

Soluzione Esercizio 2

```
1 // RichiestaDiCompetenze.java
2 import com.thoughtworks.xstream.*;
3
4 public class RichiestaDiCompetenze {
5
6     private final String descrizione;
7     private final int portaRichiedente;
8     private int portaProponente = 0;
9
10    public RichiestaDiCompetenze(String d, int pr) {
11        descrizione = d;
12        portaRichiedente = pr;
13    }
14
15    public RichiestaDiCompetenze(String xml) {
16        RichiestaDiCompetenze p = (RichiestaDiCompetenze) (new XStream()).fromXML(xml);
17        descrizione = p.descrizione;
18        portaRichiedente = p.portaRichiedente;
19        portaProponente = p.portaProponente;
20    }
21    public int restituisciPortaRichiedente() { return portaRichiedente; }
22
23    public void inserisciPortaProponente(int pp) {
24        if (!esistePortaProponente())
25            portaProponente = pp;
26    }
27
28    public boolean esistePortaProponente() { return (portaProponente != 0); }
29
30    public String toString() {
31        return (new XStream()).toXML(this);
32    }
33 }
34
35
36 1 // AgenziaDelLavoro.java
37 2
38 3 import java.net.*;
39 4 import java.io.*;
40 5 import java.sql.*;
41 6
42 7 public class AgenziaDelLavoro {
43 8     private static int portaRicezione, portaInvio;
44 9
45 10    public static void invia(int porta, RichiestaDiCompetenze r) {
46 11        try (Socket sock = new Socket("localhost", porta);
47 12            DataOutputStream dos =
48 13                new DataOutputStream(sock.getOutputStream());
49 14        ) { dos.writeUTF(r.toString());
50 15        } catch (IOException e) { e.printStackTrace();}
51 16        visualizzaSuLog(java.time.LocalDateTime.now().toString(), "invia", r.toString()); //2,...
52 17    }
53 18
54 19    public static RichiestaDiCompetenze ricevi() {
55 20        RichiestaDiCompetenze r = null;
56 21        try (
57 22            ServerSocket servsock = new ServerSocket(portaRicezione);
58 23            Socket sock = servsock.accept();
59 24            DataInputStream dis =
60 25                new DataInputStream(sock.getInputStream());
61 26        ) {
```

```

27     r = new RichiestaDiCompetenze(dis.readUTF());
28 } catch (IOException e) { e.printStackTrace();}
29 visualizzaSuLog(java.time.LocalDateTime.now().toString(), "ricevi", r.toString()); //5,8,11
30 return r;
31 }
32
33 public static void visualizzaSuLog(String marcaTemporale, String operazione,
                                   String risultato) {
34     try (
35         Connection co = DriverManager.getConnection(
36             "jdbc:mysql://localhost:3306/agenziadellavoro", "root","");
37         PreparedStatement ps = co.prepareStatement("INSERT INTO log VALUES (?, ?, ?, ?)");
38     ) {
39         ps.setString(1, marcaTemporale); ps.setInt(2, portaRicezione);
40         ps.setString(3, operazione); ps.setString(4, risultato);
41         ps.executeUpdate();
42     } catch (SQLException e) { System.err.println(e.getMessage());}
43
44 public static void main(String[] args) {
45     portaRicezione = Integer.parseInt(args[0]);
46     portaInvio = Integer.parseInt(args[1]);
47     RichiestaDiCompetenze r;
48     if (args.length==3 && args[2].endsWith("?")){ // richiedente
49         visualizzaSuLog(java.time.LocalDateTime.now().toString(), "inizio", "*****");
50         r = new RichiestaDiCompetenze(args[2], portaRicezione); //1
51         visualizzaSuLog(java.time.LocalDateTime.now().toString(), "nuovaRichiesta", r.toString()); //2
52         invia(portaInvio, r); //3
53         r = ricevi(); //7
54         invia(portaInvio, r); //9
55         ricevi(); //13
56         visualizzaSuLog(java.time.LocalDateTime.now().toString(), "fine", "*****");
57     } else { // altri
58         r = ricevi(); //4 oppure 10
59         if (args.length==2) { // 6.b oppure //10
60             invia(portaInvio, r); //6.b.1 oppure 12
61             if (!r.esistePortaProponente()) { // deve ancora fare 10
62                 r = ricevi(); //10
63                 invia(portaInvio, r); //12
64             } // se ha fatto 10 allora non deve fare altro
65         } else if (args[2].endsWith("!")) { // 6.a ha competenze
66             r.inserisciPortaProponente(portaRicezione); //6.a.1
67             invia(r.restituiscePortaRichiedente(), r); //6.a.2
68             r = ricevi(); //10
69             invia(portaInvio, r); //12
70         }
71     }
72 }
73 }

```

Log (formato contratto)

| marcatemporale | idprocesso | operazione | risultato |
|----------------|------------|-------------------------|-----------|
| 09:22:04.490 | 8080 | inizio ***** | |
| 09:22:04.887 | 8080 | nuovaR Programmatore C? | 8080 0 |
| 09:22:05.106 | 8080 | invia Programmatore C? | 8080 0 |
| 09:22:05.375 | 8081 | ricevi Programmatore C? | 8080 0 |
| 09:22:05.770 | 8081 | invia Programmatore C? | 8080 0 |
| 09:22:06.027 | 8082 | ricevi Programmatore C? | 8080 0 |
| 09:22:06.413 | 8082 | invia Programmatore C? | 8080 0 |
| 09:22:06.440 | 8080 | ricevi Programmatore C? | 8080 0 |

```
09:22:06.524 8080 invia Programmatore C?      8080  0
09:22:06.527 8081 ricevi Programmatore C?      8080  0
09:22:06.621 8081 invia Programmatore C?      8080  0
09:22:06.624 8082 ricevi Programmatore C?      8080  0
09:22:06.707 8082 invia Programmatore C?      8080  0
09:22:06.709 8080 ricevi Programmatore C?      8080  0
09:22:06.762 8080 fine *****
09:22:08.582 8080 inizio *****
09:22:08.973 8080 nuovaR Programmatore C?      8080  0
09:22:09.206 8080 invia Programmatore C?      8080  0
09:22:09.470 8081 ricevi Programmatore C?      8080  0
09:22:09.862 8081 invia Programmatore C?      8080 8081
09:22:09.891 8080 ricevi Programmatore C?      8080 8081
09:22:09.981 8080 invia Programmatore C?      8080 8081
09:22:09.983 8081 ricevi Programmatore C?      8080 8081
09:22:10.072 8081 invia Programmatore C?      8080 8081
09:22:10.341 8082 ricevi Programmatore C?      8080 8081
09:22:10.727 8082 invia Programmatore C?      8080 8081
09:22:10.731 8080 ricevi Programmatore C?      8080 8081
09:22:10.814 8080 fine *****
09:22:13.113 8080 inizio *****
09:22:13.493 8080 nuovaR Programmatore C?      8080  0
09:22:13.701 8080 invia Programmatore C?      8080  0
09:22:13.966 8081 ricevi Programmatore C?      8080  0
09:22:14.356 8081 invia Programmatore C?      8080  0
09:22:14.620 8082 ricevi Programmatore C?      8080  0
09:22:15.018 8082 invia Programmatore C?      8080 8082
09:22:15.044 8080 ricevi Programmatore C?      8080 8082
09:22:15.137 8080 invia Programmatore C?      8080 8082
09:22:15.140 8081 ricevi Programmatore C?      8080 8082
09:22:15.210 8081 invia Programmatore C?      8080 8082
09:22:15.213 8082 ricevi Programmatore C?      8080 8082
09:22:15.308 8082 invia Programmatore C?      8080 8082
09:22:15.311 8080 ricevi Programmatore C?      8080 8082
09:22:15.388 8080 fine *****
```