Command line REST Application

Scott Hurrey

Tutorial - Build a Java Command Line REST Application

The purpose of this document is to allow a developer to walk through the REST Workshop from DevCon 2016. Please note that this sample code was built and tested on Mac. It is Java code, and therefore, should be operating system agnostic, but it has not been tested on Windows.

In order to begin this tutorial, there are a few pre-requisites that are assumed to be in place:

- User has installed and started the latest Developer AMI.
- User has installed and configured GIT.
- User has cloned the repository to their local working directory.
- User has Java 11 installed and configured to JAVA_HOME.
- User has registered for an account on the Developer Portal, registered an application, obtained the key and secret, and configured the Developer Virtual Machine to accept it..

To watch a recording of the corresponding Webinar, click here.

About the Project

This project is built on Java 11, in Spring Tool Suite (an Eclipse derivative), and uses Gradle and Maven to build and install dependencies. In addition, Gradle is using the 'Application' plugin, which allows us to build and run the command line java application with one simple Gradle command – gradle run.

Architecturally, this sample relies on the RestTemplate object offered by the Spring Framework for all REST communication. The method or library you choose for your application is very much a personal preference. Internally at Anthology, this is the preferred method, and therefore, the method chosen for this workshop.

The project contains a base 'Hello Developers' application that contains all of the supporting code to get started. The code can all be found in the src/main/java directory in two base packages, bbdn.unsecurity and bbdn.rest. There are also four top-level folders containing the code we will need to implement each step of the tutorial. Let's break down the base files:

File	Package	Description
RestConstants.java	bbdn.rest	This class contains Constant values for running the code.
RestDemo.java	bbdn.rest	This class is the main class for the project. We will discuss other classes as we move them into place, but this is the only file we will modify.
UnSecurityUtil.java	bbdn.unsecurity	This class allows us to create a RestTemplate object based on whether we are testing on the DVM or in an environment with a valid SSL certificate.

RestConstants

The RestConstants.java file takes the Data I/O requirements out of the sample code, allowing the code to focus on the REST API interactions. In a real-world application, this data would be taken from configuration files or tables and UI forms and user input.

```
package bbdn.rest;
```

```java
public abstract class RestConstants {
  public final static String HOSTNAME = "http://localhost:9876";
  public final static String KEY = "";
  public final static String SECRET = "";
  public final static String AUTH_PATH = "/learn/api/public/v1/oauth2/token";
  public final static String DATASOURCE_PATH = "/learn/api/public/v1/dataSources";
  public final static String DATASOURCE_ID = "BBDN-DSK-JAVA";
  public final static String DATASOURCE_DESCRIPTION = "Demo Data Source used for REST Java
  ↪  Demo";
  public final static String TERM_PATH = "/learn/api/public/v1/terms";
  public final static String TERM_ID = "BBDN-TERM-JAVA";
  public final static String TERM_NAME = "REST Demo Term - Java";
  public final static String TERM_RAW = "Term Used For REST Demo - Java";
  public final static String TERM_DISPLAY = "Term Used For REST Demo - Java";
  public final static String COURSE_PATH = "/learn/api/public/v1/courses";
  public final static String COURSE_ID = "BBDN-Java-REST-Demo";
  public final static String COURSE_NAME = "Course Used For REST Demo - Java";
  public final static String COURSE_DESCRIPTION = "Course Used For REST Demo - Java";
  public final static String USER_PATH = "/learn/api/public/v1/users";
  public final static String USER_ID = "bbdnrestdemojavauser";
  public final static String USER_NAME = "restjavauser";
  public final static String USER_PASS = "Bl@ckb0ard!";
  public final static String USER_FIRST = "Java";
  public final static String USER_LAST = "Restdemo";
  public final static String USER_EMAIL = "developers@blackboard.com";
}
```

RestDemo

This is the heart of the application. This contains the main processing loop that performs all of the functions that we want to perform. We will review the other classes as we progress through the tutorial, but the code we will write lives in this file.

```java
package bbdn.rest;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

public class RestDemo {

  private static final Logger log = LoggerFactory.getLogger(RestDemo.class);
  public static boolean DEVMODE = false;

  public static void main(String[] args) {
    boolean OPER_ALL = false;
    boolean OPER_DATASOURCE = false;
    boolean OPER_TERM = false;
    boolean OPER_COURSE = false;
    boolean OPER_USER = false;
    boolean OPER_MEMBERSHIP = false;
    boolean OPER_CONTENT = false;
    boolean OPER_GRADES = false;

    String _hostname = RestConstants.HOSTNAME;
    int operCount = 0;
    boolean nextCommand = false;
```

```java
      int uriCount = 0;
      boolean nextUri = false;

      if(args.length > 0) {

      for(int i = 0; i < args.length; i++ ) {
        log.info("args[" + i + "]: " + args[i]);

        if (nextCommand) {
          switch(args[i]) {
            case "datasource":
              OPER_DATASOURCE = true;
              break;
            case "term":
              OPER_TERM = true;
              break;
            case "course":
              OPER_COURSE = true;
              break;
            case "user":
              OPER_USER = true;
              break;
            case "membership":
              OPER_MEMBERSHIP = true;
              break;
            case "content":
              OPER_CONTENT = true;
              break;
            case "grades":
              OPER_GRADES = true;
              break;
            case "all":
            default:
              OPER_ALL = true;
          }
          nextCommand = false;
        }
        if(nextUri) {
          _hostname = args[i];
          nextUri = false;
        }

        if (args[i].equalsIgnoreCase("-c")) {
          nextCommand = true;
          operCount += 1;
        }
        else if (args[i].equalsIgnoreCase("-t")) {
          nextUri = true;
          uriCount += 1;
        }
        else if (args[i].equalsIgnoreCase("-d")) {
          DEVMODE = true;
        }
      }
      if(operCount == 0) {
```

```java
        OPER_ALL = true;
      }
      log.info(" OPER_ALL: " + OPER_ALL + " OPER_DATASOURCE: " + OPER_DATASOURCE + " OPER_TERM:
↪    " + OPER_TERM + " OPER_COURSE: " + OPER_COURSE + " OPER_USER: " + OPER_USER + "
↪    OPER_MEMBERSHIP: " + OPER_MEMBERSHIP + " OPER_CONTENT: " + OPER_CONTENT + " OPER_GRADES: "
↪    + OPER_GRADES + " HOSTNAME: " + _hostname + " operCount: " + operCount + "uriCount: " +
↪    uriCount);
    }
    else {
      OPER_ALL = true;
    }
    log.info("Hello REST Workshop Participant!!!");
  }
}
```

UnSecurityUtil

The unSecurityUtil class allows us to bypass SSL Certificate checking when the -d flag is passed at the command line. This is useful in test environments and specifically on the Developer Virtual Machine where the SSL certificate is self-signed and therefore untrusted by Java.

```java
package bbdn.unsecurity;

import java.security.KeyManagementException;
import java.security.KeyStoreException;
import java.security.NoSuchAlgorithmException;
import java.security.cert.X509Certificate;
import javax.net.ssl.SSLContext;
import org.apache.http.conn.ssl.SSLConnectionSocketFactory;
import org.apache.http.conn.ssl.TrustStrategy;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.springframework.http.HttpStatus;
import org.springframework.http.client.HttpComponentsClientHttpRequestFactory;
import org.springframework.web.client.DefaultResponseErrorHandler;
import org.springframework.web.client.RestTemplate;
import bbdn.rest.RestDemo;

public abstract class UnSecurityUtil {
  public static RestTemplate getRestTemplate() throws KeyManagementException,
  ↪  NoSuchAlgorithmException, KeyStoreException {
    // Workaround to allow for PATCH requests
    HttpComponentsClientHttpRequestFactory requestFactory = new
  ↪  HttpComponentsClientHttpRequestFactory();

    if(RestDemo.DEVMODE) {

      TrustStrategy acceptingTrustStrategy = (X509Certificate[] chain, String authType) ->
  ↪  true;

      SSLContext sslContext = org.apache.http.ssl.SSLContexts.custom()
        .loadTrustMaterial(null, acceptingTrustStrategy)
        .build();

      SSLConnectionSocketFactory csf = new SSLConnectionSocketFactory(sslContext);
```

```
    CloseableHttpClient httpClient = HttpClients.custom()
      .setSSLSocketFactory(csf)
      .build();

    requestFactory.setHttpClient(httpClient);
  }

  RestTemplate restTemplate = new RestTemplate(requestFactory);

  // Workaround for allowing unsuccessful HTTP Errors to still print to the screen
  restTemplate.setErrorHandler(new DefaultResponseErrorHandler(){
    protected boolean hasError(HttpStatus statusCode) {
      return false;
    }});

  return(restTemplate);
  }
}
```

Usage

The first step is to modify the RestConstants.java file to include your shared key and secret, and, if not using the Developer Virtual Machine, modify hostname to your primary test environment. Then simply open the command line, navigate to the directory containing the project, and type `gradle run`. If you do not have Gradle installed, you can use `./gradlew run` instead. This will install gradle in your project directory.

In addition, the experience can by customized at the command line by specifying command line arguments to be passed to the java application. The syntax for this is `gradle run -Dexec.args=""`. The arguments would appear just as they would if typing them directly to the java executable, space- delimited. For example, to run in Devmode, you would type `gradle run -Dexec.args="-d"`. The following table lists the available command line arguments. They can be combined.

Arg	Description	Example
-t	Specify the target URL. This will override the HOSTNAME constant specified in RestConstants.	`gradle run -Dexec.args="-t https://localhost:9877`
-d	Turn on DevMode. DevMode tells the sample application to ignore certificate errors. Useful when testing on the DVM or against a test/dev server with a self-signed certificate.	`gradle run -Dexec.args="-d"`
-c	Denote a specific object to exercise. Added for example, but not well implemented as each object relies on the previous object to be created and available.	`gradle run -Dexec.args="-c datasource"`

Available commands are:

- datasource
- term
- course
- user
- membership
- content *Not Yet Implemented
- grades *Not Yet Implemented
- all

Step 0: Run Hello Developer

The first step is to simply run the application. You should have already cloned the project to a local directory. Simply open a command line window, navigate to the directory you cloned to and type `gradle run`.

What You Should See   Command line:

```
shurreymbp:BBDN-DevCon-REST-Workshop shurrey$ gradle run
:compileJava
:processResources UP-TO-DATE
:classes
:run
[main] INFO bbdn.rest.RestDemo - Hello REST Workshop Participant!!!

BUILD SUCCESSFUL

Total time: 2.805 secs
```

Figure 1: Hello World

What We Did   In this step, we simply ran the application to ensure that we had all of our libraries set, that our code ran properly, and that we have gradle installed for our project.

Step 1: Authorization

The key to any REST client application, and generally the biggest hurdle to adoption, is Authorization. There are a number of flavors out there, and the implementation chosen is often both a personal choice by the developer that built the REST API and also a function of the specific requirements and infrastructure around it. For more information regarding Learn's implementation of OAuth2, read the Basic Authentication in this section.

In the "1-Implement-OAuth" folder, you will find three files:

- Authorizer.java
  - Authorizer is the service handler class. This does the actual work of requesting the token and retrieving the results.
- Token.java
  - Token is a JSON Model class. This is what we will use to interact with the token we receive from Learn. Authorizer will request the token, and on receipt, will automatically serialize the JSON response into the Token object, allowing the application to simply interact with it as a normal Java object.
- RestDemo.java
  - The RestDemo class is what our bbdn.rest.RestDemo class will look like when we complete this step in the tutorial.

Let's get started:

1. Move `/1-Implement-OAuth/Authorizer.java` to the `bbdn.rest` package.
2. Create a new package called `bbdn.rest.objects`.
3. Move `/1-Implement-OAuth/Token.java` to the `bbdn.rest.objects` package.
4. If you want a shortcut, you can copy the `/1-Implement-OAuth/RestDemo.java` file to the `bbdn.rest` package, otherwise open `bbdn.rest.RestDemo.java` and make the following edits:
    1. Add an import to the top of your RestDemo to include `bbdn.rest.objects` space:

    ```
    import bbdn.rest.objects.*;
    ```

2. Now highlight the log statement that prints our hello developer message and replace it with:

    ```
    // Obtain Bearer Token
    Authorizer auth = new Authorizer(); // Instantiate Authorizer class
    ```

6

```
        Token token = auth.authorize(); // Authorize application and grab the Token object.
        log.debug("token: " + token.toString());
```

5. Save the file.
6. At the command line in the project directory, type `gradle run` and watch as the token is printed to the screen.

Review the RestDemo.java in the `1-Implement-OAuth` folder to compare to your RestDemo.java if you have issues.

What You Should See   command line:

What We Did   So in this step of the tutorial, we took our basic "Hello Developer" application and built out the authorization service and object we need for the rest of our application to work properly.

We first put our REST Authorization service in place. So with Basic Authentication and OAuth, there is really a few specific things that we need to do. First, we must hash our key and secret into a base64-encoded string. Once we have the utility in place to generate that hash, the first thing to do is build the header. In this application, we are using HttpHeaders. We add the Authorization Header first, in the form of Authorization: Basic . We then set the content type to form-urlencoded using the HttpHeaders method setContentType. The last thing we need to do is define the applications scope in the form of grant_type. In Learn's case, we only accept client_credentials, so this part is easy. Simply add grant_type=client_credentials in the body of our POST message and POST it to the oauth/token endpoint.

The next step was to put our Token object in place. This object is configured to allow the Jackson library to automatically serialize and deserialize our JSON payloads. It provides three basic properties that correspond directly to the JSON payload we expect to receive from Learn when we request a token: access_token, token_type, and expires_in. Once we have created and populated this object, we can then access this data whenever we need to.

Lastly, we updated our main class in order to call the authorize method and retrieve the token.

Step 2: Datasources

The first – and easiest – object to interact with is the Datasource object. It is a rudimentary object with only a few properties that need to be manipulated. It is also a very important object, as most other objects allow you to specify a datasource, which allows you to associate all of the data you create together and to your application.

In the "2-Implement-Datasource" folder, you will find three files:

- DatasourceHandler.java
  - DatasourceHandler is the service handler class. This does the actual work of building and processing HTTP requests and handling the incoming responses.
- Datasource.java
  - Datasource is a JSON Model class. This is what we will use to create datasources and to retrieve datasources from the REST responses we receive from Learn. DatasourceHandler will automatically handle the serialization and deserialization of the datasource object, allowing the application to simply interact with it as a normal Java object.
- RestDemo.java.
  - The RestDemo class is what our bbdn.rest.RestDemo class will look like when we complete this step in the tutorial.

Let's get started:

1. Move /2-Implement-Datasource/Datasource.java to the bbdn.rest.objects package.
2. Create a new package called bbdn.rest.services.
3. Move /2-Implement-Datasource/DatasourceHandler.java to the bbdn.rest.services package.
4. If you want a shortcut, you can copy the /2-Implement-Datasource/RestDemo.java file to the bbdn.rest package, otherwise open bbdn.rest.RestDemo.java and make the following edits:
   1. Add an import to the top of your RestDemo to include the bbdn.rest.services space:

   ```java
   import bbdn.rest.services.*;
   ```

2. Directly above the Authorizer instantiation, add the following code:

   ```java
   // Declare Datasource
   Datasource ds = null;
   // Declare Result to receive Delete response
   boolean result = false;
   // Initialize Handlers
   DatasourceHandler datasourceHandler = new DatasourceHandler(_hostname);
   ```

3. Directly below the log statement that writes the token information to the screen, add the following code:

   ```java
   // Datasource object
   if( OPER_ALL || OPER_DATASOURCE) {
     ds = datasourceHandler.createObject(token.getToken());
     log.debug("Create DS: " + ds.toString());
     ds = datasourceHandler.readObject(token.getToken());
     log.debug("Read DS: " + ds.toString());
     ds = datasourceHandler.updateObject(token.getToken());
     log.debug("Update DS: " + ds.toString());
   }
   if( OPER_ALL || OPER_DATASOURCE) {
     result = datasourceHandler.deleteObject(token.getToken());
     log.debug("Delete DS: " + result);
   }
   ```

5. Save the file.
6. At the command line in the project directory, type gradle run (or ./gradlew run) and watch as the token is printed to the screen.

Review the RestDemo.java in the 2-Implement-Datasource folder to compare to your RestDemo.java if you have issues.

What You Should See   For Read:

For Update:

For Delete:

```
[main] INFO bbdn.rest.services.DatasourceHandler - Datasource: Datasource [id=_11_1, externalId=BBDN-DSK-JAVA, description=Demo Data Source used for REST Java Demo, statu
s=null, code=null, message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.DatasourceHandler - DELETE
[main] INFO bbdn.rest.services.DatasourceHandler - URI is http://localhost:9876/learn/api/public/v1/dataSources/externalId:BBDN-DSK-JAVA
[main] INFO bbdn.rest.services.DatasourceHandler - Request Headers: {Authorization=[Bearer Mwqv7e3s7xFHTDNqbbMkfkvu3AH4Mrbk], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.DatasourceHandler - Request Body: Datasource [id=null, externalId=null, description=null, status=null, code=null, message=null, developerMe
ssage=null, extrainfo=null]
[main] INFO bbdn.rest.services.DatasourceHandler - Response: <204 NO Content,{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"], X-Blackboard-appserver=[dev.bbdn.local],
X-Blackboard-product=[Blackboard Learn &#8482; 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store, must-revalidate], Last-Modified=
[Sat, 27 Jul 1996 17:46:51 GMT], Expires=[Mon, 27 Jul 2015 17:46:51 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors 'self'], Set-Cookie=[JSES
SIONID=441B19A2EA9D52785ABC717F7D4FAE74; Path=/learn/api], X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab], Content-Type=[application/json;charset=UTF-8], Date=[Wed
, 27 Jul 2016 17:46:51 GMT]]>
```

What We Did   So in this step of the tutorial, we added the code necessary to create, read, update, and delete a datasource.

The first step was to put our Datasource object in place. This object is configured to allow the Jackson library to automatically serialize and deserialize our JSON payloads. It provides three basic properties that correspond directly to the JSON payload we expect to receive from Learn when we manipulate a datasource: id, name, and description. Once we have created and populated this object, we can then access this data whenever we need to.

Next we implemented our DatasourceHandler service class. This class handles the REST calls, all object generation and retrieval, and passing information back to our main class. When the class is initialized, we set up logging and make sure the handler has the correct hostname. Each operation has its own method that generates the appropriate body if necessary, then passes the necessary data to the sendRequest method to send the appropriate message and retrieve the appropriate response.

Lastly, we updated our main class in order to call the appropriate methods to interact with the Datasource Service class.

Step 3: Terms

For the third step in our tutorial we will implement the Term object. As with Datasources, Terms are also complimentary objects to other operations. At the conclusion on this tutorial, we will be setting this Term when we implement both courses and memberships. It is also the first in a more complex set of Model objects in that the base Term object requires a secondary Availability object to properly create and manipulate a Term.

In the "3-Implement-Term" folder, you will find four files:

- TermHandler.java
  - TermHandler is the service handler class. This does the actual work of building and processing HTTP requests and handling the incoming responses.
- Term.java
  - Term is a JSON Model class. This is what we will use to create terms and to retrieve terms from the REST responses we receive from Learn. TermHandler will automatically handle the serialization and deserialization of the term object, allowing the application to simply interact with it as a normal Java object.
- Availability.java
  - Availability is also a model class. This class contains one field, "available," which will be set to yes or now. This class is only called from within the Term class, and this is what allows Jackson to handle the serialization and deserialization of complex nested objects.
- RestDemo.java
  - The RestDemo class is what our bbdn.rest.RestDemo class will look like when we complete this step in the tutorial.

Let's get started:

1. Move /3-Implement-Term/TermHandler.java to the bbdn.rest.services package.
2. Move /3-Implement-Term/Term.java to the bbdn.rest.objects package.
3. Move /3-Implement-Term/Availability.java to the bbdn.rest.objects package.
4. If you want a shortcut, you can copy the /3-Implement-Term/RestDemo.java file to the bbdn.rest package, otherwise open bbdn.rest.RestDemo.java and make the following edits:
   1. Directly below *Datasource ds = null;*, add:

   ```
   Term tm = null;
   ```

2. Directly below *DatasourceHandler datasourceHandler = new DatasourceHandler(_hostname);*, add:

```java
    TermHandler termHandler = new TermHandler(_hostname);
```

3. Between the closing curly bracket for the first *if(OPER_ALL || OPER_DATASOURCE) {* and the start of the second, add:

```java
// Term object
if( OPER_ALL || OPER_TERM) {
  tm = termHandler.createObject(token.getToken(), ds.getId());
  log.debug("Create TM: " + tm.toString());
  tm = termHandler.readObject(token.getToken());
  log.debug("Read TM: " + tm.toString());
  tm = termHandler.updateObject(token.getToken(), ds.getId());
  log.debug("Update TM: " + tm.toString());
}

// Delete objects
if( OPER_ALL || OPER_TERM) {
  result = termHandler.deleteObject(token.getToken());
  log.debug("Delete TM: " + result);
}
```

5. Save the file.
6. At the command line in the project directory, type gradle run (or ./gradlew run) and watch as the token is printed to the screen.

Review the RestDemo.java in the 3-Implementing-Term folder to compare to your RestDemo.java if you have issues.

What You Should See   For Create:

For Read:

For Update:

For Delete:

What We Did   So in this step of the tutorial, we added the code necessary to create, read, update, and delete a term.

The first step was to put our Term object in place, along with its companion Availability object. This object is actually re-used by other classes we will implement in the final step. These objects are configured to allow the Jackson library to automatically serialize and deserialize our JSON payloads. It provides the basic properties that correspond directly to the JSON payload we expect to receive from Learn when we manipulate a term. Once we have created and populated this object, we can then access this data whenever we need to.

Next we implemented our TermHandler service class. This class handles the REST calls, all object generation and retrieval, and passing information back to our main class. When the class is initialized, we set up logging and make sure the handler has the correct hostname. Each operation has its own method that generates the appropriate body if necessary, then passes the necessary data to the sendRequest method to send the appropriate message and retrieve the appropriate response.

Lastly, we updated our main class in order to call the appropriate methods to interact with the Term Service class.

Step 4: Remaining Objects

The remainder of the tutorial is really just about moving the remaining objects and handlers in place and repeating the previously established pattern in the RestDemo class.

In the "4-Implement-Remaining-Objects" folder, you will find a plethora of files. You will also notice a familiar pattern:

- Handler classes are the service handler classes for their respective objects. As before, these do the actual work of manipulating objects and retrieving the results.

- Classes named after objects are JSON Model classes. These are what we will use to interact with the JSON we receive from Learn. The handler classes will handle the REST API interaction, and automatically serialize and deserialize the JSON payloads into the corresponding object, allowing the application to simply interact with it as a normal Java object.

- Several of these model classes are nested objects for other base classes.

- The RestDemo class is what our bbdn.rest.RestDemo class will look like when we complete this step in the tutorial.

Let's get started:

1. Move /4-Implement-Remaining-Objects/CourseHandler.java to the bbdn.rest.services package.
2. Move /4-Implement-Remaining-Objects/UserHandler.java to the bbdn.rest.services package.
3. Move /4-Implement-Remaining-Objects/MembershipHandler.java to the bbdn.rest.services package.
4. Move /4-Implement-Remaining-Objects/Course.java to the bbdn.rest.objects package.
5. Move /4-Implement-Remaining-Objects/User.java to the bbdn.rest.objects package.
6. Move /4-Implement-Remaining-Objects/Membership.java to the bbdn.rest.objects package.
7. Move /4-Implement-Remaining-Objects/Contact.java to the bbdn.rest.objects package.
8. Move /4-Implement-Remaining-Objects/Name.java to the bbdn.rest.objects package.
9. Move /4-Implement-Remaining-Objects/RestDemo.java to the bbdn.rest package.
10. At the command line in the project directory, type gradle run (or ./gradlew run) and watch as the token is printed to the screen.

Review the RestDemo.java in the 4-Implement-Remaining-Objects folder to compare to your RestDemo.java if you have issues.

What You Should See   Command line:

```
shurreymbp:BBDN-DevCon-REST-Workshop shurrey$ cp 4-Implement-Remaining-Objects/RestDemo.java
 ↪  src/main/java/bbdn/rest/
shurreymbp:BBDN-DevCon-REST-Workshop shurrey$ gradle run
:compileJava
```

:processResources UP-TO-DATE
:classes
:run
[main] INFO bbdn.rest.Authorizer – Value to hash:
↪ d03caa33-1095-47b9-bc67-f5cd634430b1:QSFClAMu5KmoG8yFbHTi7pjhsseJl4uz
[main] INFO bbdn.rest.Authorizer – Hashed Value:
↪ ZDAzY2FhMzMtMTA5NS00N2I5LWJjNjctZjVjZDYzNDQzMGIxOlFTRkNsQU11NUttb0c4eUZiSFRpN3BqaHNzZUpsNHV6
[main] INFO bbdn.rest.Authorizer – Request Headers: {Authorization=[Basic
↪ ZDAzY2FhMzMtMTA5NS00N2I5LWJjNjctZjVjZDYzNDQzMGIxOlFTRkNsQU11NUttb0c4eUZiSFRpN3BqaHNzZUpsNHV6],
↪ Content-Type=[application/x-www-form-urlencoded]}
[main] INFO bbdn.rest.Authorizer – Request Body:grant_type=client_credentials
[main] INFO bbdn.rest.Authorizer – Response: <200
↪ OK,Token{access_token=uC5G2HvEsI1DWP24xD2lnaG35EQsenZs', token_type=bearer',
↪ expires_in=1639},{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪ X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn ™
↪ 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪ must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:05 GMT], Expires=[Mon, 27 Jul 2015
↪ 19:50:05 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪ 'self'], Set-Cookie=[JSESSIONID=5457EE8C14E1C940EE662CE047A3DF31;
↪ Path=/learn/api],X-Blackboard-Context-Version=[3000.1.1-rel.5+7a1d308],
↪ Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed,27
↪ Jul 2016 19:50:06 GMT]}>
[main] INFO bbdn.rest.Authorizer – Access Token:
↪ Token{access_token=uC5G2HvEsI1DWP24xD2lnaG35EQsenZs', token_type=bearer',
expires_in=1639}
[main] INFO bbdn.rest.services.DatasourceHandler – CREATE
[main] INFO bbdn.rest.services.DatasourceHandler – URI is
↪ http://localhost:9876/learn/api/public/v1/dataSources
[main] INFO bbdn.rest.services.DatasourceHandler – Request Headers:{Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.DatasourceHandler – Request Body: Datasource [id=null,
↪ externalId=BBDN-DSK-JAVA, description=Demo Data Source used for REST Java Demo,
↪ status=null, code=null, message=null, developerMessage=null,extrainfo=null]
[main] INFO bbdn.rest.services.DatasourceHandler – Response: <201 Created,Datasource [id=_13_1,
↪ externalId=BBDN-DSK-JAVA, description=Demo Data Source used for REST Java Demo,
↪ status=null, code=null, message=null, developerMessage=null,
↪ extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪ X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn ™
↪ 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪ must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:06 GMT], Expires=[Mon, 27 Jul 2015
↪ 19:50:06 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪ 'self'], Set-Cookie=[JSESSIONID=0BEAD73C4DFFC8C01B037883D467B533; Path=/learn/api],
↪ Location=[/learn/api/public/v1/dataSources/_13_1],X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab
↪ Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed,27
↪ Jul 2016 19:50:06 GMT]}>
[main] INFO bbdn.rest.services.DatasourceHandler – Datasource: Datasource [id=_13_1,
↪ externalId=BBDN-DSK-JAVA, description=Demo Data Source used for REST Java Demo,
↪ status=null, code=null, message=null, developerMessage=null,extrainfo=null]
[main] INFO bbdn.rest.services.DatasourceHandler – READ
[main] INFO bbdn.rest.services.DatasourceHandler – URI is
↪ http://localhost:9876/learn/api/public/v1/dataSources/externalId:BBDN-DSK-JAVA
[main] INFO bbdn.rest.services.DatasourceHandler – Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}

```
[main] INFO bbdn.rest.services.DatasourceHandler - Request Body: Datasource [id=null,
↪ externalId=null, description=null, status=null, code=null,message=null,
↪ developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.DatasourceHandler - Response: <200 OK,Datasource [id=_13_1,
↪ externalId=BBDN-DSK-JAVA, description=Demo Data Source used for REST Java Demo,
↪ status=null, code=null, message=null, developerMessage=null,
↪ extrainfo=null],{Server=[Apache-Coyote/1.1],P3P=[CP="CAO PSA OUR"],
↪ X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪ 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache],Cache-Control=[no-cache, max-age=0, no-store,
↪ must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:06 GMT], Expires=[Mon, 27 Jul 2015
↪ 19:50:06 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪ 'self'], Set-Cookie=[JSESSIONID=5DDDADB64296A3E350B157B6D44B7B7F;Path=/learn/api],
↪ X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],Content-Type=[application/json;charset=UTF-8],
↪ Transfer-Encoding=[chunked],Date=[Wed, 27 Jul 2016 19:50:06 GMT]}>
[main] INFO bbdn.rest.services.DatasourceHandler - Datasource: Datasource [id=_13_1,
↪ externalId=BBDN-DSK-JAVA, description=Demo Data Source used for REST Java Demo,
↪ status=null, code=null, message=null, developerMessage=null,
extrainfo=null]
[main] INFO bbdn.rest.services.DatasourceHandler - UPDATE
[main] INFO bbdn.rest.services.DatasourceHandler - URI is
↪ http://localhost:9876/learn/api/public/v1/dataSources/externalId:BBDN-DSK-JAVA
[main] INFO bbdn.rest.services.DatasourceHandler - Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.DatasourceHandler - Request Body: Datasource [id=null,
↪ externalId=BBDN-DSK-JAVA, description=Demo Data Source used for REST Java Demo,
↪ status=null, code=null, message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.DatasourceHandler - Response: <200 OK,Datasource [id=_13_1,
↪ externalId=BBDN-DSK-JAVA, description=Demo Data Source used for REST Java Demo,
↪ status=null, code=null, message=null, developerMessage=null,
↪ extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪ X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪ 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪ must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:06 GMT], Expires=[Mon, 27 Jul 2015
↪ 19:50:06 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪ 'self'], Set-Cookie=[JSESSIONID=F8464644EAA806822A7121A3AAEE0531;Path=/learn/api],
↪ X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],Content-Type=[application/json;charset=UTF-8],
↪ Transfer-Encoding=[chunked],Date=[Wed, 27 Jul 2016 19:50:06 GMT]}>
[main] INFO bbdn.rest.services.DatasourceHandler - Datasource: Datasource [id=_13_1,
↪ externalId=BBDN-DSK-JAVA, description=Demo Data Source used for REST Java Demo,
↪ status=null, code=null, message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.TermHandler - CREATE
[main] INFO bbdn.rest.services.TermHandler - URI is
↪ http://localhost:9876/learn/api/public/v1/terms
[main] INFO bbdn.rest.services.TermHandler - Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.TermHandler - Request Body: Term [id=null,
↪ externalId=BBDN-TERM-JAVA, dataSourceId=_13_1, name=REST Demo Term - Java, description=Term
↪ Used For REST Demo - Java, availability=Availability [available=Yes], status=null,
↪ code=null, message=null, developerMessage=null, extrainfo=null]
```

```
[main] INFO bbdn.rest.services.TermHandler - Response: <201 Created,Term [id=_7_1,
↪ externalId=BBDN-TERM-JAVA, dataSourceId=_13_1, name=REST Demo Term - Java, description=Term
↪ Used For REST Demo - Java, availability=Availability [available=Yes], status=null,
↪ code=null, message=null, developerMessage=null,
↪ extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪ X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪ 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪ must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:07 GMT], Expires=[Mon, 27 Jul 2015
↪ 19:50:07 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪ 'self'], Set-Cookie=[JSESSIONID=2498E5AC7B7F40B9B4A53BFCC3762029; Path=/learn/api],
↪ Location=[/learn/api/public/v1/terms/_7_1],
↪ X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪ Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed,27
↪ Jul 2016 19:50:06 GMT]}>
[main] INFO bbdn.rest.services.TermHandler - Term: Term [id=_7_1,externalId=BBDN-TERM-JAVA,
↪ dataSourceId=_13_1, name=REST Demo Term - Java, description=Term Used For REST Demo - Java,
↪ availability=Availability [available=Yes], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.TermHandler - READ
[main] INFO bbdn.rest.services.TermHandler - URI is
↪ [http://localhost:9876/learn/api/public/v1/terms/externalId:BBDN-TERM-JAVA
[main] INFO bbdn.rest.services.TermHandler - Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.TermHandler - Request Body: Term [id=null, externalId=null,
↪ dataSourceId=null, name=null, description=null, availability=null, status=null, code=null,
↪ message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.TermHandler - Response: <200 OK,Term [id=_7_1,
↪ externalId=BBDN-TERM-JAVA, dataSourceId=_13_1, name=REST Demo Term - Java, description=Term
↪ Used For REST Demo - Java, availability=Availability [available=Yes], status=null,
↪ code=null, message=null, developerMessage=null,
↪ extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪ X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪ 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪ must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:07 GMT], Expires=[Mon, 27 Jul 2015
↪ 19:50:07 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪ 'self'], Set-Cookie=[JSESSIONID=4341909E1D54C7A129E5883CE21E82E0; Path=/learn/api],
↪ X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪ Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed, 27
↪ Jul 2016 19:50:06 GMT]}>
[main] INFO bbdn.rest.services.TermHandler - Term: Term [id=_7_1, externalId=BBDN-TERM-JAVA,
↪ dataSourceId=_13_1, name=REST Demo Term - Java, description=Term Used For REST Demo - Java,
↪ availability=Availability [available=Yes], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.TermHandler - UPDATE
[main] INFO bbdn.rest.services.TermHandler - URI is
↪ http://localhost:9876/learn/api/public/v1/terms/externalId:BBDN-TERM-JAVA
[main] INFO bbdn.rest.services.TermHandler - Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.TermHandler - Request Body: Term [id=null,
↪ externalId=BBDN-TERM-JAVA, dataSourceId=_13_1, name=REST Demo Term - Java, description=Term
↪ Used For REST Demo - Java, availability=Availability [available=Yes], status=null,
↪ code=null, message=null, developerMessage=null, extrainfo=null]
```

```
[main] INFO bbdn.rest.services.TermHandler - Response: <200 OK,Term [id=_7_1,
↪  externalId=BBDN-TERM-JAVA, dataSourceId=_13_1, name=REST Demo Term - Java, description=Term
↪  Used For REST Demo - Java, availability=Availability [available=Yes], status=null,
↪  code=null, message=null, developerMessage=null,
↪  extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪  X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪  3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪  must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:07 GMT], Expires=[Mon, 27 Jul 2015
↪  19:50:07 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪  'self'], Set-Cookie=[JSESSIONID=097D9AF4FF4AAE88C29BD42C9AA0526B; Path=/learn/api],
↪  X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪  Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed, 27
↪  Jul 2016 19:50:07 GMT]}>
[main] INFO bbdn.rest.services.TermHandler - Term: Term [id=_7_1, externalId=BBDN-TERM-JAVA,
↪  dataSourceId=_13_1, name=REST Demo Term - Java, description=Term Used For REST Demo - Java,
↪  availability=Availability [available=Yes], status=null, code=null, message=null,
↪  developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.CourseHandler - CREATE
[main] INFO bbdn.rest.services.CourseHandler - Course [id=null, uuid=null,
↪  externalId=BBDN-Java-REST-Demo, dataSourceId=_13_1, courseId=BBDN-Java-REST-Demo,
↪  name=Course Used For REST Demo - Java, description=Course Used For REST Demo - Java,
↪  created=null, organization=false, ultraStatus=null, allowGuests=true, readOnly=false,
↪  termId=_7_1, availability=Availability [available=Yes], status=null, code=null,
↪  message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.CourseHandler - URI is
↪  http://localhost:9876/learn/api/public/v1/courses
[main] INFO bbdn.rest.services.CourseHandler - Request Headers: {Authorization=[Bearer
↪  uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.CourseHandler - Request Body: Course [id=null, uuid=null,
↪  externalId=BBDN-Java-REST-Demo, dataSourceId=_13_1, courseId=BBDN-Java-REST-Demo,
↪  name=Course Used For REST Demo - Java, description=Course Used For REST Demo - Java,
↪  created=null, organization=false, ultraStatus=null, allowGuests=true, readOnly=false,
↪  termId=_7_1, availability=Availability [available=Yes], status=null, code=null,
↪  message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.CourseHandler - Response: <201 Created,Course [id=_7_1,
↪  uuid=9ec59f4d3e094cd5b2025858a28a3b39, externalId=BBDN-Java-REST-Demo, dataSourceId=_13_1,
↪  courseId=BBDN-Java-REST-Demo, name=Course Used For REST Demo - Java, description=Course
↪  Used For REST Demo - Java, created=2016-07-27T19:50:07.376Z, organization=false,
↪  ultraStatus=Classic, allowGuests=true, readOnly=false, termId=_7_1,
↪  availability=Availability [available=Yes], status=null, code=null, message=null,
↪  developerMessage=null, extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪  X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪  3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪  must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:07 GMT], Expires=[Mon, 27 Jul 2015
↪  19:50:07 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪  'self'], Set-Cookie=[JSESSIONID=54A6544B4CDB9B039EFA56487A85035B; Path=/learn/api],
↪  Location=[/learn/api/public/v1/courses/_7_1],
↪  X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪  Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed, 27
↪  Jul 2016 19:50:07 GMT]}>
```

```
[main] INFO bbdn.rest.services.CourseHandler - Course: Course [id=_7_1,
↪ uuid=9ec59f4d3e094cd5b2025858a28a3b39, externalId=BBDN-Java-REST-Demo, dataSourceId=_13_1,
↪ courseId=BBDN-Java-REST-Demo, name=Course Used For REST Demo - Java, description=Course
↪ Used For REST Demo - Java, created=2016-07-27T19:50:07.376Z, organization=false,
↪ ultraStatus=Classic, allowGuests=true, readOnly=false, termId=_7_1,
↪ availability=Availability [available=Yes], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.CourseHandler - READ
[main] INFO bbdn.rest.services.CourseHandler - URI is
↪ http://localhost:9876/learn/api/public/v1/courses/externalId:BBDN-Java-REST-Demo
[main] INFO bbdn.rest.services.CourseHandler - Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.CourseHandler - Request Body: Course [id=null, uuid=null,
↪ externalId=null, dataSourceId=null, courseId=null, name=null, description=null,
↪ created=null, organization=false, ultraStatus=null, allowGuests=false, readOnly=false,
↪ termId=null, availability=null, status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.CourseHandler - Response: <200 OK,Course [id=_7_1,
↪ uuid=9ec59f4d3e094cd5b2025858a28a3b39, externalId=BBDN-Java-REST-Demo, dataSourceId=_13_1,
↪ courseId=BBDN-Java-REST-Demo, name=Course Used For REST Demo - Java, description=Course
↪ Used For REST Demo - Java, created=2016-07-27T19:50:07.376Z, organization=false,
↪ ultraStatus=Classic, allowGuests=true, readOnly=false, termId=_7_1,
↪ availability=Availability [available=Yes], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪ X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪ 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪ must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:07 GMT], Expires=[Mon, 27 Jul 2015
↪ 19:50:07 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪ 'self'], Set-Cookie=[JSESSIONID=DF5C615760A8438DFF112613D84926EF; Path=/learn/api],
↪ X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪ Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed, 27
↪ Jul 2016 19:50:07 GMT]}>
[main] INFO bbdn.rest.services.CourseHandler - Course: Course [id=_7_1,
↪ uuid=9ec59f4d3e094cd5b2025858a28a3b39, externalId=BBDN-Java-REST-Demo, dataSourceId=_13_1,
↪ courseId=BBDN-Java-REST-Demo, name=Course Used For REST Demo - Java, description=Course
↪ Used For REST Demo - Java, created=2016-07-27T19:50:07.376Z, organization=false,
↪ ultraStatus=Classic, allowGuests=true, readOnly=false, termId=_7_1,
↪ availability=Availability [available=Yes], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.CourseHandler - UPDATE
[main] INFO bbdn.rest.services.CourseHandler - Course [id=null, uuid=null,
↪ externalId=BBDN-Java-REST-Demo, dataSourceId=_13_1, courseId=BBDN-Java-REST-Demo,
↪ name=Course Used For REST Demo - Java, description=Course Used For REST Demo - Java,
↪ created=null, organization=false, ultraStatus=null, allowGuests=true, readOnly=false,
↪ termId=_7_1, availability=Availability [available=Yes], status=null, code=null,
↪ message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.CourseHandler - URI is
↪ http://localhost:9876/learn/api/public/v1/courses/externalId:BBDN-Java-REST-Demo
[main] INFO bbdn.rest.services.CourseHandler - Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
```

```
[main] INFO bbdn.rest.services.CourseHandler - Request Body: Course [id=null, uuid=null,
↪    externalId=BBDN-Java-REST-Demo, dataSourceId=_13_1, courseId=BBDN-Java-REST-Demo,
↪    name=Course Used For REST Demo - Java, description=Course Used For REST Demo - Java,
↪    created=null, organization=false, ultraStatus=null, allowGuests=true, readOnly=false,
↪    termId=_7_1, availability=Availability [available=Yes], status=null, code=null,
↪    message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.CourseHandler - Response: <200 OK,Course [id=_7_1,
↪    uuid=9ec59f4d3e094cd5b2025858a28a3b39, externalId=BBDN-Java-REST-Demo, dataSourceId=_13_1,
↪    courseId=BBDN-Java-REST-Demo, name=Course Used For REST Demo - Java, description=Course
↪    Used For REST Demo - Java, created=2016-07-27T19:50:07.376Z, organization=false,
↪    ultraStatus=Classic, allowGuests=true, readOnly=false, termId=_7_1,
↪    availability=Availability [available=Yes], status=null, code=null, message=null,
↪    developerMessage=null, extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪    X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪    3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪    must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:07 GMT], Expires=[Mon, 27 Jul 2015
↪    19:50:07 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪    'self'], Set-Cookie=[JSESSIONID=64C50ECFDE762D6C97C5409AB26DF00B; Path=/learn/api],
↪    X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪    Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed, 27
↪    Jul 2016 19:50:07 GMT]}>
[main] INFO bbdn.rest.services.CourseHandler - Course: Course [id=_7_1,
↪    uuid=9ec59f4d3e094cd5b2025858a28a3b39, externalId=BBDN-Java-REST-Demo, dataSourceId=_13_1,
↪    courseId=BBDN-Java-REST-Demo, name=Course Used For REST Demo - Java, description=Course
↪    Used For REST Demo - Java, created=2016-07-27T19:50:07.376Z, organization=false,
↪    ultraStatus=Classic, allowGuests=true, readOnly=false, termId=_7_1,
↪    availability=Availability [available=Yes], status=null, code=null, message=null,
↪    developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.UserHandler - CREATE
[main] INFO bbdn.rest.services.UserHandler - User [id=null, uuid=null,
↪    externalId=bbdnrestdemojavauser, dataSourceId=_13_1, userName=restjavauser,
↪    password=Bl@ckb0ard!, studentId=null, educationLevel=null, gender=null, birthDate=null,
↪    created=null, lastLogin=null, availability=Availability [available=Yes], name=Name
↪    [given=Java, family=Restdemo], contact=Contact [email=developers@blackboard.com],
↪    status=null, code=null, message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.UserHandler - URI is
↪    http://localhost:9876/learn/api/public/v1/users
[main] INFO bbdn.rest.services.UserHandler - Request Headers: {Authorization=[Bearer
↪    uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.UserHandler - Request Body: User [id=null, uuid=null,
↪    externalId=bbdnrestdemojavauser, dataSourceId=_13_1, userName=restjavauser,
↪    password=Bl@ckb0ard!, studentId=null, educationLevel=null, gender=null, birthDate=null,
↪    created=null, lastLogin=null, availability=Availability [available=Yes], name=Name
↪    [given=Java, family=Restdemo], contact=Contact
↪    [[email=developers@blackboard.com](mailto:email=developers@blackboard.com)], status=null,
↪    code=null, message=null, developerMessage=null, extrainfo=null]
```

```
[main] INFO bbdn.rest.services.UserHandler – Response: <201 Created,User [id=_9_1,
↪ uuid=c55303eaa7d541aab75ebd218d1a6e20, externalId=bbdnrestdemojavauser, dataSourceId=_13_1,
↪ userName=restjavauser, password=null, studentId=null, educationLevel=Unknown,
↪ gender=Unknown, birthDate=null, created=2016-07-27T19:50:07.970Z, lastLogin=null,
↪ availability=Availability [available=Yes], name=Name [given=Java, family=Restdemo],
↪ contact=Contact [email=developers@blackboard.com], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪ X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪ 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪ must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:07 GMT], Expires=[Mon, 27 Jul 2015
↪ 19:50:07 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪ 'self'], Set-Cookie=[JSESSIONID=679348D067502E72AD1ECAD13B33322E; Path=/learn/api],
↪ Location=[/learn/api/public/v1/users/_9_1],
↪ X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪ Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed, 27
↪ Jul 2016 19:50:07 GMT]}>
[main] INFO bbdn.rest.services.UserHandler – User: User [id=_9_1,
↪ uuid=c55303eaa7d541aab75ebd218d1a6e20, externalId=bbdnrestdemojavauser, dataSourceId=_13_1,
↪ userName=restjavauser, password=null, studentId=null, educationLevel=Unknown,
↪ gender=Unknown, birthDate=null, created=2016-07-27T19:50:07.970Z, lastLogin=null,
↪ availability=Availability [available=Yes], name=Name [given=Java, family=Restdemo],
↪ contact=Contact [email=developers@blackboard.com], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.UserHandler – READ
[main] INFO bbdn.rest.services.UserHandler – URI is
↪ http://localhost:9876/learn/api/public/v1/users/externalId:bbdnrestdemojavauser
[main] INFO bbdn.rest.services.UserHandler – Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.UserHandler – Request Body: User [id=null, uuid=null,
↪ externalId=null, dataSourceId=null, userName=null, password=null, studentId=null,
↪ educationLevel=null, gender=null, birthDate=null, created=null, lastLogin=null,
↪ availability=null, name=null, contact=null, status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.UserHandler – Response: <200 OK,User [id=_9_1,
↪ uuid=c55303eaa7d541aab75ebd218d1a6e20, externalId=bbdnrestdemojavauser, dataSourceId=_13_1,
↪ userName=restjavauser, password=null, studentId=null, educationLevel=Unknown,
↪ gender=Unknown, birthDate=null, created=2016-07-27T19:50:07.970Z, lastLogin=null,
↪ availability=Availability [available=Yes], name=Name [given=Java, family=Restdemo],
↪ contact=Contact [email=developers@blackboard.com], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪ X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪ 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪ must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:08 GMT], Expires=[Mon, 27 Jul 2015
↪ 19:50:08 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪ 'self'], Set-Cookie=[JSESSIONID=6754BF5FA616BB198CC2118EC1E3F5CB; Path=/learn/api],
↪ X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪ Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed,27
↪ Jul 2016 19:50:07 GMT]}>
[main] INFO bbdn.rest.services.UserHandler – User: User [id=_9_1,
↪ uuid=c55303eaa7d541aab75ebd218d1a6e20, externalId=bbdnrestdemojavauser, dataSourceId=_13_1,
↪ userName=restjavauser, password=null, studentId=null, educationLevel=Unknown,
↪ gender=Unknown, birthDate=null, created=2016-07-27T19:50:07.970Z, lastLogin=null,
↪ availability=Availability [available=Yes], name=Name [given=Java, family=Restdemo],
↪ contact=Contact [email=developers@blackboard.com], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null]
```

```
[main] INFO bbdn.rest.services.UserHandler - UPDATE
[main] INFO bbdn.rest.services.UserHandler - User [id=null, uuid=null,
↪ externalId=bbdnrestdemojavauser, dataSourceId=_13_1, userName=restjavauser,
↪ password=Bl@ckb0ard!, studentId=null, educationLevel=null, gender=null, birthDate=null,
↪ created=null, lastLogin=null, availability=Availability [available=Yes], name=Name
↪ [given=Java, family=Restdemo], contact=Contact [email=developers@blackboard.com],
↪ status=null, code=null, message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.UserHandler - URI is
↪ http://localhost:9876/learn/api/public/v1/users/externalId:bbdnrestdemojavauser
[main] INFO bbdn.rest.services.UserHandler - Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.UserHandler - Request Body: User [id=null, uuid=null,
↪ externalId=bbdnrestdemojavauser, dataSourceId=_13_1, userName=restjavauser,
↪ password=Bl@ckb0ard!, studentId=null, educationLevel=null, gender=null, birthDate=null,
↪ created=null, lastLogin=null, availability=Availability [available=Yes], name=Name
↪ [given=Java, family=Restdemo], contact=Contact [email=developers@blackboard.com],
↪ status=null, code=null, message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.UserHandler - Response: <200 OK,User [id=_9_1,
↪ uuid=c55303eaa7d541aab75ebd218d1a6e20, externalId=bbdnrestdemojavauser, dataSourceId=_13_1,
↪ userName=restjavauser, password=null, studentId=null, educationLevel=Unknown,
↪ gender=Unknown, birthDate=null, created=2016-07-27T19:50:07.970Z, lastLogin=null,
↪ availability=Availability [available=Yes], name=Name [given=Java, family=Restdemo],
↪ contact=Contact [email=developers@blackboard.com], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null],{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪ X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪ 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪ must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:08 GMT], Expires=[Mon, 27 Jul 2015
↪ 19:50:08 GMT], X-Frame-Options=[SAMEORIGIN], Content-Security-Policy=[frame-ancestors
↪ 'self'], Set-Cookie=[JSESSIONID=D8AC51CE6341F1C29832F9B9764C1A74; Path=/learn/api],
↪ X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪ Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed, 27
↪ Jul 2016 19:50:08 GMT]}>
[main] INFO bbdn.rest.services.UserHandler - User: User [id=_9_1,
↪ uuid=c55303eaa7d541aab75ebd218d1a6e20, externalId=bbdnrestdemojavauser, dataSourceId=_13_1,
↪ userName=restjavauser, password=null, studentId=null, educationLevel=Unknown,
↪ gender=Unknown, birthDate=null, created=2016-07-27T19:50:07.970Z, lastLogin=null,
↪ availability=Availability [available=Yes], name=Name [given=Java, family=Restdemo],
↪ contact=Contact [email=developers@blackboard.com], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.MembershipHandler - CREATE
[main] INFO bbdn.rest.services.MembershipHandler - Membership [userId=_9_1, courseId=_7_1,
↪ dataSourceId=_13_1, created=null, courseRoleId=Instructor, availability=Availability
↪ [available=Yes], status=null, code=null, message=null, developerMessage=null,
↪ extrainfo=null]
[main] INFO bbdn.rest.services.MembershipHandler - URI is
↪ [http://localhost:9876/learn/api/public/v1/courses/externalId:BBDN-Java-REST-Demo/users/externalId:bbdn
[main] INFO bbdn.rest.services.MembershipHandler - Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.MembershipHandler - Request Body: Membership [userId=_9_1,
↪ courseId=_7_1, dataSourceId=_13_1, created=null, courseRoleId=Instructor,
↪ availability=Availability [available=Yes], status=null, code=null, message=null,
↪ developerMessage=null, extrainfo=null]
```

```
[main] INFO bbdn.rest.services.MembershipHandler - Response: <201 Created,Membership
↪ [userId=_9_1, courseId=_7_1, dataSourceId=_13_1, created=2016-07-27T19:50:08.539Z,
↪ courseRoleId=Instructor, availability=Availability [available=Yes], status=null, code=null,
↪ message=null, developerMessage=null, extrainfo=null],{Server=[Apache-Coyote/1.1],
↪ P3P=[CP="CAO PSA OUR"], X-Blackboard-appserver=[dev.bbdn.local],
↪ X-Blackboard-product=[Learn &#8482; 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache],
↪ Cache-Control=[no-cache, max-age=0, no-store, must-revalidate], Last-Modified=[Sat, 27 Jul
↪ 1996 19:50:08 GMT], Expires=[Mon, 27 Jul 2015 19:50:08 GMT], X-Frame-Options=[SAMEORIGIN],
↪ Content-Security-Policy=[frame-ancestors 'self'],
↪ Set-Cookie=[JSESSIONID=6BCBBFC919575F4FA108817EF78FFC73; Path=/learn/api],
↪ Location=[/learn/api/public/v1/courses/_7_1/users/_9_1],
↪ X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪ Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed, 27
↪ Jul 2016 19:50:08 GMT]}>
[main] INFO bbdn.rest.services.MembershipHandler - Membership: Membership [userId=_9_1,
↪ courseId=_7_1, dataSourceId=_13_1, created=2016-07-27T19:50:08.539Z,
↪ courseRoleId=Instructor, availability=Availability [available=Yes], status=null, code=null,
↪ message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.MembershipHandler - READ
[main] INFO bbdn.rest.services.MembershipHandler - URI is
↪ http://localhost:9876/learn/api/public/v1/courses/externalId:BBDN-Java-REST-Demo/users/externalId:bbdn
[main] INFO bbdn.rest.services.MembershipHandler - Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.MembershipHandler - Request Body: Membership [userId=null,
↪ courseId=null, dataSourceId=null, created=null, courseRoleId=null, availability=null,
↪ status=null, code=null, message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.MembershipHandler - Response: <200 OK,Membership [userId=_9_1,
↪ courseId=_7_1, dataSourceId=_13_1, created=2016-07-27T19:50:08.539Z,
↪ courseRoleId=Instructor, availability=Availability [available=Yes], status=null, code=null,
↪ message=null, developerMessage=null, extrainfo=null],{Server=[Apache-Coyote/1.1],
↪ P3P=[CP="CAO PSA OUR"], X-Blackboard-appserver=[dev.bbdn.local],
↪ X-Blackboard-product=[Learn &#8482; 3000.1.1-rel.7+a3f61d9],Pragma=[no-cache],
↪ Cache-Control=[no-cache, max-age=0, no-store, must-revalidate], Last-Modified=[Sat, 27 Jul
↪ 1996 19:50:08 GMT], Expires=[Mon, 27 Jul 2015 19:50:08 GMT], X-Frame-Options=[SAMEORIGIN],
↪ Content-Security-Policy=[frame-ancestors 'self'],
↪ Set-Cookie=[JSESSIONID=64E622F16D37FF787E3319321D9B3740; Path=/learn/api],
↪ X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪ Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed, 27
↪ Jul 2016 19:50:08 GMT]}>
[main] INFO bbdn.rest.services.MembershipHandler - Membership: Membership [userId=_9_1,
↪ courseId=_7_1, dataSourceId=_13_1, created=2016-07-27T19:50:08.539Z,
↪ courseRoleId=Instructor, availability=Availability [available=Yes], status=null, code=null,
↪ message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.MembershipHandler - UPDATE
[main] INFO bbdn.rest.services.MembershipHandler - Membership [userId=_9_1, courseId=_7_1,
↪ dataSourceId=_13_1, created=null, courseRoleId=Instructor, availability=Availability
↪ [available=Yes], status=null, code=null, message=null, developerMessage=null,
↪ extrainfo=null]
[main] INFO bbdn.rest.services.MembershipHandler - URI is
↪ http://localhost:9876/learn/api/public/v1/courses/externalId:BBDN-Java-REST-Demo/users/externalId:bbdn
[main] INFO bbdn.rest.services.MembershipHandler - Request Headers: {Authorization=[Bearer
↪ uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
```

```
[main] INFO bbdn.rest.services.MembershipHandler - Request Body: Membership [userId=_9_1,
↪  courseId=_7_1, dataSourceId=_13_1, created=null,courseRoleId=Instructor,
↪  availability=Availability [available=Yes],status=null, code=null, message=null,
↪  developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.MembershipHandler - Response: <200 OK,Membership [userId=_9_1,
↪  courseId=_7_1, dataSourceId=_13_1, created=2016-07-27T19:50:08.539Z,
↪  courseRoleId=Instructor, availability=Availability [available=Yes], status=null, code=null,
↪  message=null, developerMessage=null, extrainfo=null],{Server=[Apache-Coyote/1.1],
↪  P3P=[CP="CAO PSA OUR"], X-Blackboard-appserver=[dev.bbdn.local],
↪  X-Blackboard-product=[Learn &#8482; 3000.1.1-rel.7+a3f61d9], Pragma=[no-cache],
↪  Cache-Control=[no-cache, max-age=0, no-store, must-revalidate], Last-Modified=[Sat, 27 Jul
↪  1996 19:50:08 GMT], Expires=[Mon, 27 Jul 2015 19:50:08 GMT], X-Frame-Options=[SAMEORIGIN],
↪  Content-Security-Policy=[frame-ancestors 'self'],
↪  Set-Cookie=[JSESSIONID=C26CD9BF1C6D987BEF83B8BFB2508692; Path=/learn/api],
↪  X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪  Content-Type=[application/json;charset=UTF-8], Transfer-Encoding=[chunked], Date=[Wed, 27
↪  Jul 2016 19:50:08 GMT]}>
[main] INFO bbdn.rest.services.MembershipHandler - Membership: Membership [userId=_9_1,
↪  courseId=_7_1, dataSourceId=_13_1, created=2016-07-27T19:50:08.539Z,
↪  courseRoleId=Instructor, availability=Availability [available=Yes], status=null, code=null,
↪  message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.MembershipHandler - DELETE
[main] INFO bbdn.rest.services.MembershipHandler - URI is
↪  http://localhost:9876/learn/api/public/v1/courses/externalId:BBDN-Java-REST-Demo/users/externalId:bbdn
[main] INFO bbdn.rest.services.MembershipHandler - Request Headers: {Authorization=[Bearer
↪  uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.MembershipHandler - Request Body: Membership [userId=null,
↪  courseId=null, dataSourceId=null, created=null, courseRoleId=null, availability=null,
↪  status=null, code=null, message=null, developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.MembershipHandler - Response: <204 No
↪  Content,{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪  X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn &#8482;
↪  3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0, no-store,
↪  must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:08 GMT], Expires=[Mon, 27 Jul 2015
↪  19:50:08 GMT], X-Frame-Options=[SAMEORIGIN],Content-Security-Policy=[frame-ancestors
↪  'self'], Set-Cookie=[JSESSIONID=614DC8400A97E7A1EE8A5D84FB5C91F9;
↪  Path=/learn/api],X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪  Content-Type=[application/json;charset=UTF-8], Date=[Wed, 27 Jul 2016 19:50:08 GMT]}>
[main] INFO bbdn.rest.services.UserHandler - DELETE
[main] INFO bbdn.rest.services.UserHandler - URI is
↪  http://localhost:9876/learn/api/public/v1/users/externalId:bbdnrestdemojavauser
[main] INFO bbdn.rest.services.UserHandler - Request Headers: {Authorization=[Bearer
↪  uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.UserHandler - Request Body: User [id=null, uuid=null,
↪  externalId=null, dataSourceId=null, userName=null, password=null, studentId=null,
↪  educationLevel=null, gender=null, birthDate=null, created=null, lastLogin=null,
↪  availability=null, name=null, contact=null, status=null, code=null, message=null,
↪  developerMessage=null, extrainfo=null]
```

```
[main] INFO bbdn.rest.services.UserHandler - Response: <204 No
↪   Content,{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪   X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn
↪   &#8482;3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0,
↪   no-store, must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:09 GMT], Expires=[Mon, 27
↪   Jul 2015 19:50:09 GMT],
↪   X-Frame-Options=[SAMEORIGIN],Content-Security-Policy=[frame-ancestors 'self'],
↪   Set-Cookie=[JSESSIONID=7962DE99F72343771643822431BB5EA3; Path=/learn/api],
↪   X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪   Content-Type=[application/json;charset=UTF-8], Date=[Wed, 27 Jul 2016 19:50:09 GMT]}>
[main] INFO bbdn.rest.services.CourseHandler - DELETE
[main] INFO bbdn.rest.services.CourseHandler - URI is
↪   http://localhost:9876/learn/api/public/v1/courses/externalId:BBDN-Java-REST-Demo
[main] INFO bbdn.rest.services.CourseHandler - Request Headers: {Authorization=[Bearer
↪   uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.CourseHandler - Request Body: Course [id=null, uuid=null,
↪   externalId=null, dataSourceId=null, courseId=null, name=null, description=null,
↪   created=null, organization=false, ultraStatus=null, allowGuests=false, readOnly=false,
↪   termId=null, availability=null, status=null, code=null, message=null,
↪   developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.CourseHandler - Response: <204 No
↪   Content,{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪   X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn
↪   &#8482;3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0,
↪   no-store, must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:09 GMT], Expires=[Mon, 27
↪   Jul 2015 19:50:09 GMT],
↪   X-Frame-Options=[SAMEORIGIN],Content-Security-Policy=[frame-ancestors 'self'],
↪   Set-Cookie=[JSESSIONID=D1F4182A949F2FE8FA7CA98C0BF59CB7;
↪   Path=/learn/api],X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪   Content-Type=[application/json;charset=UTF-8], Date=[Wed, 27 Jul 2016 19:50:09 GMT]}>
[main] INFO bbdn.rest.services.TermHandler - DELETE
[main] INFO bbdn.rest.services.TermHandler - URI is
↪   http://localhost:9876/learn/api/public/v1/terms/externalId:BBDN-TERM-JAVA
[main] INFO bbdn.rest.services.TermHandler - Request Headers: {Authorization=[Bearer
↪   uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
[main] INFO bbdn.rest.services.TermHandler - Request Body: Term [id=null,externalId=null,
↪   dataSourceId=null, name=null, description=null,availability=null, status=null, code=null,
↪   message=null,developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.TermHandler - Response: <204 No
↪   Content,{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪   X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn
↪   &#8482;3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0,
↪   no-store, must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:10 GMT], Expires=[Mon, 27
↪   Jul 2015 19:50:10 GMT],
↪   X-Frame-Options=[SAMEORIGIN],Content-Security-Policy=[frame-ancestors 'self'],
↪   Set-Cookie=[JSESSIONID=FF3871AC05EB0F3B28018DAFA0D7AB1D;
↪   Path=/learn/api],X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪   Content-Type=[application/json;charset=UTF-8], Date=[Wed, 27 Jul 2016 19:50:09 GMT]}>
[main] INFO bbdn.rest.services.DatasourceHandler - DELETE
[main] INFO bbdn.rest.services.DatasourceHandler - URI is
↪   http://localhost:9876/learn/api/public/v1/dataSources/externalId:BBDN-DSK-JAVA
[main] INFO bbdn.rest.services.DatasourceHandler - Request Headers: {Authorization=[Bearer
↪   uC5G2HvEsI1DWP24xD2lnaG35EQsenZs], Content-Type=[application/json]}
```

```
[main] INFO bbdn.rest.services.DatasourceHandler - Request Body: Datasource [id=null,
↪   externalId=null, description=null, status=null, code=null, message=null,
↪   developerMessage=null, extrainfo=null]
[main] INFO bbdn.rest.services.DatasourceHandler - Response: <204 No
↪   Content,{Server=[Apache-Coyote/1.1], P3P=[CP="CAO PSA OUR"],
↪   X-Blackboard-appserver=[dev.bbdn.local], X-Blackboard-product=[Learn
↪   &#8482;3000.1.1-rel.7+a3f61d9], Pragma=[no-cache], Cache-Control=[no-cache, max-age=0,
↪   no-store, must-revalidate], Last-Modified=[Sat, 27 Jul 1996 19:50:10 GMT], Expires=[Mon, 27
↪   Jul 2015 19:50:10 GMT],
↪   X-Frame-Options=[SAMEORIGIN],Content-Security-Policy=[frame-ancestors 'self'],
↪   Set-Cookie=[JSESSIONID=E7B5A007C1E329DFF0F8F477E580A4CB;
↪   Path=/learn/api],X-Blackboard-Context-Version=[3000.1.1-rel.5+0bc08ab],
↪   Content-Type=[application/json;charset=UTF-8], Date=[Wed, 27 Jul 2016 19:50:09 GMT]}>


BUILD SUCCESSFUL
Total time: 7.356 secs
```

What We Did   In this tutorial we built a Java command line application that creates, reads, updates, and deletes the five core REST objects: datasources, terms, courses, users, and memberships. Along the way, we learned how to institute OAuth, create HTTP messages and interact with the Learn REST APIs, how to use Jackson to serialize and deserialize objects, and how to use Gradle to build and run command line Java applications.