Поиск

Карта квестов Лекции CS50 Android

Socket

Java Collections 10 уровень, 7 лекция

ОТКРЫТА

— Привет, Амиго!

- Здорово, Риша!
- Тема сегодняшней лекции Сокеты. Socket в переводе с английского Розетка.

Ты уже знаешь, что у каждого компьютера в сети есть его уникальный ІР-адрес.

- Ага.
- И вот представь, что у тебя есть несколько компьютеров и на каждом компьютере запущено с десяток программ, которые работают с интернетом: Skype, ICQ, и т.д.

И эти программы хотят общаться между собой.

Надо сделать так, чтобы они друг другу «не мешали». Чтобы Skype связывался со Skype'ом, ICQ с ICQ и т.д.

Помнишь, как это проблему решили с URL и веб-серверами?

- Ага, добавили порты.
- Именно.

Это все равно, что в доме сделать много маленьких комнат и объявить его многоквартирным. Каждый порт – это как отдельная квартира.

Так вот IP-адрес – это уникальный номер компьютера, а IP-адрес+порт – это уникальный номер некой «квартиры» (ячейки) в компьютере, которую может занять программа.

Вот такая уникальная ячейка и называется сокет.

У сокета есть его уникальный номер, и состоит он из IP-адреса и номера порта.

- Ага. Т.е. сокет это номер некой виртуальной ячейки в компьютере, которую может занимать программа? А другая программа будет слать ей в эту ячейку сообщения, и таким образом они будут общаться?
- Не знаю, как ты это понял, но именно так и есть.
- Это моя робоинтуиция подсказала.
- Отлично. Тогда давай немного подробностей.

Сокеты – это фактически самый базовый и самый примитивный способ межсетевого взаимодействия программ.

В Java для работы с сокетами есть два класса. Это классы **Socket** и **ServerSocket**.

ServerSocket – это специальный класс, объекты которого выполняют роль сервера – т.е. могу обслуживать запросы, пришедшие на определенный сокет.

Класс **Socket** – это фактически Socket-клиент, с помощью него мы можем послать сообщение некоторому сокету и получить ответ.

Как послать сообщение какому-то сокету:

```
Пример
      //создаем сокет
 1
 2
      Socket clientSocket = new Socket("localhost", 4444);
 3
 4
      //получаем OutputStream
      OutputStream outputStream = clientSocket.getOutputStream();
 5
      PrintWriter out = new PrintWriter(outputStream, true);
 6
 7
      out.println("Kiss my shiny metal ass!");
      out.flush();
 8
 9
      //читаем ответ
10
      InputStream inputStream = clientSocket.getInputStream();
11
12
      BufferedReader in = new BufferedReader(new InputStreamReader(inputStream));
13
      String answer = in.readLine();
```

- Все очень напоминает скачивание файла из интернета.
- Это потому, детка, что там тоже используются сокеты.

Сокеты используются в основе всего связанного с сетью, ну или почти всего.

Вот тут можешь почитать дополнительную информацию.

- Спасибо за лекцию, Риша.
- Это еще не все. Размечтался.

Теперь мы разберемся, как работает серверный сокет.

Он работает чуток посложнее.

```
Пример
```

```
1
     //создаем объект сервер-сокет
     ServerSocket serverSocket = new ServerSocket(4444); //порт
2
3
4
     //в цикле обрабатываем входящие соединения.
     while (true)
5
6
     {
7
      //метод accept ждет, пока кто-то не подключится.
8
      Socket socket = serverSocket.accept();
9
10
      //читаем сообщение
      InputStream inputStream = socket.getInputStream();
11
12
      BufferedReader in = new BufferedReader(new InputStreamReader(inputStream));
      String message = in.readLine();
13
14
      //придумываем ответ - просто разворачиваем строку задом наперед
15
      String reverseMessage = new StringBuilder(message).reverse().toString();
16
17
      //отправляем ответ
18
19
      OutputStream outputStream = socket.getOutputStream();
      PrintWriter out = new PrintWriter(outputStream, true);
20
      out.println(reverseMessage);
21
      out.flush();
22
23
     }
```

Хочу обратить твое внимание на несколько моментов.

Момент 1: Для создания сокета нужно указывать IP-адрес (или домен) и порт. Для создания серверного сокета – только порт. Серверный сокет появляется только на том компьютере, где его создали.

Момент 2: У класса ServerSocket есть метод ассерt(), который, если его вызвать, будет ждать входящее соединение. Т.е. метод будет выполняться бесконечно долго, пока какой-то клиентский сокет не попробует обратиться к нему. Тогда метод ассерt() примет соединение, создаст объект сокет для коммуникации и после этого вернет этот объект.

С точки зрения Java-программиста, сокет – это два потока – InputStream из которого можно читать сообщения/данные и OutputStream, куда можно писать сообщения/данные.

Когда ты создаешь серверный сокет, фактически появляется порт, к которому могут подключаться сокеты с других компьютеров. Но для этого им надо правильно указать номер порта нашего сокета и IP-адрес нашего компьютера. Ну, или его доменное имя.

Вот тебе интересный пример, можешь попробовать разобрать его и выполнить:

http://habrahabr.ru/post/69136/

Там весь смысл в том, что с помощью серверного сокета пишется примитивнейший веб-сервер, к которому можно просто обращаться из браузера.

— Ого! Веб-сервер? Круто! Я изучу его очень внимательно.

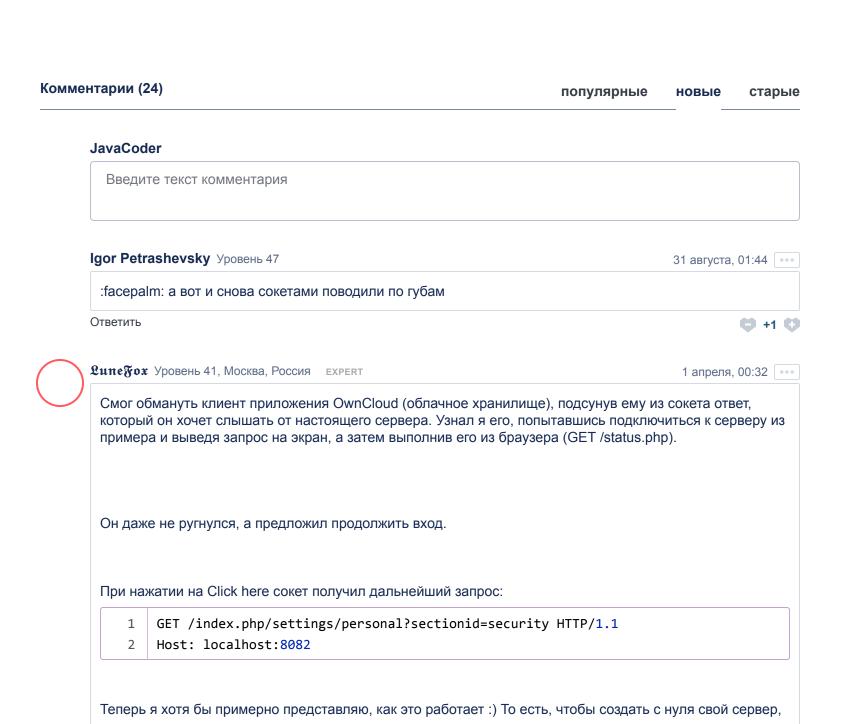
Спасибо, Риша.

— Все, Амиго, иди отдыхать!

< Предыдущая лекция



+24

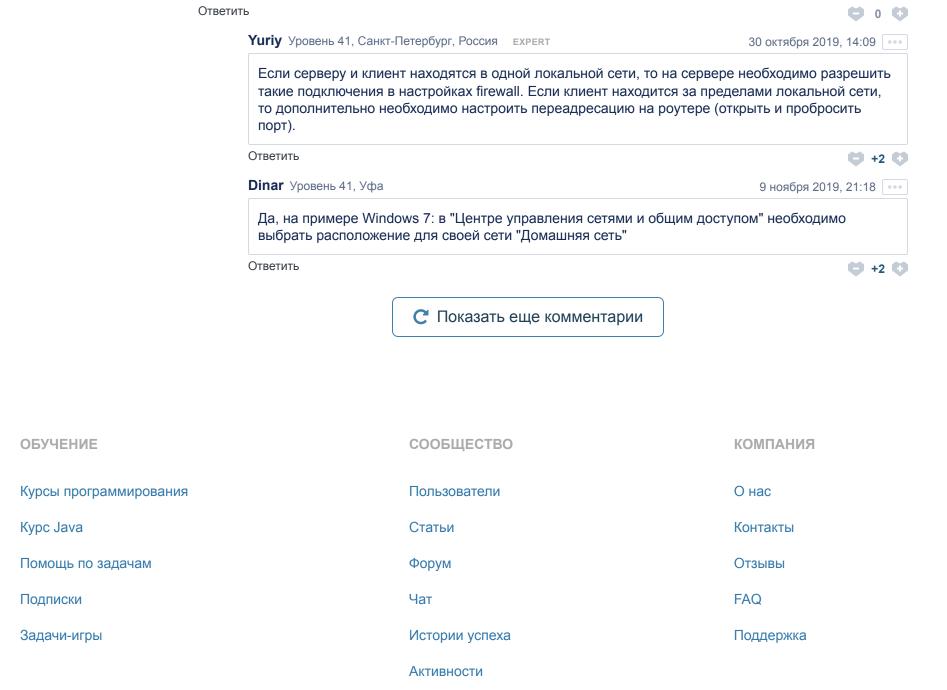


отклонять. Ответить **O** 0 **O** Igor Petrashevsky Уровень 47 31 августа, 01:47 http, ftp, snmp, pop3, smtp - это все надстройки над древнейшим протоколом telnet, который появился до tcp/ip и умел только в текст. Сделаете telnet - сделаете все остальное, путем добавки/смены костылей. Это полезно в учебных целях, т.к. для практики есть готовые фрэймворки со всеми необходимыми велосипедами. Ответить 0 0 Fomka_Wyverno Уровень 41, Киев, Украина 27 ноября 2021, 22:14 Отсылка на Бэндера? out.println("Kiss my shiny metal ass!"); Ответить 0 Ars Уровень 41 5 декабря 2021, 13:19 Тут весь курс - отсылка на Футураму. А точнее раньше все персонажи были оттуда. Но потом переделали. Ответить +3 Valua Sinicyn Уровень 41, Харьков, Украина 4 марта 2021, 13:32 ••• Сокеты на человеческом. Ответить **+4** Женя Уровень 41, Минск 23 февраля 2021, 10:59 "— Не знаю, как ты это понял, но именно так и есть". Наверно, если бы у валидатора было сознание, он бы то же самое после успешной валидации сообщал. Ответить +13 Alexey Prilessky Уровень 40, Минск 15 декабря 2020, 19:44 Интересно про это читать, когда уже в Multithreading Написали свой онлайн чат) +8 👣 🗟 Ответить Roman Backend Developer B Yandex 22 февраля 2021, 13:12 ••• То чувство, когда сначала пошел в коллекции Ответить 0 0 Konstantin Уровень 41, Одесса 28 сентября 2020, 10:11 "http://habrahabr.ru/post/69136/" - шёл 2020 год Ответить **+1 (7)** Владислав Backend Developer 5 августа 2020, 21:21 По крупицам, со временем, то, что было не понятно еще уровней 5 или 8 назад, то, что казалось абсолютной дичью, начинает выстраиваться в целостную картину. И так с любой максимально не понятной по-началу темой. Спасибо создателям за такие уроки. Ответить +6 Дворник Уровень 41, Санкт-Петербург, Россия 9 апреля 2020, 12:05 http://www.quizful.net/post/java-socket-programming помогло. А статья на хабре ничего не поясняет, что и почему, и как должно быть. Ответить +6 Anonymous #2413783 Java Developer 24 декабря 2020, 18:18 Спасибо за ссылку, осознанно переписал, почитал комменты, запустил у себя на компе. Только после этого пришло понимание Ответить **C** +1 **C** Svetlana Vydrina Уровень 41 26 июня, 18:16 Ссылка не работает (Ответить 0 0 Даниил Salesforce Developer в Viseven маster 29 октября 2019, 23:25 По примеру с хабра запустил свой сервер, но так и не понял как сделать так что бы с других IP адресов

получать к нему доступ. Пробовал узнать свой IP и с других устройств (на на своём Wi-Fi) заходить на

этот сервер - ничего не получилось. Может кто знает как это сделать?

нужно читать заголовки запросов и реагировать на них определённым образом, а неизвестные запросы



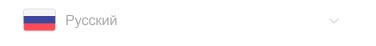


RUSH

JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в один клик, необходимый минимум теории по основам Java и мотивирующие фишки, которые помогут пройти курс до конца: игры, опросы, интересные проекты и статьи об эффективном обучении и карьере Java-девелопера.

ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА



СКАЧИВАЙТЕ НАШИ ПРИЛОЖЕНИЯ







"Программистами не рождаются" © 2022 JavaRush