

# Lecture 2: extra

TIES4560 SOA and Cloud Computing  
Autumn 2023



# REST Web Services

*Get Access Code from DropBox after Authorization.*

```
...
public void sendRequest(String str) throws URISyntaxException, IOException {
    // basically builds corresponding GET request that will be returned to the front-end..
    String appKey = "b7c81t23025nus6"; //get from AppConsole when create the DropBox App
    String redirectURI="http://localhost:8080/MyDBoxClient/"; //any url to where you want to redirect the user
    URI uri=new URI("https://www.dropbox.com/oauth2/authorize");

    StringBuilder requestUri=new StringBuilder(uri.toString());
    requestUri.append("?client_id=");
    requestUri.append(URLEncoder.encode(appKey, "UTF-8"));
    requestUri.append("&response_type=code");
    requestUri.append("&redirect_uri="+redirectURI.toString());

    queryResult = requestUri.toString();
}
...
```

```
...
function doQuery_back(result)
{
    window.location = result;
}
...
```

*Get Access Token from DropBox.*

```
...
public void accessToken(String codeStr) throws URISyntaxException, IOException {

    String code = ""+codeStr; //code get from previous step
    String appKey = "b7c81t2d02vnes6"; //get from AppConsole when create the DropBox App
    String appSecret = "bnsslfxtb4hwkh7"; //get from AppConsole when create the DropBox App
    String redirectURI="http://localhost:8080/MyDBoxClient/"; //any url to where you want to redirect the user

    StringBuilder tokenUri=new StringBuilder("code=");
    tokenUri.append(URLEncoder.encode(code,"UTF-8"));
    tokenUri.append("&grant_type=");
    tokenUri.append(URLEncoder.encode("authorization_code","UTF-8"));
    tokenUri.append("&client_id=");
    tokenUri.append(URLEncoder.encode(appKey,"UTF-8"));
    tokenUri.append("&client_secret=");
    tokenUri.append(URLEncoder.encode(appSecret,"UTF-8"));
    tokenUri.append("&redirect_uri="+redirectURI);

    URL url=new URL("https://api.dropbox.com/oauth2/token");

    HttpURLConnection connection = (HttpURLConnection) url.openConnection();
    try {
        connection.setDoOutput(true);
        connection.setRequestMethod("POST");
        connection.setRequestProperty("Content-Type", "application/x-www-form-urlencoded");
        connection.setRequestProperty("Content-Length", "" + tokenUri.toString().length());

        OutputStreamWriter outputStreamWriter = new OutputStreamWriter(connection.getOutputStream());
        outputStreamWriter.write(tokenUri.toString());
        outputStreamWriter.flush();

    }
    ...
}
```

# REST Web Services

*Get Access Token from DropBox.*

```
...
BufferedReader in = new BufferedReader(new InputStreamReader(connection.getInputStream()));
String inputLine;
StringBuffer response = new StringBuffer();

while ((inputLine = in.readLine()) != null) {
    response.append(inputLine);
}
in.close();

queryResult = response.toString();

} finally {
    connection.disconnect();
}
}
...
```

*Get Account Info from DropBox.*

```
public void getAccountInfo(String tokenStr, String accountIDStr) throws URISyntaxException, IOException {

    String access_token = ""+tokenStr;
    String content = "{\"account_id\": \"" + accountIDStr + "\"}";
    URL url = new URL("https://api.dropboxapi.com/2/users/get_account");

    HttpURLConnection connection = (HttpURLConnection) url.openConnection();
    try {
        connection.setDoOutput(true);
        connection.setRequestMethod("POST");
        connection.setRequestProperty("Authorization", "Bearer "+access_token);
        connection.setRequestProperty("Content-Type", "application/json");
        connection.setRequestProperty("Content-Length", "" + content.length());

        OutputStreamWriter outputStreamWriter = new OutputStreamWriter(connection.getOutputStream());
        outputStreamWriter.write(content);
        outputStreamWriter.flush();

        BufferedReader in = new BufferedReader(new InputStreamReader(connection.getInputStream()));
        String inputLine;
        StringBuffer response = new StringBuffer();

        while ((inputLine = in.readLine()) != null) {
            response.append(inputLine);
        }
        in.close();

        queryResult = response.toString();
    } finally {
        connection.disconnect();
    }
}
```

*Upload a file to DropBox.*

```
public void uploadFile(String token, String path) throws URISyntaxException, IOException {

    String access_token = ""+token;
    String sourcePath = ""+path; //required file path on local file system
    Path pathFile = Paths.get(sourcePath);
    byte[] data = Files.readAllBytes(pathFile);
    String content = "{ \"path\": \"/MyDBoxClient_App01_files/images/image_initial_uploaded.png\", \"mode\":
        \"add\", \"autorename\": true, \"mute\": false, \"strict_conflict\": false}";

    URL url = new URL("https://content.dropboxapi.com/2/files/upload");

    HttpURLConnection connection = (HttpURLConnection) url.openConnection();
    try {
        connection.setDoOutput(true);
        connection.setRequestMethod("POST");
        connection.setRequestProperty("Authorization", "Bearer "+access_token);
        connection.setRequestProperty("Content-Type", "application/octet-stream");
        connection.setRequestProperty("Dropbox-API-Arg", "" + content);
        connection.setRequestProperty("Content-Length", String.valueOf(data.length));

        OutputStream outputStream = connection.getOutputStream();
        outputStream.write(data);
        outputStream.flush();
        BufferedReader in = new BufferedReader(new InputStreamReader(connection.getInputStream()));
        String inputLine;
        StringBuffer response = new StringBuffer();

        while ((inputLine = in.readLine()) != null) { response.append(inputLine); }
        in.close();

        queryResult = response.toString();
    } finally { connection.disconnect(); }
}
```