150 Java의 정석定石 3판 - 연습문제 풀이

# Chapter

네트워킹

Networking

## [ 연습문제 ]

[16-1]

ip주소가

192.168.10.100이고 서브넷 마스크(subnet mask)가

255.255.255.0일

때, 네트워크 주소와 호스트 주소를 계산하여 화면에 출력하는 프로그램을 작성하시오. 단, 비트연산자를 사용해서 계산해야 한다.

네트워크 주소:192.168.10.0. 호스트 주소:0.0.0.100.

[실행결과]

[16-2] 다음 중 TCP의 특징이 아닌 것은?

1. 전화와 같은 1:1 연결기반의 프로토콜이다.
2. 데이터의 전송순서가 보장된다.
3. UDP보다 전송속도가 빠르다.
4. 데이터의 수신여부를 확인한다.

[16-3]

TextField에

URL을 입력하고

Enter키를 누르면 해당 페이지의 소스를 보여주는

'Source Viwer'프로그램이다. 오.

예제15-4를 참고해서

(1)에 알맞은 코드를 넣어 완성하시

import java.net.\*; import java.io.\*; import java.awt.\*;

import java.awt.event.\*;

class SourceViewer extends Frame { TextField tf = new TextField(); TextArea ta = new TextArea();

SourceViewer(String title) { super(title);

add(tf, "North");

add(ta, "Center");

tf.addActionListener(new ActionListener(){ public void actionPerformed(ActionEvent ae) {

displaySource();

}

});

[연습문제]**/ch16/Exercise16\_3.java**

addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent we) { System.exit(0);

}

});

setBounds(500, 200, 500, 300);

setVisible(true);

}

void displaySource() { URL url = null;

BufferedReader input = null;

String address = tf.getText().trim(); String line = "";

ta.setText(""); try {

if(!address.startsWith("http://"))

address = "http://"+address;

**/\***

**(1)** 알맞은 코드를 넣어 완성하시오.

**\*/**

input.close();

} catch(Exception e) { ta.setText("유효하지 않은 URL입니다.");

}

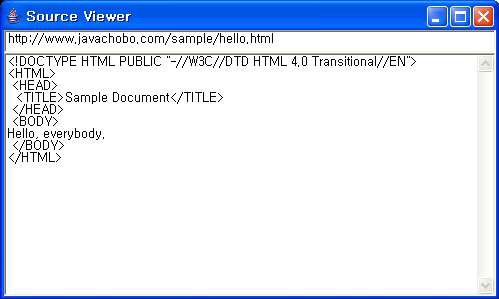
} // displaySource()

public static void main(String[] args) {

SourceViewer mainWin = new SourceViewer("Source Viewer");

}

}



[실행결과]

[16-4] 다음의 코드는

TCP통신을 하는 예제16-6, 16-7을 결합하여

GUI채팅프로그램을

작성한 것이다. (1)~(4)에 알맞은 코드를 넣어 프로그램을 완성하시오.

[연습문제]**/ch16/ChatServer.java**

import java.net.\*; import java.io.\*; import java.awt.\*;

import java.awt.event.\*;

class ChatServer extends Frame { String nickname = "";

DataOutputStream out; DataInputStream in;

Panel p = new Panel(); TextArea ta = new TextArea();

TextField tf = new TextField();

ChatServer(String nickname) { super("Chatting"); this.nickname = nickname;

p.setLayout(new BorderLayout()); p.add(tf, "Center");

add(ta, "Center");

add(p, "South");

addWindowListener(new WindowAdapter(){

public void windowClosing(WindowEvent e) { System.exit(0);

}

});

EventHandler handler = new EventHandler(); ta.addFocusListener(handler); tf.addFocusListener(handler); tf.addActionListener(handler);

ta.setEditable(false); setBounds(200, 200, 300, 200);

setVisible(true); tf.requestFocus();

}

void startServer() {

ServerSocket serverSocket = null; Socket socket = null;

try {

**/\***

**(1)** 아래의 로직에 맞게 코드를 작성하시오**.**

1. 서비소켓을 생성하여 **7777**번 포트와 결합시킨다**.**
2. **ta**에 **"**서버가 준비되었습니다**."**라고 보여준다**.**
3. 상대방의 연결을 기다린다**.**
4. **ta**에 **"**상대방과 연결되었습니다**."**라고 보여준다**.**

ta.append("\r\n" +"상대방과 연결되었습니다.");

1. 연결된 상대방 소켓의 입력스트림과 출력스트립을 얻어온다**.**
2. 반복문을 이용해서 입력스트림이 **null**이 아닌 동안 입력스트림으로부터 데이터를 읽어서 변수 **msg**에 저장한다**.**

**\*/**

**ta.append("\r\n" + msg);**

} catch (Exception e) { e.printStackTrace();

}

}

public static void main(String[] args) { if(args.length != 1) {

System.out.println("USAGE : java ChatServer NICKNAME"); System.exit(0);

}

ChatServer chatWin = new ChatServer(args[0]); chatWin.startServer();

} // main

class EventHandler extends FocusAdapter implements ActionListener { public void actionPerformed(ActionEvent ae) {

String msg = tf.getText();

if("".equals(msg)) return;

**/\***

**(2)** 알맞은 코드를 넣어 완성하시오**.**

**\*/**

ta.append("\r\n" + nickname +">"+ msg); tf.setText("");

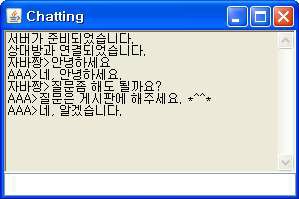
}

public void focusGained(FocusEvent e) { tf.requestFocus();

}

} // class EventHandler

} // class



[실행결과]

[연습문제]**/ch16/ChatClient.java**

import java.net.\*; import java.io.\*; import java.awt.\*;

import java.awt.event.\*;

class ChatClient extends Frame { String nickname = "";

String serverIp = ""; int serverPort = 0;

DataOutputStream out; DataInputStream in;

Panel p = new Panel(); TextArea ta = new TextArea();

TextField tf = new TextField();

ChatClient(String nickname, String serverIp, String serverPort) { super("Chatting with " + serverIp +":" + serverPort); this.nickname = nickname;

this.serverIp = serverIp;

this.serverPort = Integer.parseInt(serverPort); setBounds(600, 200, 300, 200);

p.setLayout(new BorderLayout()); p.add(tf, "Center");

add(ta, "Center");

add(p, "South");

addWindowListener(new WindowAdapter(){

public void windowClosing(WindowEvent e) { System.exit(0);

}

});

EventHandler handler = new EventHandler(); ta.addFocusListener(handler); tf.addFocusListener(handler); tf.addActionListener(handler);

ta.setEditable(false);

setVisible(true); tf.requestFocus();

}

void startClient() { try {

**/\***

1. 아래의 로직에 맞게 코드를 작성하시오**.**
2. 소켓을 생성하여 **serverIp**의 **serverPort**에 연결한다**.**
3. **ta**에 **"**상대방과 연결되었습니다**."**라고 보여준다**.**

ta.setText("상대방과 연결되었습니다.");

1. 연결된 상대방 소켓의 입력스트림과 출력스트립을 얻어온다**.**
2. 반복문을 이용해서 입력스트림이 **null**이 아닌 동안

입력스트림으로부터 데이터를 읽어서 변수 **msg**에 저장한다**.**

**\*/**

ta.append("\r\n" + msg);

} catch(ConnectException ce) { ta.setText("상대방과 연결할 수 없습니다."); ce.printStackTrace();

} catch(IOException ie) { ie.printStackTrace();

} catch(Exception e) { e.printStackTrace();

}

}

public static void main(String[] args) { if(args.length != 3) {

System.out.println("USAGE : java ChatClient NICKNAME SERVER\_IP SERVER\_PORT");

System.exit(0);

}

ChatClient chatWin = new ChatClient(args[0],args[1],args[2]); chatWin.startClient();

} // main

class EventHandler extends FocusAdapter implements ActionListener { public void actionPerformed(ActionEvent ae) {

String msg = tf.getText();

if("".equals(msg)) return;

**/\***

**(4)** 알맞은 코드를 넣어 완성하시오**.**

**\*/**

ta.append("\r\n" + nickname +">"+ msg); tf.setText("");

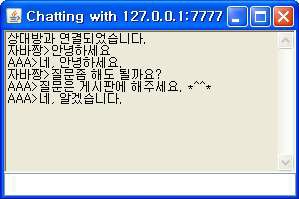
}

public void focusGained(FocusEvent e) { tf.requestFocus();

}

} // class EventHandler

} // class



[실행결과]