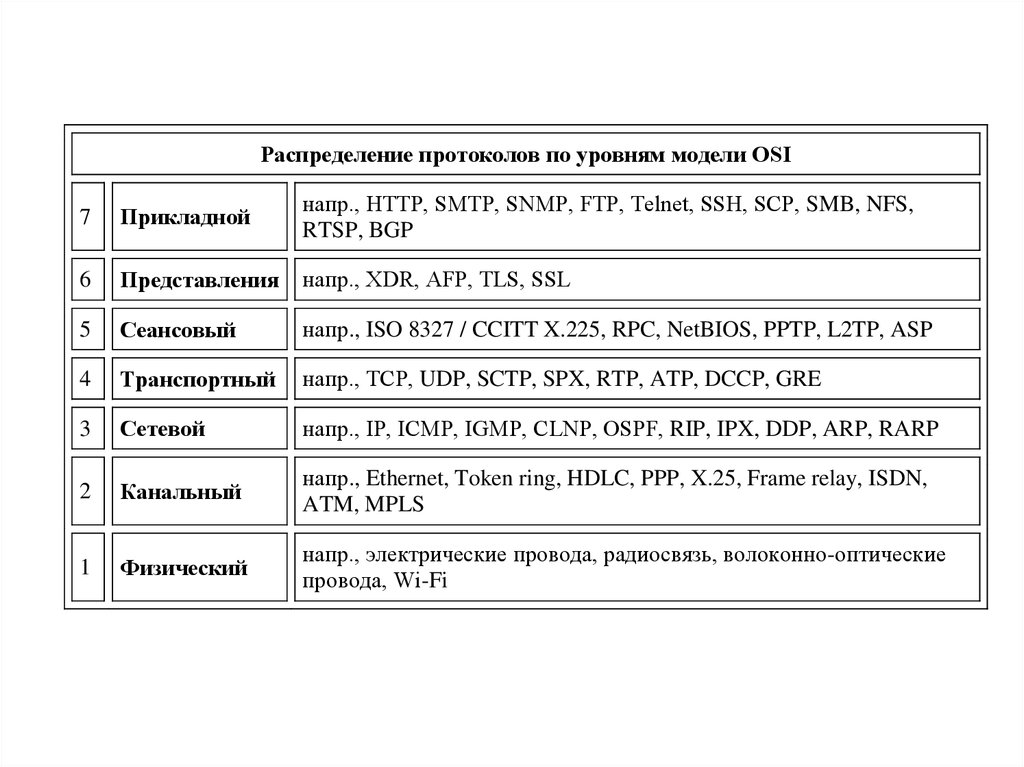
# Ответьте на следующие вопросы



1. **Поясните понятие «TCP-порт».**

Число от 1 до 65535, указывает, какой программе предназначается пакет. Каждый открытый порт – это установленное соединение между двумя точками – соответствующей программой и узлом соединения.

Число для идентификации конкретного процесса или службы.

1. **Поясните понятие «сетевой сокет».**

Точка для отправки и приема данных в сети; часть программного интерфейса для обеспечения передачи данных (между узлами сети или процессами). А вообще, по Шиману, сокет – это IP + port.

1. **Поясните понятие «WebSocket».**

Протокол передачи данных, основанный на TCP, для установки длительного соединения и обмена сообщениями между клиентом и сервером в режиме *дуплексной* связи.

1. **Поясните процедуру установки соединения между WS-сервером и WS-клиентом.**

Соединение устанавливается следующим образом:

1. Клиент посылает обычный HTTP-запрос, называемый рукопожатием, с заголовками Upgrade и Connection (Upgrade: websocket)
2. Сервер решает, устанавливать WebSocket-соединение или нет, и отправит заголовок Upgrade (Upgrade: WebSocket)
3. Устанавливается соединение и HTTP заменяется на TCP/IP-соединение, далее можно отправлять данные
4. **Поясните понятие «широковещательное сообщение».**

Сообщение, посылаемое всем узлам сети.

1. **Поясните принцип организации потокового ввода/вывода через WS-соединение.**

Процесс включает установку соединения, отправку/прием данных, фрагментацию сообщений, закрытие соединения и обработку ошибок.

1. Создать поток WebSocketStream
2. Создать поток чтения или записи из модуля fs
3. source.pipe(destination)
4. **Поясните принцип действия и назначение механизма «ping/pong».**

Особо легкие служебные пакеты. Отправляются для того, чтобы проверить, жив вообще наш собеседник или нет. При получении ping необходимо как можно скорее отправить в ответ pong.

1. **Поясните аббревиатуру «RPC».**

Remote Procedure Call – удаленный вызов процедуры – вызов некоторой функции из другого места (другое адресное пространство, другой узел сети), причем это выглядит, как локальный вызов.

1. **Поясните принцип работы RPC-механизма, предоставляемого пакетом rpc-websockets.**
2. server.register(‘rpc-method’, <callback>);
3. если protected, то server.setAuth(login, password)
4. client.call(‘rpc-method’, params).then(res => sout(res));
5. если protected, то client.login().then(call());
6. **Установка соединения:**
   1. Клиент устанавливает веб-сокет-соединение с сервером.
7. **Регистрация удаленных процедур:**
   1. Сервер регистрирует функции для удаленного вызова.
8. **Вызов удаленных процедур:**
   1. Клиент отправляет запросы на вызов удаленных процедур через веб-сокет.
9. **Обработка запросов на сервере:**
   1. Сервер принимает запросы и вызывает соответствующие удаленные процедуры.
10. **Выполнение удаленных процедур:**
    1. Удаленные процедуры выполняются на сервере.
11. **Отправка результатов клиенту:**
    1. Результаты выполнения отправляются обратно клиенту через веб-сокет.
12. **Обработка ответов на клиенте:**
    1. Клиент обрабатывает результаты выполнения удаленных процедур.

Начало формы

1. **Поясните принцип действия и назначение механизма «subscriber/publisher».**

Publisher генерирует события – server.emit(‘event’)

Subscriber подписывается – client.subscribe(‘event’)

1. **Поясните принцип действия и назначение механизма уведомлений.**

Клиент может отправлять серверу уведомления: client.notify(‘A’)

Сервер их обрабатывает: server.register(‘A’, <callback>)