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».

\* Сеть на основе TCP/IP  
\* стандарты Internet (RFC, STD) – там прописаны протоколы  
\* службы Интернет (DNS, FTP, Telnet…)  
\* организации, упр-щие этой сетью (ISOC, IETF, IANA, IAB…)

1. Дайте определение понятию «Служба Интернет».

Служба Интернет = протокол + сервер  
(DNS, Telnet, FTP, POP3…)  
виды услуг, оказ. серверами

1. Дайте определение понятию «Узел сети Интернет».

У-во, имеющее IP и подключение сети Интернет (обычно к сети Интернет-провайдера). Каждый узел характ. Своей аппаратурой и ОС.

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

Приложение с клиент-серверной архитектурой  
\* клиент и сервер взаимод. между собой в соотв. с заданными правилами (**протоколами**)  
\* дб установлено **соединение**  
\* инициатор соединения – всегда **клиент**

1. Дайте определение понятию «сетевой протокол».

набор правил и д-й, кот. позвол. осущ. соед и обмен д-ми между 2+ вкл в сеть у-вами

1. Перечислите основные свойства протокола HTTP.

\* не завис от соед  
\* взаимод через соед  
\* не привязан к ТД

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

\* строка запроса  
\* заголовок

\* пустая строка  
\* тело сообщения

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

\* строка статуса  
\* заголовок  
\* пустая строка  
\* тело сообщения

1. Дайте определение понятию «web-приложение».

клиент-серверное приложение, у к-го клиент и сервер взаимод. по протоколу HTTP

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

\* frontend – клиентская часть web-прилож  
\* backend – серверная часть

1. Дайте определение понятию «кроссплатформенное приложение».

приложение, способ. работать на 2+ программно-аппаратной платформе (аппаратура + ОС)  
Кроссплатформенность мб достигнута:  
\* на уровне компилятора (С++, С)- перекомплир. в др.пл.  
\* на уровне среды исполнения (Java/JVM, C#/.NET CORE, JS/Node.js/V8) – не надо

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».

ее выполнение осущ. в 2 фазы:  
\* заявка на исполнение  
\* получение рез-та  
участвуют 2 механизма:  
\* А-мех. формир. заявку и потом получ. рез-т  
\* В-мех. получ от А заявку, исполн операцию и отпр рез А

Пока В работает, А тоже работает

1. Поясните принцип выполнения асинхронного запроса с помощью объекта XMLHTTPRequest и Fetch.

fetch добавл д-е через ф-цию  
XMLHTTPRequest м. без перезагрузки доб.д-е

1. Поясните основное назначение сервера NODE.JS.

создание масштабируемых сетевых приложений

1. Перечислите основные свойства сервера NODE.JS.

\* осн на Chrome V8  
\* среда исп на JS  
\* ориент на события  
\* однопоточный  
\* не блокир код при ввод/выводе  
\* исп библиотеки