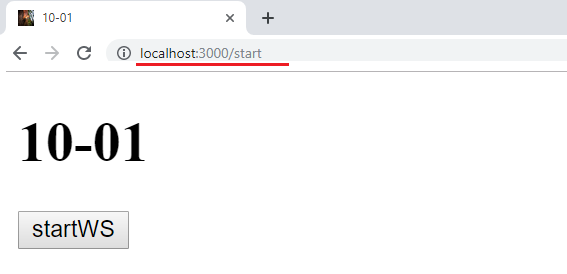
Лабораторная работа 10

ПСКП

ПОИТ-3

**Задание 01**

1. Разработай приложение **10-01**, представляющее собой HTTP-сервер и WebSocket(WS)-север одновременно.
2. HTTP-сервер должен прослушивать порт **3000**.
3. WS-север должен прослушивать порт **4000**.
4. HTTP-сервер реагирует только на GET-запрос **/start**, на остальные запросы HTTP-сервер должен возвращать ответ со статусом **400**.
5. На GET-запрос **/start** HTTP-сервер возвращает HTML-страницу следующего вида.



1. При нажатии кнопки **startWS**, должна запускаться JS-функция, периодически каждые 3 сек. оправляющая сообщения WS-серверу следующего вида.

**10-01-client: *n***, где ***n*** – последовательный номер отправляемого серверу сообщения.

1. JS-функция должна отображать сообщения принятые от сервера.
2. JS-функция должна автоматически остановить передачу сообщений через 25 сек и закрыть WS-соединение.
3. WS-сервер принимает сообщения от клиента.
4. WS-сервер отображает принятые сообщения от клиента.
5. WS-сервер периодически каждые 5 сек. Оправляет клиенту сообщения следующего вида.

**10-01-server: *n->k***, где ***n*** – номер из последнего сообщения клиента, а ***k*** - последовательный номер отправляемого клиенту сообщения.

1. Проверьте работоспособность приложения. Объясните эффект, наблюдаемый при многократном нажатии на кнопку **startWS.**

**Задание 02**

1. Разработай приложение **10-02**, представляющее собой WebSocket(WS)-клиент.
2. Приложение **10-02** должно быть функционально аналогично JS-функции, разработанной в первом задании.
3. Проверьте работоспособность приложения совместно с сервером, разработанном в первом задании.
4. Запустите несколько экземпляров приложения **10-02** для взаимодействия с одним общим сервером. Объясните наблюдаемый эффект.

**Задание 03**

1. Разработай приложение **10-03**, представляющее собой широковещательный WebSocket-сервер.
2. Разработайте приложение **10-03a** - WebSocket-клиент.
3. Продемонстрируйте работоспособность широковещательного WebSocket-сервера **10-03,** взаимодействующего с тремя клиентами **10-03a.**

**Задание 04** Ответьте на следующие вопросы

1. Поясните разницу между полудуплексном и дуплексным каналами данными.

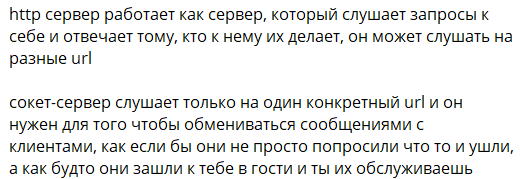
При *полудуплексной* передаче сигналы могут передаваться в обоих направлениях, но в каждый момент времени – только в одном. В качестве примера можно привести модем – в каждый момент времени он может либо только принимать, либо только передавать информацию.

При *дуплексной* передаче сигналы передаются в обоих направлениях одновременно. Примером устройства с дуплексным режимом передачи является обычный телефон.

1. Какой тип канала применяется HTTP-протоколом.

дуплексный

1. Поясните понятие **WebSocket**

**WebSocket** — это самое кардинальное расширение протокола HTTP с его появления. Это не финтифлюшки, это **сдвиг  парадигмы HTTP**. Изначально синхронный протокол, построенный по модели «запрос — ответ», становится **полностью асинхронным и симметричным**. Теперь уже нет клиента и сервера с фиксированными ролями, а есть два равноправных участника обмена данными. Каждый работает сам по себе, и когда надо отправляет данные другому. Отправил — и пошел дальше, ничего ждать не надо. Вторая сторона ответит, когда захочет — может не сразу, а может и вообще не ответит. Протокол дает полную свободу в обмене данными, вам решать как это использовать. 

1. Какой тип канала применяется WebSocket-протоколом.

полудуплексный

1. Поясните понятия «широковещательное сообщение» (broadcast), «широковещательный сервер».

**Широковещательные сообщения** -тип **сообщений**, не имеющих адресата. Рассылаются на все устройства, находящиеся в сети.

