

BÀI 06 TCP Socket

DatagramExample

```
3  import java.net.DatagramPacket;
4  import java.net.InetAddress;
5  import java.net.UnknownHostException;
6
7  public class DatagramExample {
8
9      public static void main(String[] args) {
10         String s = "Hello Hutech";
11         byte[] data = s.getBytes();
12         try {
13             InetAddress ia = InetAddress.getByName("www.hutech.edu.vn");
14             int port = 7;
15             DatagramPacket dp;
16             dp = new DatagramPacket(data, data.length, ia, port);
17             System.out.println("Address: " + dp.getAddress());
18             System.out.println("Port: " + dp.getPort());
19             System.out.println("Length: " + dp.getLength());
20             System.out.println("Data: ");
21             s = new String(dp.getData(), dp.getOffset(), dp.getLength());
22             System.out.println(s);
23         } catch (UnknownHostException e) {
24             System.out.println(e);
25         }
26     }
27 }
28
```

UDPClient

```
3  import java.io.BufferedReader;
4  import java.io.InputStreamReader;
5  import java.net.DatagramPacket;
6  import java.net.DatagramSocket;
7  import java.net.InetAddress;
8
9  public class UDPClient {
10
11      public static void main(String args[]) throws Exception {
12
13         DatagramSocket clientSocket = new DatagramSocket();
14
15         InetAddress IPAddress = InetAddress.getByName("localhost");
16
17         BufferedReader inFromUser
18             = new BufferedReader(new InputStreamReader(System.in));
19
20         byte[] sendData = new byte[1024];
21         byte[] receiveData = new byte[1024];
22
23         String sentence = inFromUser.readLine();
24         sendData = sentence.getBytes();
25         DatagramPacket sendPacket
26             = new DatagramPacket(sendData, sendData.length, IPAddress, 9876);
27
28         clientSocket.send(sendPacket);
29
30         DatagramPacket receivePacket
31             = new DatagramPacket(receiveData, receiveData.length);
32     }
33 }
```

```

33         clientSocket.receive(receivePacket);
34
35         String modifiedSentence
36             = new String(receivePacket.getData());
37
38         System.out.println("FROM SERVER: " + modifiedSentence);
39         clientSocket.close();
40     }
41 }
42

```

UDPServer

```

3  import java.net.DatagramPacket;
4  import java.net.DatagramSocket;
5  import java.net.InetAddress;
6
7  public class UDPServer {
8
9      public static void main(String args[]) throws Exception {
10
11          DatagramSocket serverSocket = new DatagramSocket(9876);
12
13          byte[] receiveData = new byte[1024];
14          byte[] sendData = new byte[1024];
15
16          while (true) {
17
18              DatagramPacket receivePacket
19                  = new DatagramPacket(receiveData, receiveData.length);
20              serverSocket.receive(receivePacket);
21              String sentence = new String(receivePacket.getData());
22
23              InetAddress IPAddress = receivePacket.getAddress();
24
25              int port = receivePacket.getPort();
26
27              String capitalizedSentence = sentence.toUpperCase();
28
29              sendData = capitalizedSentence.getBytes();
30
31              DatagramPacket sendPacket
32                  = new DatagramPacket(sendData, sendData.length, IPAddress,
33                                      port);
34
35              serverSocket.send(sendPacket);
36          }
37      }
38  }
39
40

```

UDPClientChatEx1

```

3 import java.io.BufferedReader;
4 import java.io.IOException;
5 import java.io.InputStreamReader;
6 import java.net.DatagramPacket;
7 import java.net.DatagramSocket;
8 import java.net.InetAddress;
9 import java.net.SocketException;
10 import java.net.UnknownHostException;
11
12 public class UDPCliChatEx1 {
13
14     public static void main(String[] args) throws UnknownHostException, IOException {
15         BufferedReader in = new BufferedReader(new InputStreamReader(System.in));
16         InetAddress ip_server = InetAddress.getByName("localhost");
17         byte[] buffer = new byte[1024];
18         int port_server = 8888;
19         try {
20             DatagramSocket socket = new DatagramSocket();
21             DatagramPacket packet = new DatagramPacket(buffer, buffer.length);
22             boolean client_end = false;
23             String st_send = "", st_receive = "";
24
25             //Send data
26             do {
27                 byte[] send_data;
28                 System.out.print("\nClient: ");
29                 st_send = in.readLine();
30                 send_data = st_send.getBytes();
31                 DatagramPacket send_packet = new DatagramPacket(send_data, send_data.length, ip_server, port_server);
32                 socket.send(send_packet);
33
34                 if (!st_send.equalsIgnoreCase("bye")) {
35                     socket.receive(packet);
36
37                     st_receive = new String(packet.getData(), 0, packet.getLength());
38                     System.out.print("Server: " + st_receive);
39                 }
40
41                 if (st_send.equalsIgnoreCase("bye") || st_receive.equalsIgnoreCase("bye")) {
42                     client_end = true;
43                 }
44             } while (!client_end);
45             socket.close();
46         } catch (SocketException e) {
47         }
48     }
49 }
50

```

UDPServerChatEx1

```

3  import java.io.BufferedReader;
4  import java.io.IOException;
5  import java.io.InputStreamReader;
6  import java.net.DatagramPacket;
7  import java.net.DatagramSocket;
8  import java.net.InetAddress;
9  import java.net.SocketException;
10
11  public class UDPServerChatEx1 {
12
13      public static void main(String[] args) throws IOException {
14          BufferedReader in = new BufferedReader(new InputStreamReader(System.in));
15          byte[] buffer = new byte[1024];
16          int port = 8888;
17          try {
18              DatagramSocket socket = new DatagramSocket(port);
19              System.out.println("Server connect port 8888 ...");
20              DatagramPacket packet = new DatagramPacket(buffer, buffer.length);
21              boolean server_end = false;
22              String st_send = "", st_receive = "";
23
24              //Receive data
25              do {
26                  socket.receive(packet);
27                  st_receive = new String(packet.getData(), 0, packet.getLength());
28                  System.out.print("Client: " + st_receive);
29
30                  if (!st_receive.equalsIgnoreCase("bye")) {
31                      InetAddress ip_client = packet.getAddress();
32                      int port_client = packet.getPort();
33                      byte[] send_data;
34                      System.out.print("\nServer: ");
35                      st_send = in.readLine();
36
37                      send_data = st_send.getBytes();
38                      DatagramPacket send_packet;
39                      send_packet = new DatagramPacket(send_data, send_data.length, ip_client, port_client);
40                      socket.send(send_packet);
41                  }
42
43                  if (st_receive.equalsIgnoreCase("bye") || st_send.equalsIgnoreCase("bye")) {
44                      server_end = true;
45                  }
46              } while (!server_end);
47          } catch (SocketException e) {
48          }
49      }
50  }
51
52

```

UDPPortScanner

```
3  import java.net.DatagramSocket;
4  import java.net.SocketException;
5
6  public class UDPPortScanner {
7
8      public static void main(String[] args) {
9          for (int port = 0; port <= 65535; port++) {
10             try {
11                 DatagramSocket server = new DatagramSocket(port);
12                 server.close();
13             } catch (SocketException e) {
14                 System.out.println("Port: " + port + " in use");
15             }
16         }
17     }
18 }
19
```