William Kenji Kiplinger
https://github.com/Kenjum/CS380-P4/

IPv6Client.java

```
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.net.InetAddress;
import java.net.Socket;
import java.nio.ByteBuffer;

import javax.xml.bind.DatatypeConverter;

public class Ipv6Client {

        public static void main(String[] args) throws IOException {
                try {
                        Socket socket = new Socket("18.221.102.182", 38004);
                        System.out.println("connected to server");
                        InputStream is = socket.getInputStream();
                        OutputStream os = socket.getOutputStream();

                        // IPv6 Header
                        byte[] ipv6 = new byte[40]; // size of signed 8 bit
                        ipv6[0] = 96; // Version (4 bits)
                        // [0110] [0000
                        ipv6[1] = 0; // Traffic Class (8 bits)
                        // [0110] [0000 0000] [0000
                        ipv6[2] = 0; // Flow Label (20 bits)
                        // [0110] [0000 0000] [0000 0000 0000
                        ipv6[3] = 0;
                        // [0110] [0000 0000] [0000 0000 0000 0000 0000]
                        // Payload length (16 bits)
                        ipv6[6] = 17; // Next Header (8 bits) UDP protocol value
                        ipv6[7] = 20; // Hop Limit (8 bits)

                        // Implement assuming it is a valid IPv4 address that has been
                        // extended to IPv6 for a device that does not use IPv6
                        InetAddress sourceAddr = InetAddress.getLocalHost();
                        byte[] source = sourceAddr.getAddress();

                        ipv6[18] = -1; // Source Address (128 bits)
                        ipv6[19] = -1; // -1 gives 0xFF
                        ipv6[20] = source[0]; // 127.
                        ipv6[21] = source[1]; // 0.
                        ipv6[22] = source[2]; // 0.
                        ipv6[23] = source[3]; // 1 works as well
```

```
            ipv6[34] = -1; // Destination Address (128 bits)
            ipv6[35] = -1; // -1 gives FF
            // This is the socket address 18.221.102.182
            ipv6[36] = 18;
            ipv6[37] = (byte) 221;
            ipv6[38] = 102;
            ipv6[39] = (byte) 182;

            // for each iteration that doubles
            int pLength = 1; // size of data
            for (int i = 1; i <= 12; i++) {
                ipv6[4] = 0; // Payload length
                ipv6[5] = 0; // Payload length
                pLength *= 2;
                // adjusting payload size
                ipv6[4] = (byte) ((pLength >> 8) & 0xFF);
                ipv6[5] = (byte) ((pLength & 0xFF));

                // 40 bytes of header and then size of data that will be sent
                // and checked
                byte[] dataSend = new byte[(int) (ipv6.length + pLength)];
                for (int j = 0; j < 40; j++){
                    dataSend[j] = ipv6[j];
                }
                os.write(dataSend);                 //send data to server

                //Receive response from server
                byte[] response = new byte[4];
                response[0] = (byte) is.read();
                response[1] = (byte) is.read();
                response[2] = (byte) is.read();
                response[3] = (byte) is.read();

                System.out.println("data length: " + pLength + "\nResponse: 0x"
                        + DatatypeConverter.printHexBinary(response) + "\n");
            }

            socket.close();
        } catch (Exception e) {
            System.out.println("ERROR: An error occured to the connection");
        }
    }
}
```