Отлично! Вот ключевые вопросы по теме **Node.js: Основы (создание сервера, обработка HTTP-запросов)** с ответами и примерами.

**Вопросы по Node.js (создание сервера и HTTP-запросы)**

**1. Что такое Node.js и для чего он используется?**

**2. Как создать простой HTTP-сервер на Node.js?**

**3. Как обрабатывать разные маршруты (URL) в Node.js без фреймворков?**

**4. Как получать данные из GET-запросов?**

**5. Как обрабатывать POST-запросы и получать тело запроса?**

**6. Как запустить сервер и проверить его работу?**

**Ответы с примерами**

**1. Что такое Node.js и для чего он используется?**

**Ответ:**  
Node.js — это среда выполнения JavaScript на сервере, позволяющая создавать серверные приложения и обрабатывать HTTP-запросы.

**2. Как создать простой HTTP-сервер на Node.js?**

**Пример:**

const http = require('http');

const server = http.createServer((req, res) => {

res.writeHead(200, {'Content-Type': 'text/plain'});

res.end('Привет, Node.js сервер!');

});

server.listen(3000, () => {

console.log('Сервер запущен на http://localhost:3000');

});

**3. Как обрабатывать разные маршруты (URL) в Node.js без фреймворков?**

**Пример:**

const http = require('http');

const server = http.createServer((req, res) => {

if (req.url === '/') {

res.end('Главная страница');

} else if (req.url === '/about') {

res.end('О нас');

} else {

res.statusCode = 404;

res.end('Страница не найдена');

}

});

server.listen(3000);

**4. Как получать данные из GET-запросов?**

**Пример:**

const http = require('http');

const url = require('url');

const server = http.createServer((req, res) => {

const parsedUrl = url.parse(req.url, true);

const name = parsedUrl.query.name || 'гость';

res.end(`Привет, ${name}!`);

});

server.listen(3000);

**5. Как обрабатывать POST-запросы и получать тело запроса?**

**Пример:**

const http = require('http');

const server = http.createServer((req, res) => {

if (req.method === 'POST') {

let body = '';

req.on('data', chunk => {

body += chunk.toString();

});

req.on('end', () => {

res.end(`Получены данные: ${body}`);

});

} else {

res.end('Отправьте POST-запрос');

}

});

server.listen(3000);

**6. Как запустить сервер и проверить его работу?**

**Ответ:**

* Сохрани код сервера в файл, например, server.js.
* Выполни команду в терминале:

node server.js

* Открой браузер и перейди на http://localhost:3000.
* Для POST-запросов можно использовать инструменты, например, Postman или curl.

Если хочешь, могу помочь с созданием более сложного сервера или добавить примеры с использованием фреймворков (Express.js).  
Как считаешь?