Лабораторная работа 01

ПСКП

ПОИТ-3

**Задание 01**

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

**Задание 02**

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

**Задание 03**

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

**Задание 04.** Ответьте на следующие вопросы.

1. Дайте определение понятию «Интернет».
2. Дайте определение понятию «Служба Интернет».
3. Дайте определение понятию «Узел сети Интернет».
4. Дайте определение понятию «клиент-серверное приложение».
5. Дайте определение понятию «сетевой протокол».
6. Перечислите основные свойства протокола HTTP.
7. Перечислите состав информации, пересылаемой в HTTP-запросе.
8. Перечислите состав информации, пересылаемой в HTTP-ответе.
9. Дайте определение понятию «web-приложение».
10. Дайте определение понятиям «frontend» и «backend».
11. Дайте определение понятию «кроссплатформенное приложение».
12. Изобразите и поясните общую схему web-приложения.
13. Назовите основные технологии разработки серверных кроссплатформенных приложений.
14. Поясните понятие «асинхронная операция».
15. Поясните принцип выполнения асинхронного запроса с помощью объекта XMLHTTPRequest и Fetch.
16. Поясните основное назначение сервера NODE.JS.
17. Перечислите основные свойства сервера NODE.JS.
18. **Дайте определение понятию «Интернет».**

Всемирная система объединённых компьютерных сетей для хранения, обработки и передачи информации. Часто упоминается как Всемирная сеть и Глобальная сеть, а также просто Сеть.

1. **Дайте определение понятию «Служба Интернет».**

**Службы Интернета** — это системы, предоставляющие услуги пользователям Интернета. К ним относятся: электронная почта, WWW, телеконференции, списки рассылки, FTP, IRC, а также другие продукты, использующие Интернет как среду передачи информации:

1) сеть на основе TCP/IP;

2) стандарты Internet (RFC, STD);

3) службы Интернет (DNS, SMTP/POP3/IMAP, WWW, FTP, Telnet, SSH,…);

4) организации, управляющие сетью Internet (ISOC, IETF, ICANN, IANA, IAB, …).

1. **Дайте определение понятию «Узел сети Интернет».**

Узел сети Internet - это часть компьютерной сети или устройство, соединенное с другими участниками (узлами) данной сети. Это может быть как компьютер, так и специальный коммутатор, маршрутизатор или концентратор.

1. **Дайте определение понятию «клиент-серверное приложение».**

Клиент-серверное приложение = приложение (программа) с клиент-серверной архитектурой: приложение, состоящее из двух компонент – клиента и сервера; клиент и сервер взаимодействуют между собой в соответствии с заданными правилами (спецификациями, протоколами); для взаимодействия между клиентом и сервером в соответствии с правилами (спецификацией, протоколом) должно быть установлено соединение; инициатором соединения всегда является клиент.

1. **Дайте определение понятию «сетевой протокол».**

Клиент и сервер взаимодействуют между собой в соответствии с заданными правилами (спецификациями, **протоколами**)

1. **Перечислите основные свойства протокола HTTP.**

HTTP — это протокол, позволяющий получать различные ресурсы, например HTML-документы. Протокол HTTP лежит в основе обмена данными в Интернете. HTTP является протоколом клиент-серверного взаимодействия, что означает инициирование запросов к серверу самим получателем, обычно веб-браузером (web-browser). Полученный итоговый документ будет (может) состоять из различных поддокументов являющихся частью итогового документа: например, из отдельно полученного текста, описания структуры документа, изображений, видео-файлов, скриптов и многого другого.

1. **Перечислите состав информации, пересылаемой в HTTP-запросе.**

- метод;

- URI;

- версия протокола (HTTP/1.1);

- заголовки (пары: имя/заголовок);

- параметры (пары: имя/заголовок);

- расширение.

1. **Перечислите состав информации, пересылаемой в HTTP-ответе.**

- версия протокола (HTTP/1.1);

- код состояния (1xx, 2xx, 3xx, 4xx, 5xx);

- пояснение к коду состояния;

- заголовки (пары: имя/заголовок);

- расширение.

1. **Дайте определение понятию «web-приложение».**

Веб-приложение — клиент-серверное приложение, в котором клиент взаимодействует с веб-сервером при помощи браузера

1. **Дайте определение понятиям «frontend» и «backend».**

Фронтенд (англ. Front-end) — клиентская сторона пользовательского интерфейса к программно-аппаратной части сервиса.

Бекенд (англ. back-end) — программно-аппаратная часть сервиса.

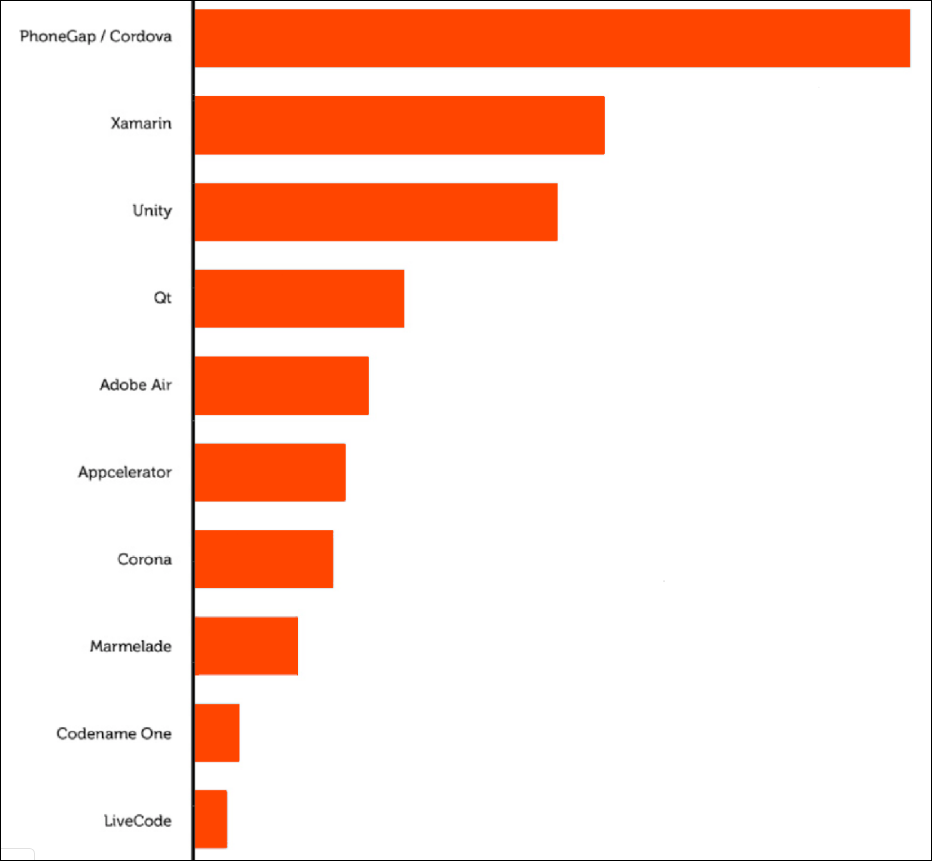
1. **Дайте определение понятию «кроссплатформенное приложение».**

Приложений, которые могут работать в различных операционных системах. После написания кода приложения его можно развернуть на разных устройствах и платформах, не беспокоясь о проблемах несовместимости. Это универсальный подход, который широко используется для экономии времени и денег на разработку. Часто для этого используются специализированные кроссплатформенные фреймворки.

1. **Изобразите и поясните общую схему web-приложения.**



1. **Назовите основные технологии разработки серверных кроссплатформенных приложений.**



1. **Поясните понятие «асинхронная операция».**

Асинхронность позволяет вынести отдельные задачи из основного потока в специальные асинхронные методы или блоки кода.

1. Поясните принцип выполнения асинхронного запроса с помощью объекта XMLHTTPRequest и Fetch.
2. **Поясните основное назначение сервера NODE.JS.**

**Node.js** — это опенсорсная кроссплатформенная среда выполнения для JavaScript, которая работает на серверах. Платформа Node.js построена на базе JavaScript движка V8 от Google, который используется в браузере Google Chrome. Данная платформа, в основном, используется для создания веб-серверов, однако сфера её применения этим не ограничивается.

1. **Перечислите основные свойства сервера NODE.JS.**

▍Скорость

Одной из основных привлекательных особенностей Node.js является скорость. JavaScript-код, выполняемый в среде Node.js, может быть в два раза быстрее, чем код, написанный на компилируемых языках, вроде C или Java, и на порядки быстрее интерпретируемых языков наподобие Python или Ruby. Причиной подобного является неблокирующая архитектура платформы, а конкретные результаты зависят от используемых тестов производительности, но, в целом, Node.js — это очень быстрая платформа.

▍Простота

Платформа Node.js проста в освоении и использовании. На самом деле, она прямо-таки очень проста, особенно это заметно в сравнении с некоторыми другими серверными платформами.

▍JavaScript

В среде Node.js выполняется код, написанный на JavaScript. Это означает, что миллионы фронтенд-разработчиков, которые уже пользуются JavaScript в браузере, могут писать и серверный, и клиентский код на одном и том же языке программирования без необходимости изучать совершенно новый инструмент для перехода к серверной разработке.

В браузере и на сервере используются одинаковые концепции языка. Кроме того, в Node.js можно оперативно переходить на использование новых стандартов ECMAScript по мере их реализации на платформе. Для этого не нужно ждать до тех пор, пока пользователи обновят браузеры, так как Node.js — это серверная среда, которую полностью контролирует разработчик. В результате новые возможности языка оказываются доступными при установке поддерживающей их версии Node.js.

▍Движок V8

В основе Node.js, помимо других решений, лежит опенсорсный JavaScript-движок V8 от Google, применяемый в браузере Google Chrome и в других браузерах. Это означает, что Node.js пользуется наработками тысяч инженеров, которые сделали среду выполнения JavaScript Chrome невероятно быстрой и продолжают работать в направлении совершенствования V8.

▍Асинхронность

В традиционных языках программирования (C, Java, Python, PHP) все инструкции, по умолчанию, являются блокирующими, если только разработчик явным образом не позаботится об асинхронном выполнении кода. В результате если, например, в такой среде, произвести сетевой запрос для загрузки некоего JSON-кода, выполнение потока, из которого сделан запрос, будет приостановлено до тех пор, пока не завершится получение и обработка ответа.

▍Библиотеки

Благодаря простоте и удобству работы с менеджером пакетов для Node.js, который называется npm, экосистема Node.js прямо-таки процветает. Сейчас в реестре npm имеется более полумиллиона опенсорсных пакетов, которые может свободно использовать любой Node.js-разработчик.