сервера.
8. Выполните POST-запрос (содержащий данные в теле сообщения) к серверу с помощью POSTMAN и убедитесь в работоспособности сервера.

Задание 04. Ответьте на следующие вопросы.

9. Дайте определение понятию «Интернет».
10. Дайте определение понятию «Служба Интернет».
11. Дайте определение понятию «Узел сети Интернет».
12. Дайте определение понятию «клиент-серверное приложение».
13. Дайте определение понятию «сетевой протокол».
14. Перечислите основные свойства протокола HTTP.
15. Перечислите состав информации, пересылаемой в HTTP-запросе.
16. Перечислите состав информации, пересылаемой в HTTP-ответе.

17. Дайте определение понятию «web-приложение».
18. Дайте определение понятиям «frontend» и «backend».
19. Дайте определение понятию «кроссплатформенное приложение».
20. Изобразите и поясните общую схему web-приложения.
21. Назовите основные технологии разработки серверных кроссплатформенных приложений.
22. Поясните понятие «асинхронная операция».
23. Поясните принцип выполнения асинхронного запроса с помощью объекта XMLHttpRequest и Fetch.
24. Поясните основное назначение сервера NODE.JS.
25. Перечислите основные свойства сервера NODE.JS.