```java
 1 package model.helpers;
 2
 3 import java.io.BufferedReader;
40
41 public class Utils {
42
43     public static final String AUTH_URL = "https://
   www.eecs.yorku.ca/~roumani/servers/auth/oauth.cgi";
44
45     public static String doHttpGET(Map<String, String> params,
   String Url, String contentType) throws Exception
46     {
47         StringBuilder result = new StringBuilder();
48         for (Map.Entry<String, String> entry: params.entrySet()) {
49             result.append(URLEncoder.encode(entry.getKey(),
   "UTF-8"));
50             result.append("=");
51             result.append(URLEncoder.encode(entry.getValue(),
   "UTF-8"));
52             result.append("&");
53         }
54         String parameters = result.toString();
55
56         URL url = new URL(Url + "?" + parameters);
57         HttpURLConnection con = (HttpURLConnection)
   url.openConnection();
58         con.setRequestMethod("GET");
59         con.setRequestProperty("Content-Type", contentType);
60         con.setDoOutput(true);
61
62         BufferedReader in = new BufferedReader(
63                 new InputStreamReader(con.getInputStream()));
64
65         String inputLine;
66         StringBuffer response = new StringBuffer();
67
68         while ((inputLine = in.readLine()) != null) {
69             response.append(inputLine);
70         }
71         in.close();
72         con.disconnect();
73
74         return response.toString();
75     }
```

```java
76
77     public static void doAuth(String username, String pass, String
   back) throws Exception {
78         HttpURLConnection connection = (HttpURLConnection) new
   URL(AUTH_URL + "?back=" + back).openConnection();
79         String encoded =
   Base64.getEncoder().encodeToString((username+":"+pass).getBytes(St
   andardCharsets.UTF_8));   //Java 8
80         connection.setRequestProperty("Authorization", "Basic
   "+encoded);
81     }
82
83     public static void renderXsltTHtml(String xml, InputStream
   xslIs, Writer out) throws Exception {
84         DocumentBuilder factory_db =
   DocumentBuilderFactory.newInstance().newDocumentBuilder();
85         Document doc = factory_db.parse(new InputSource(new
   StringReader(xml)));
86         Source xmlSource = new DOMSource(doc);
87
88         TransformerFactory transfomerFactory =
   TransformerFactory.newInstance();
89         Templates cachedXSLT = transfomerFactory.newTemplates(new
   StreamSource(xslIs));
90         Transformer transformer = cachedXSLT.newTransformer();
91
92         transformer.transform(xmlSource, new StreamResult(out));
93     }
94
95     public static double round(double value, int places) {
96         if (places < 0) throw new IllegalArgumentException();
97
98         BigDecimal bd = new BigDecimal(value);
99         bd = bd.setScale(places, RoundingMode.HALF_UP);
100         return bd.doubleValue();
101     }
102
103     public static Boolean isValidUser(Account user) {
104         boolean result = false;
105         if (user != null && user.getName() != null && !
   user.getName().isEmpty()
106                 && user.getUsername() != null && !
   user.getUsername().isEmpty()) {
107             result = true;
```

```
108         }
109         return result;
110     }
111
112 }
113
```