**«УТВЕРЖДАЮ»**

**Зав. кафедрой ИСиТ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Смелов**

**Экзаменационные вопросы дисциплины**

**«Программирование серверных кроссплатформенных приложений»**

**для студентов 3-го курса специальности ПОИТ**

1. Протокол HTTP, основные свойства HTTP, структура запроса и ответа. Протокол HTTPS. Понятие web-приложения, структура и принципы работы web-приложения. Понятие асинхронности.
2. HTTP-аутентификация.
3. Протокол WebSockets, основные свойства, процедура установки соединения. WebSockets API.
4. Разработка простейшего HTTP-сервера в Node.js. Извлечение данных из HTTP-запроса, формирование данных HTTP-ответа. Пример. Тестирование с помощью браузера AJAX (XMLHTTPRequest/Fetch).
5. Разработка HTTP-сервера в Node.js. Обработка GET, POST, PUT и DELETE-запросов. Генерация ответа с кодом 404. Пример. Тестирование с помощью POSTMAN.
6. Разработка HTTP-сервера в Node.js. Обработка запросов к статическим ресурсам: html, css, js, png, msword. Пример. Тестирование с помощью браузера.
7. Разработка HTTP-сервера в Node.js. Обработка query-параметров GET-запроса. Пример. Тестирование с помощью браузера.
8. Разработка HTTP-сервера в Node.js. Обработка uri-параметров GET-запроса. Пример. Тестирование с помощью браузера.
9. Разработка HTTP-сервера в Node.js. Обработка параметров POST-запроса. Пример. Тестирование с помощью браузера (<form>) и POSTMAN.
10. Разработка HTTP-сервера в Node.js. Обработка json-сообщения в POST-запросе. Пример. Тестирование с помощью POSTMAN.
11. Разработка HTTP-сервера в Node.js. Пересылка файла в POST-запросе (upload). Пример. Тестирование с помощью браузера.
12. Разработка HTTP-сервера в Node.js. Пересылка файла в ответе (download). Пример. Тестирование с помощью браузера.
13. Разработка HTTP-клиента в Node.js. Оправка GET запроса с query-параметрами. Пример. Тестирование с помощью с Node.js-сервера.
14. Разработка HTTP-клиента в Node.js. Оправка POST-запроса с параметрами в теле. Пример. Тестирование с помощью с Node.js-сервера.
15. Разработка HTTP-клиента в Node.js. Оправка POST-запроса с json-сообщением. Пример. Тестирование с помощью с Node.js-сервера.
16. Разработка HTTP-клиента в Node.js. Обработка json-ответа. Пример. Тестирование с помощью с Node.js-сервера.
17. Разработка HTTP-клиента в Node.js. Пересылка файла на сервер в POST-запросе (upload). Пример. Тестирование с помощью с Node.js-сервера.
18. Разработка HTTP-клиента в Node.js. Обработка ответа с файлом (download). Пример. Тестирование с помощью с Node.js-сервера.
19. Разработка Websockets-приложения: Node.js-сервер, браузер-клиент. Пример.
20. Разработка Websockets-приложения: обработка json-сообщений, Node.js-сервер, Node.js-клиент. Пример.
21. Разработка RPC-Websockets-сервера. Пример. Тестирование: Node.js-клиент.
22. Применение функции pipe для обработки данных (файла) файловой системы и записи в http-ответ. Пример.
23. Разработка приложения, выполняющего запрос к SQL-базе данных: выполнение динамического SELECT-запроса.

1. Разработка приложения, выполняющего запрос к SQL-базе данных: выполнение динамического INSERT-запроса. Пример.
2. Разработка приложения, выполняющего запрос к SQL-базе данных: выполнение динамического UPDATE-запроса. Пример.
3. Разработка приложения, выполняющего запрос к SQL-базе данных: выполнение динамического DELETE-запроса. Пример.
4. Разработка приложения, выполняющего запрос к SQL-базе данных: вызов удаленной процедуры. Пример.
5. Применение СУБД Redis. Основные принципы работы. Пример.
6. Применение пакета Sequelize. Основные принципы работы. Пример.
7. Пакет Express. Основные принципы работы. Middleware-код. Пример.
8. Пакет Express. Основные принципы работы. Маршрутизация. Пример.
9. Пакет Express. Основные принципы работы. Статические файлы. Пример.
10. Пакет Express. Основные принципы работы. Обработка Cookie. Пример.
11. Пакет Express. Основные принципы работы. Шаблонизация Handlebars. Пример
12. Пакет Express. Основные принципы работы. Шаблонизация Handlebars. Пример.
13. Пакет Express. Обработка

Доцент каф. ИСиТ В.В. Смелов

**В билете 3 вопроса: 1 и 2 вопросы из списка (1-57), 3-й вопрос – демонстрация одной из лабораторных работ (1-16).**

**На экзамене студент обязан предоставить все выполненные (1-16) лабораторные работы. Студент, который не предоставит полный список выполненных лабораторных работ автоматически получает неудовлетворительную оценку.**