

# Bootcamp Tutorial

## Prerequisites

- Install maven 3
- Configure git

```
git config --global user.name "FULL NAME"
git config --global user.email "EMAIL@proofpoint.com"
```

## Create base project from maven archetype

Create a project directory and run the command.



Make sure to replace USERNAME with your login name (the first part of your @proofpoint.com email address).

```
$ mvn archetype:generate -U -B \
    -DarchetypeGroupId=com.proofpoint.platform \
    -DarchetypeArtifactId=sample-server-archetype \
    -DarchetypeVersion=0.63 \
    -DgroupId=com.proofpoint.USERNAME.bootcamp \
    -DartifactId=person \
    -Dpackage=com.proofpoint.USERNAME.bootcamp

[INFO] Scanning for projects...
Downloading:
http://repo1.maven.org/maven2/org/apache/maven/plugins/maven-
metadata.xml
Downloading:
http://maven.us.proofpoint.com:8082/nexus/content/groups/proofpoint-
dev/org/apache/maven/plugins/maven-metadata.xml
...
[INFO]
[INFO] -----
[INFO] Building Maven Stub Project (No POM) 1
[INFO] -----
Downloading:
http://maven.us.proofpoint.com:8082/nexus/content/groups/proofpoint-
dev/org/apache/maven/plugins/maven-metadata.xml
Downloading:
```

<http://maven.us.proofpoint.com:8082/nexus/content/repositories/proofpoint-eng-snapshots/org/apache/maven/plugins/maven-metadata.xml>

...

[INFO]

[INFO] >>> maven-archetype-plugin:2.2:generate (default-cli) @ standalone-pom >>>

[INFO]

[INFO] <<< maven-archetype-plugin:2.2:generate (default-cli) @ standalone-pom <<<

[INFO]

[INFO] --- maven-archetype-plugin:2.2:generate (default-cli) @ standalone-pom ---

Downloading:

<http://maven.us.proofpoint.com:8082/nexus/content/groups/proofpoint-dev/org/apache/maven/plugins/maven-archetype-plugin/2.2/maven-archetype-plugin-2.2.pom>

...

[INFO] Generating project in Batch mode

[INFO] Archetype repository missing. Using the one from [com.proofpoint.platform:sample-server-archetype:0.61] found in catalog remote

...

[INFO] -----  
-----

[INFO] Using following parameters for creating project from Archetype: sample-server-archetype:0.63

[INFO] -----  
-----

[INFO] Parameter: groupId, Value: com.proofpoint.mkent.bootcamp

[INFO] Parameter: artifactId, Value: person

[INFO] Parameter: version, Value: 1.0-SNAPSHOT

[INFO] Parameter: package, Value: com.proofpoint.mkent.bootcamp

[INFO] Parameter: packageInPathFormat, Value: com/proofpoint/mkent/bootcamp

[INFO] Parameter: version, Value: 1.0-SNAPSHOT

[INFO] Parameter: package, Value: com.proofpoint.mkent.bootcamp

[INFO] Parameter: groupId, Value: com.proofpoint.mkent.bootcamp

[INFO] Parameter: artifactId, Value: person

[INFO] project created from Archetype in dir: /Users/mkent/dev/bootcamp/bootcamp4/person

[INFO] -----  
-----

[INFO] BUILD SUCCESS

[INFO] -----  
-----

[INFO] Total time: 14.682s

```
[INFO] Finished at: Mon May 07 07:37:11 PDT 2012
```

```
[INFO] Final Memory: 7M/81M
```

```
[INFO] -----  
-----
```

This command will create a directory named person, which contains a sample person server project.

## Build the project

```
$ cd person
```

```
$ mvn install
```

```
[INFO] Scanning for projects...
```

```
...
```

```
[INFO]
```

```
[INFO] -----  
-----
```

```
[INFO] Building person 1.0-SNAPSHOT
```

```
[INFO] -----  
-----
```

```
[...SNIP downloading the entire internet...]
```

```
[INFO] --- maven-enforcer-plugin:1.0.1:enforce (enforce-versions) @  
person ---
```

```
[INFO] The requirePluginVersions rule is currently not compatible with  
Maven3.
```

```
[INFO]
```

```
[INFO] --- maven-resources-plugin:2.4.3:resources (default-resources) @  
person ---
```

```
[INFO] Using 'UTF-8' encoding to copy filtered resources.
```

```
[INFO] skip non existing resourceDirectory  
/Users/mkent/dev/bootcamp/bootcamp4/person/src/main/resources
```

```
[INFO]
```

```
[INFO] --- maven-compiler-plugin:2.3.2:compile (default-compile) @  
person ---
```

```
[INFO] Compiling 10 source files to  
/Users/mkent/dev/bootcamp/bootcamp4/person/target/classes
```

```
[INFO]
```

```
[INFO] --- maven-resources-plugin:2.4.3:testResources (default-  
testResources) @ person ---
```

```
[INFO] Using 'UTF-8' encoding to copy filtered resources.
```

```
[INFO] Copying 2 resources
```

```
[INFO]
```

```
[INFO] --- maven-compiler-plugin:2.3.2:testCompile (default-
testCompile) @ person ---
[INFO] Compiling 7 source files to
/Users/mkent/dev/bootcamp/bootcamp4/person/target/test-classes
[INFO]
[INFO] --- maven-surefire-plugin:2.8.1:test (default-test) @ person ---
[INFO] Surefire report directory:
/Users/mkent/dev/bootcamp/bootcamp4/person/target/surefire-reports
```

---

## TESTS

---

Running TestSuite

[... SNIP most verbose test output in all of the world ...]

Tests run: 32, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 13.801 sec

Results :

Tests run: 32, Failures: 0, Errors: 0, Skipped: 0

```
[INFO]
[INFO] --- maven-jar-plugin:2.3.1:jar (default-jar) @ person ---
[INFO] Building jar:
/Users/mkent/dev/bootcamp/bootcamp4/person/target/person-1.0-
SNAPSHOT.jar
[INFO]
[INFO] --- maven-source-plugin:2.1.2:jar-no-fork (create-source-jar) @
person ---
[INFO] Building jar:
/Users/mkent/dev/bootcamp/bootcamp4/person/target/person-1.0-SNAPSHOT-
sources.jar
[INFO]
[INFO] --- maven-assembly-plugin:2.2:single (package) @ person ---
...
[INFO] Building tar :
/Users/mkent/dev/bootcamp/bootcamp4/person/target/person-1.0-SNAPSHOT-
distribution.tar.gz
[INFO]
[INFO] --- build-helper-maven-plugin:1.5:attach-artifact (attach-
distribution-tar.gz) @ person ---
[INFO]
```

```

[INFO] --- build-helper-maven-plugin:1.5:attach-artifact (attach-
readme) @ person ---
[INFO]
[INFO] --- maven-duplicate-finder-plugin:1.0.3:check (default) @ person
---
Downloading:
http://maven.us.proofpoint.com:8082/nexus/content/groups/proofpoint-
dev/com/ning/maven/plugins/maven-duplicate-finder-plugin/1.0.3/maven-
duplicate-finder-plugin-1.0.3.pom
...
[INFO] Checking compile classpath
[INFO] Checking runtime classpath
[INFO] Checking test classpath
[INFO]
[INFO] --- maven-install-plugin:2.3.1:install (default-install) @
person ---
[INFO] Installing
/Users/mkent/dev/bootcamp/bootcamp4/person/target/person-1.0-
SNAPSHOT.jar to
/Users/mkent/.m2/repository/com/proofpoint/mkent/bootcamp/person/1.0-
SNAPSHOT/person-1.0-SNAPSHOT.jar
[INFO] Installing /Users/mkent/dev/bootcamp/bootcamp4/person/pom.xml to
/Users/mkent/.m2/repository/com/proofpoint/mkent/bootcamp/person/1.0-
SNAPSHOT/person-1.0-SNAPSHOT.pom
[INFO] Installing
/Users/mkent/dev/bootcamp/bootcamp4/person/target/person-1.0-SNAPSHOT-
sources.jar to
/Users/mkent/.m2/repository/com/proofpoint/mkent/bootcamp/person/1.0-
SNAPSHOT/person-1.0-SNAPSHOT-sources.jar
[INFO] Installing
/Users/mkent/dev/bootcamp/bootcamp4/person/target/person-1.0-SNAPSHOT-
distribution.tar.gz to
/Users/mkent/.m2/repository/com/proofpoint/mkent/bootcamp/person/1.0-
SNAPSHOT/person-1.0-SNAPSHOT.tar.gz
[INFO] Installing /Users/mkent/dev/bootcamp/bootcamp4/person/README.txt
to
/Users/mkent/.m2/repository/com/proofpoint/mkent/bootcamp/person/1.0-
SNAPSHOT/person-1.0-SNAPSHOT-readme.txt
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1:29.156s
[INFO] Finished at: Mon May 07 07:39:21 PDT 2012
[INFO] Final Memory: 16M/81M
[INFO] -----

```

If the BUILD is SUCCESS then continue to next step, if the build FAILS then please ask the instructor or email your build output to hmc-bootcamp@proofpoint.com.

## Run the sample application

At this point the instructions are different for windows/non windows users since the application cannot be launched from command line on windows. The common section continues from "Output of Run" section below

### For windows users

## Open project from within IntelliJ

1. Launch IntelliJ
2. Select Open Project
3. Navigate to the directory where the project sources are located (i.e. the "person" directory) and open the file named pom.xml

## Run the server from within IntelliJ

1. Open Main class by choosing the "Go To" -> "Class..." menu and entering "Main"
2. Open "Main"->"Edit Configurations" and add "-Dconfig=etc/config.properties" under "VM parameters"
3. Right click on the main() method and select "Run"
4. Continues from "Output of Run" section below

### Mac/linux users

### Inspect the build artifacts

```
$ cd target
$ ls
archive-tmp/
classes/
generated-sources/
maven-archiver/
person-1.0-SNAPSHOT-distribution.tar.gz
person-1.0-SNAPSHOT-sources.jar
person-1.0-SNAPSHOT.jar
surefire/
surefire-reports/
test-classes/
```

- The `.jar` file contains the compiled classes from this project.
- The `-sources.jar` file contains the java source code for this project.
- The `.tar.gz` is the final server binary that we use in production.

## Unpack the application

```
$ tar xzf person-1.0-SNAPSHOT-distribution.tar.gz
```

## Add configuration

```
$ cp -R -p ../etc person-1.0-SNAPSHOT
$ touch person-1.0-SNAPSHOT/etc/jvm.config
$ ls person-1.0-SNAPSHOT/etc/
config.properties
jvm.config
log.properties
```

## Run server

```
$ cd person-1.0-SNAPSHOT
$ bin/launcher run
```

## Output of run (via intellij or launcher)

```
2011-11-15T21:21:37.253-
0800 INFO main com.proofpoint.log.Logging Logging to stderr
2011-11-15T21:21:37.266-
0800 INFO main com.proofpoint.bootstrap.Bootstrap Loading
configuration
2011-11-15T21:21:37.351-
0800 INFO main com.proofpoint.bootstrap.Bootstrap Initializing
logging
...
2011-11-15T21:21:39.820-
0800 INFO main com.proofpoint.bootstrap.LifecycleManager Life
cycle starting...
2011-11-15T21:21:39.821-
0800 INFO main com.proofpoint.bootstrap.LifecycleManager Life
cycle startup complete. System ready.
2011-11-15T21:21:39.822-0800 ERROR Announcer-0
com.proofpoint.discovery.client.Announcer Cannot connect to discovery
server for announce: No discovery servers are available
```

You can ignore the ERROR from the `com.proofpoint.discovery.client.Announcer`, this is only indicating that we don't have a discovery server to point to yet. We'll cover that later on.

## Test Server

In another terminal window:

```
$ curl -X PUT -H "content-type: application/json" --data '{ "name":  
"Matt", "email": "mstephenson@proofpoint.com"}'  
"http://localhost:8080/v1/person/mattstep"  
$ curl "http://localhost:8080/v1/person/"
```

output: [{"email":"mstephenson@proofpoint.com","name":"Matt"}]

```
$ curl "http://localhost:8080/v1/person/mattstep"
```

output:{"email":"mstephenson@proofpoint.com","name":"Matt","self":"<http://localhost:8080/v1/person/mattstep>"}

Kill the application with CTRL-C.

## Add a distribution management section to your POM

Place the following code into your pom.xml (file under your project directory) under the root element (i.e. as a child of the project tag)

```
<distributionManagement>  
  <repository>  
    <id>proofpoint</id>  
    <name>Proofpoint Releases</name>  
    <url>http://maven.us.proofpoint.com:8082/nexus/content/repositories/proofpoint-releases</url>  
  </repository>  
  <snapshotRepository>  
    <id>proofpoint-snapshots</id>  
    <name>Proofpoint Snapshots</name>  
    <url>http://maven.us.proofpoint.com:8082/nexus/content/repositories/proofpoint-snapshots</url>  
  </snapshotRepository>  
</distributionManagement>
```



# Add the project to your bootcamp git repository

Change directory to the root of the project

1. Change out of the project directory and clone your project repository:

```
$ cd ..  
$ git clone git@git.us.proofpoint.com:bootcamp4-${USERNAME}
```

1. Move project directory (person) into the project repository:

```
mv person bootcamp4-${USERNAME}  
cd bootcamp4-${USERNAME}/person
```

1. Commit files to local git repository

```
$ mvn clean  
$ git add .
```

```
$ git status  
# On branch master  
#  
# Initial commit  
#  
# Changes to be committed:  
#   (use "git rm --cached <file>..." to unstage)  
#  
#       new file:   .gitignore  
#       new file:   README.txt  
#       new file:   etc/config.properties  
#       new file:   etc/log.properties  
#       new file:   pom.xml  
#       new  
file:   src/main/java/com/proofpoint/mkent/bootcamp/Main.java  
#       new  
file:   src/main/java/com/proofpoint/mkent/bootcamp/MainModule.java  
#       new  
file:   src/main/java/com/proofpoint/mkent/bootcamp/Person.java  
#       new  
file:   src/main/java/com/proofpoint/mkent/bootcamp/PersonRepresentation.java  
#       new  
file:   src/main/java/com/proofpoint/mkent/bootcamp/PersonResource.java
```

```

#      new
file:  src/main/java/com/proofpoint/mkent/bootcamp/PersonStore.java
#      new
file:  src/main/java/com/proofpoint/mkent/bootcamp/PersonStoreStats.java
#      new
file:  src/main/java/com/proofpoint/mkent/bootcamp/PersonsResource.java
#      new
file:  src/main/java/com/proofpoint/mkent/bootcamp/StoreConfig.java
#      new
file:  src/test/java/com/proofpoint/mkent/bootcamp/TestPerson.java
#      new
file:  src/test/java/com/proofpoint/mkent/bootcamp/TestPersonRepresentation.j
#      new
file:  src/test/java/com/proofpoint/mkent/bootcamp/TestPersonResource.java
#      new
file:  src/test/java/com/proofpoint/mkent/bootcamp/TestPersonStore.java
#      new
file:  src/test/java/com/proofpoint/mkent/bootcamp/TestPersonsResource.java
#      new
file:  src/test/java/com/proofpoint/mkent/bootcamp/TestServer.java
#      new
file:  src/test/java/com/proofpoint/mkent/bootcamp/TestStoreConfig.java
#      new file:  src/test/resources/list.json
#      new file:  src/test/resources/single.json
#

```

```
$ git commit -m "Initial checkin"
```

```

[[master (root-commit) 9d19afe] Initial checkin
 23 files changed, 1438 insertions(+), 0 deletions(-)
 create mode 100644 .gitignore
 create mode 100644 README.txt
 create mode 100644 etc/config.properties
 create mode 100644 etc/log.properties
 create mode 100644 pom.xml
 create mode 100644
src/main/java/com/proofpoint/mkent/bootcamp/Main.java
 create mode 100644
src/main/java/com/proofpoint/mkent/bootcamp/MainModule.java
 create mode 100644
src/main/java/com/proofpoint/mkent/bootcamp/Person.java
 create mode 100644
src/main/java/com/proofpoint/mkent/bootcamp/PersonRepresentation.java
 create mode 100644
src/main/java/com/proofpoint/mkent/bootcamp/PersonResource.java
 create mode 100644
src/main/java/com/proofpoint/mkent/bootcamp/PersonStore.java

```

```

    create mode 100644
src/main/java/com/proofpoint/mkent/bootcamp/PersonStoreStats.java
    create mode 100644
src/main/java/com/proofpoint/mkent/bootcamp/PersonsResource.java
    create mode 100644
src/main/java/com/proofpoint/mkent/bootcamp/StoreConfig.java
    create mode 100644
src/test/java/com/proofpoint/mkent/bootcamp/TestPerson.java
    create mode 100644
src/test/java/com/proofpoint/mkent/bootcamp/TestPersonRepresentation.java
    create mode 100644
src/test/java/com/proofpoint/mkent/bootcamp/TestPersonResource.java
    create mode 100644
src/test/java/com/proofpoint/mkent/bootcamp/TestPersonStore.java
    create mode 100644
src/test/java/com/proofpoint/mkent/bootcamp/TestPersonsResource.java
    create mode 100644
src/test/java/com/proofpoint/mkent/bootcamp/TestServer.java
    create mode 100644
src/test/java/com/proofpoint/mkent/bootcamp/TestStoreConfig.java
    create mode 100644 src/test/resources/list.json
    create mode 100644 src/test/resources/single.json

```


```

$ git log
commit 9d19afecb08c48b03236115c91ecafb675b7bb9a
Author: Mark Kent <mkent@proofpoint.com>
Date:   Wed Apr 13 10:01:32 2011 -0700

```

Initial checkin

### 36. Push to your remote bootcamp repository

 Replace USERNAME with your actual user name

1.

```

$ git push origin master
Counting objects: 40, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (29/29), done.
Writing objects: 100% (40/40), 12.47 KiB, done.
Total 40 (delta 10), reused 0 (delta 0)
To git@git.us.proofpoint.com:bootcamp4-mkent
 * [new branch]      master -> master

```



Replace USERNAME with your actual user name

## Open project from within IntelliJ (windows users can skip this since they are done with it)

1. Launch IntelliJ
2. Select Open Project
3. Navigate to the directory where the project sources are located (i.e. the "person" directory) and open the file named pom.xml

## Run the server from within IntelliJ

1. Open Main class by choosing the "Go To" -> "Class..." menu and entering "Main"
2. Open "Main"->"Edit Configurations" and add "-Dconfig=etc/config.properties" under "VM parameters"
3. Right click on the main() method and select "Run"

### Test from the command line:

```
$ curl -X PUT -H "content-type: application/json" --data '{ "name":  
"Matt", "email": "mstephenson@proofpoint.com"}'  
"http://localhost:8080/v1/person/mattstep"
```

On Windows don't use single quotes:

```
>curl -X PUT -H "content-type: application/json" --data "{ \"name\":  
\"Matt\", \"email\": \"mstephenson@proofpoint.com\"}"  
"http://localhost:8080/v1/person/mattstep"
```

```
$ curl "http://localhost:8080/v1/person/"
```

```
output: [{"email":"mstephenson@proofpoint.com","name":"Matt"}]
```

```
$ curl "http://localhost:8080/v1/person/mattstep"
```

output:

```
{"email":"mstephenson@proofpoint.com","name":"Matt","self":"http://localhost:8080/v1"}
```

# Add a feature to the server

In the current version of the sample server, the "person" entries contain just a name and email, but no id. This makes it hard for clients to know what url to call to DELETE or GET specific entries. We want to add an "id" field and expose it as part of the http API, as in the following example:

```
{
  "id": "alice",
  "name": "Alice",
  "email": "alice@proofpoint.com"
}
```

The API needs to validate that the id specified in the json structure matches the id that's part of the url and return a "400 Bad Request" error otherwise.

So, this call should fail:

```
$ curl -X PUT -H "content-type: application/json" --data '{ "id":
"alice", "name": "Alice", "email": "alice@proofpoint.com"}'
"http://localhost:8080/v1/person/bob"
```

On Windows:

```
>curl -X PUT -H "content-type: application/json" --data "{ \"id\":
\"alice\", \"name\": \"Alice\", \"email\": \"alice@proofpoint.com\"}"
"http://localhost:8080/v1/person/bob"
```

As part of this task you should:

1. Implement the change and unit tests
2. Always ensure that your build runs successfully before going to next steps.
3. Commit it to your local repository
4. Push it to your remote git repository

NOTE: Doing smaller commits is consider better. So do not wait until the last moment to commit changes.