Java

Allgemeines

Die folgende Java-Klasse `DDBRest` demonstriert den Abruf von Daten (speziell des Views eines Datensatzes) über die API der Deutschen Digitalen Bibliothek. Die Quelltextdatei `DDBRest.java` kann heruntergeladen und mit den folgen Befehlen compiliert sowie ausgeführt werden. Der Dateipfad Java-Kompiler `javac` sollte in der Umgebungsvariablen PATH des Betriebssystems definiert sein, andernfalls muss der vollständige Pfad mit Dateinamen (bspw. `C:\Programme\Java\jdk1.6.0_45\bin\javac`) angegeben werden.

Download

UNIX & Windows

```
> javac DDBRest.java
> java DDBRest
```

Es ist zu beachten, dass nur Daten abgefragt werden können, wenn ein gültiger API Key angegeben wird (Variable `key`). Andernfalls wird eine Exception mit dem Hinweis „Forbidden“ geworfen.

Nach erfolgreichem Ausführen des Programms wird auf der Konsole dreimal der gleiche Datensatz ausgegeben:

1. Datensatz im XML-Format mit Authentifizierung über den HTTP request header
2. Datensatz im JSON-Format mit Authentifizierung über den HTTP request header
3. Datensatz im JSON-Format mit Authentifizierung über den Query Parameter

Quelltext

Klasse `DDBRest`

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLEncoder;
import java.util.HashMap;

public class DDBRest {

    /**
     * Opens a HTTP connection, gets the response and converts into to a String.
     *
     * @param urlStr Servers URL
     * @param properties Keys and values for HTTP request properties
     * @return Servers response
     * @throws IOException  If connection could not be established or response code is
!=200
     */
    public static String httpGet(String urlStr, HashMap<String, String> properties)
throws IOException {
        if (properties == null) {
            properties = new HashMap<String, String>();
        }
```

```
        // open HTTP connection with URL
        URL url = new URL(urlStr);
        HttpURLConnection conn = (HttpURLConnection) url.openConnection();
        // set properties if any do exist
        for (String key : properties.keySet()) {
            conn.setRequestProperty(key, properties.get(key));
        }
        // test if request was successful (status 200)
        if (conn.getResponseCode() != 200) {
            throw new IOException(conn.getResponseMessage());
        }
        // buffer the result into a string
        InputStreamReader isr = new InputStreamReader(conn.getInputStream(), "UTF-8");
        BufferedReader br = new BufferedReader(isr);
        StringBuilder sb = new StringBuilder();
        String line;
        while ((line = br.readLine()) != null) {
            sb.append(line);
        }
        br.close();
        isr.close();
        conn.disconnect();
        return sb.toString();
    }

    public static void main(String[] args) throws IOException {

        // URL of DDB server with dataset ID and requested method
        final String url =
"https://api.deutsche-digitale-bibliothek.de/items/OAXO2AGT7YH35YYHN3YKBXJMEI77W3FF/vi
ew";
        final String key = "abcdefgh12345678";

        // get XML data via HTTP request header authentication
        String httpXmlResult = httpGet(url, new HashMap<String, String>() {
            {
                put("Authorization", "OAuth oauth_consumer_key=\"" + key + "\"");
                put("Accept", "application/xml");
            }
        });
        System.out.println(httpXmlResult); // print results

        // get JSON data via HTTP request header authentication
        String httpJsonResult = httpGet(url, new HashMap<String, String>() {
            {
                put("Authorization", "OAuth oauth_consumer_key=\"" + key + "\"");
                put("Accept", "application/json");
            }
        });
        System.out.println(httpJsonResult); // print results

        // get JSON data via query parameter authentication
        // remember: use URL encoded Strings online -> URLEncoder.encode(s, enc)
        String queryJsonURL = url + "?oauth_consumer_key=" + URLEncoder.encode(key,
"UTF-8");
        String queryJsonResult = httpGet(queryJsonURL, null);
```

```
        System.out.println(queryJsonResult); // print results
    }
}
```