1. Дайте определение понятию «Интернет». (не лк)

информационно-коммуникационная сеть и всемирная система объединённых компьютерных сетей для хранения и передачи информации.

1. Дайте определение понятию «Служба Интернет».(не лк)

это системы, предоставляющие услуги пользователям Интернета (DNS, SMTP/POP3/IMAP, WWW, FTP, Telnet, SSH)

1. Дайте определение понятию «Узел сети Интернет».

устройство, имеющее IP-адрес и подключенное сети Интернет (обычно к сети Интернет-провайдера). Каждый узел характеризуется своей программно-аппаратной платформой – аппаратурой и операционной системой.

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

приложение (программа) с клиент-серверной архитектурой:

• состоит из двух компонент – клиента и сервера;

• клиент и сервер взаимодействуют между собой в соответствии с заданными правилами (протоколами);

• для взаимодействия между клиентом и сервером в соответствии с правилами (протоколом) должно быть установлено соединение;

• инициатором соединения – клиент

1. Дайте определение понятию «сетевой протокол». (не лк)

набор определённых правил или соглашений интерфейса логического уровня, который определяет обмен данными между различными программами.

1. Перечислите основные свойства протокола HTTP.

* версии HTTP/1.1 – действующий (текстовый), HTTP/2 – действующий (бинарный), HTTP/3 (раньше HTTP-over-QUIC, стадия черновика);

• два типа абонентов: клиент и сервер;

• два типа сообщений: request и response;

• от клиента к серверу – request;

• от сервера к клиенту – response;

• на один request всегда один response, иначе ошибка;

• одному response всегда один request, иначе ошибка;

• stateless; • TCP-порты: 80, 443;

• для адресации используется URI или URL;

• поддерживается W3C, описан в нескольких RFC.

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

• метод; • URI; • версия протокола (HTTP/1.1); • заголовки (пары: имя/значение); • тело

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); • тело.

1. Перечислите группы заголовков HTTP.

General, Request, Response, Entity, Пользовательские?

1. Дайте определение понятию «web-приложение».

клиент-серверное приложение, у которого клиент и сервер взаимодействуют по протоколу HTTP. Когда говорят о разработке webприложения, говорят о разработке frontend (клиента) и backend (сервера) Курс посвящен разработке серверной части web-приложения или иначе разработке web-сервера (backend).

1. Дайте определение понятиям «frontend» и «backend». (не лк)

frontend - презентационная часть информационной или программной системы, её пользовательский интерфейс и связанные с ним компоненты, (клиент).

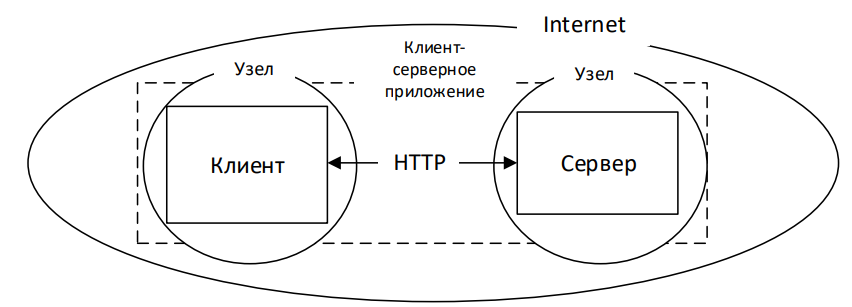
backend - это разработка бизнес-логики продукта (сайта или веб-приложения), (сервер).

1. Дайте определение понятию «кроссплатформенное приложение».

приложение, способное работать на более чем одной программно-аппаратной (аппаратура + операционная система) платформе.

1)на уровне компилятора (С, С++); 2) на уровне среды (или фреймворка) исполнения (Java/JVM, C#/.NET CORE/CLR, JS/Node, …)

1. Изобразите и поясните общую схему web-приложения.



1. Назовите основные технологии разработки серверных кроссплатформенных приложений.

• PHP / Apache, LAMP;

• Java / JVM / Application Server;

• C# / ASP.NET CORE;

• Python / Django;

• Ruby on Rails;

• JS / Node.js, …

1. Поясните понятие «асинхронная операция».

1) заявка на исполнение;

2) получение результата; при этом участвуют два механизма: A-механизм, формирующий заявку и потом получающий результат; B-механизм, получающий заявку от A, исполняющий операцию и отправляющий результат A; продолжительность исполнения операции B-механизмом, как правило, непредсказуемо; в то время пока Bмеханизм исполняет операцию, Амеханизм выполняет собственную работу.

1. Поясните основное назначение NODE.JS.

программная платформа для разработки серверных web-приложений на языке JS/V8.

1. Перечислите основные свойства NODE.JS.

* основан на **Chrome V8**;
* **среда (контейнер) исполнения** приложений на JavaScript;
* поддерживает механизм **асинхронности**;
* ориентирован на **события**;
* **однопоточный** (код приложения исполняется только в одном потоке, один стек вызовов); обычно в серверах для каждого соединения создается свой поток, в Node.js все соединения обрабатываются в одном JS-потоке;
* **не блокирует** выполнение кода при вводе/выводе (в файловой системе до 4х одновременно);
* в состав Node.js входят инструменты: **npm** – пакетный менеджер; **gyp** - Python-генератор проектов; **gtest** – Google фреймворк для тестирования С++ приложений;
* использует библиотеки: **V8** – библиотека V8 Engine, **libuv** – библиотека для абстрагирования неблокирующих операций ввода/вывода (представляет собой обертку над epoll, kqueue, IOCP); **llhttp** – легковесный парсер http-сообщений (написан на C и не выполняет никаких системных вызовов); **c-ares** -библиотека для работы с DNS; **OpenSSL** – библиотека для криптографии; **zlib** – сжатие и распаковка.