First Steps

- Prerequisites
- Building Fedora
- Running the Fedora webapp
- REST API Examples
  - Create a new object
  - Delete an object

This document will walk you through getting the code, building the webapp, and exercising the REST API.

Prerequisites

Fedora requires Java 8 and Maven 3 (only for building the project). Check that you have them installed before proceeding:

```
$ java -version
java version "1.8.0_73"
Java(TM) SE Runtime Environment (build 1.8.0_73-b02)
Java HotSpot(TM) 64-Bit Server VM (build 25.73-b02, mixed mode)

$ mvn -version
Apache Maven 3.3.9 (bb52d8502b132ec0a5a3f6c09453c07478323dc5; 2015-11-10T10:41:47-06:00)
Maven home: /Users/kford1/Programs/apache-maven-3.3.9
Java version: 1.8.0_73, vendor: Oracle Corporation
Java home: /Library/Java/JavaVirtualMachines/jdk1.8.0_73.jdk/Contents/Home/jre
Default locale: en_US, platform encoding: UTF-8
OS name: "mac os x", version: "10.13.6", arch: "x86_64", family: "mac"
```

Building Fedora

You can download pre-compiled binaries and run Fedora in an existing Tomcat or Jetty instance or pre-packaged Jetty instances, or build Fedora from source (it's not difficult or time consuming).

Clone the latest code from Github:

```
$ git clone https://github.com/fcrepo4/fcrepo4.git
Cloning into 'fc4'...
remote: Counting objects: 40152, done.
remote: Compressing objects: 100% (12759/12759), done.
Receiving objects: 14% (5622/40152), 3.68 MiB | 912 KiB/s
[...]
$ cd fcrepo4
```

And build the project (the first time you build Fedora may take some time and bandwidth to download all the development dependencies):
Running the Fedora webapp

The webapp can be run from a bundled instance of jetty. The application will listen on port 8080 by default. If you would like to run the webapp on a different port, you can add the "jetty.port" option in your maven command (e.g. mvn -Djetty.port=8181 jetty:run)

```
$ cd fcrepo-webapp
$ MAVEN_OPTS="-Xmx512m" mvn jetty:run
```

There's a nice, friendly HTML interface waiting for you at http://localhost:8080/rest/ (the default username and password is fedoraAdmin)

REST API Examples

The REST API documentation provides a full listing of the features and functions of the REST API. Here are some examples to get you started:
The REST API supports a variety of RDF serialization formats, including:
Create a new object
  * Trying ::1...
  * TCP_NODELAY set
  * Connected to localhost (::1) port 8080 (#0)
  * Server auth using Basic with user 'fedoraAdmin'
  > POST /rest/ HTTP/1.1
  > Host: localhost:8080
  > Authorization: Basic ZmVkb3JhQWRtaW46ZmVkb3JhQWRtaW4=
  > User-Agent: curl/7.60.0
  > Accept: */*
  >
  < HTTP/1.1 201 Created
  < Date: Thu, 02 Aug 2018 14:50:46 GMT
  < Expires: Thu, 01 Jan 1970 00:00:00 GMT
  < Set-Cookie: rememberMe=deleteMe; Path=/; Max-Age=0; Expires=Wed, 01-Aug-2018 14:50:46 GMT
  < ETag: W/"187414816f9004f45508769998bec981c590bea4"
  < Last-Modified: Thu, 02 Aug 2018 14:50:46 GMT
  < Link: <http://localhost:8080/static/constraints/ContainerConstraints.rdf>; rel="http://www.w3.org/ns/ldp#constrainedBy"
  < Link: <http://localhost:8080/rest/39965e35-4cad-4b96-b826-0c8612f773d5/fcr:acl>; rel="acl"
  < Location: http://localhost:8080/rest/39965e35-4cad-4b96-b826-0c8612f773d5
  < Content-Type: text/plain
  < Content-Length: 63
  < Server: Jetty(9.3.1.v20150714)
  <
  * Connection #0 to host localhost left intact
  http://localhost:8080/rest/80/7f/50/8e/a2/7f508ea2-15f7-48de-92a3-2575a1b749cf