Socket Programming in Java

Savong Bou
Center for Computational Sciences
University of Tsukuba

概要

- **別のマシン場合**：接続依頼にServerのIPが必要
- **同じマシン場合**：localhostなので、ServerのIPが不要

Client側のプログラミング

```
Socket socket = new Socket("127.0.0.1", 5000)
```

ServerのIP
か
同じマシン場合「http://localhost/」

TCPポート

コミュニケーション

```
//端末から入力を受け取ります
input = new DataInputStream(System.in);
```

```
//出力をソケットに送信します
out = new DataOutputStream(
            socket.getOutputStream());
```

接続を閉じ

```
// close the connection
    try {
        input.close();
        out.close();
        socket.close();
    }
    catch (IOException i) {
        System.out.println(i);
    }
```

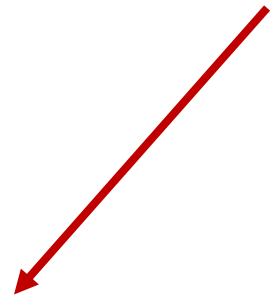
Clientプログラムのスケルトン

```
private Socket socket = null;
private DataInputStream input = null;
private DataOutputStream out = null;
// establish a connection
      try {
         socket = new Socket(address, port);
         System.out.println("Connected");

         // takes input from terminal
         input = new DataInputStream(System.in);

         // sends output to the socket
         out = new DataOutputStream(
            socket.getOutputStream());
      }
      catch (UnknownHostException u) {
               return;
      }
      catch (IOException i) {
         System.out.println(i);
         return;
      }
```

```
      // close the connection
      try {
         input.close();
         out.close();
         socket.close();
      }
```

Server側のプログラミング

```
//サーバーが開始されました
ServerSocket server = new ServerSocket(port);
```



TCPポート

Server側のプログラミング

```
//クライアントが受け入れました・接続された
Socket socket = server.accept();
```

コミュニケーション

```
in = new DataInputStream(
            new BufferedInputStream(socket.getInputStream()));

        String line = "";

        // reads message from client until "Over" is sent
        while (!line.equals("Over"))
        {
            try
            {
                line = in.readUTF();
                System.out.println(line);

            }
            catch(IOException i)
            {
                System.out.println(i);
            }
        }
```

Serverプログラムのスケルトン

```
ServerSocket server = new ServerSocket(port);
 System.out.println("Server started");

System.out.println("Waiting for a client ...");

Socket socket = server.accept();
System.out.println("Client accepted");

// takes input from the client socket
DataInputStream in = new DataInputStream(
            new BufferedInputStream(socket.getInputStream()));

// close the connection
      try {
         input.close();
         out.close();
         socket.close();
      }
```