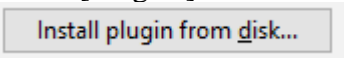




## Introduction to building and running Rest Samples

### Install Plugin

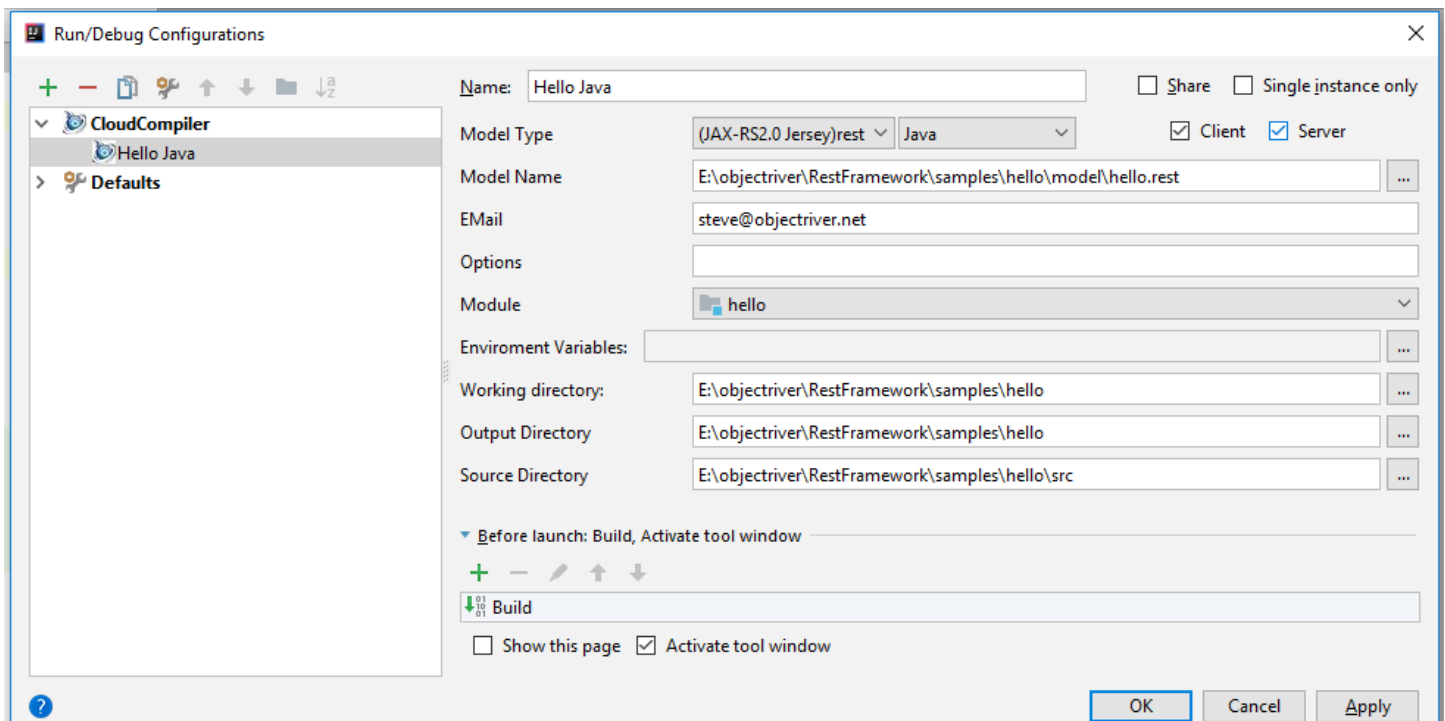
- IntelliJ Install CloudCompiler Plugin
  - File → Settings
  - Select [Plugins]
  - 
  - Directory  
    <objectriver>/sdk/intellij/ObjectRiverCloudCompilerPlugin.zip
  - Restart IntelliJ

### Set OBJECTRIVER\_HOME env variable

- set OBJECTRIVER\_HOME=C:/objectriver

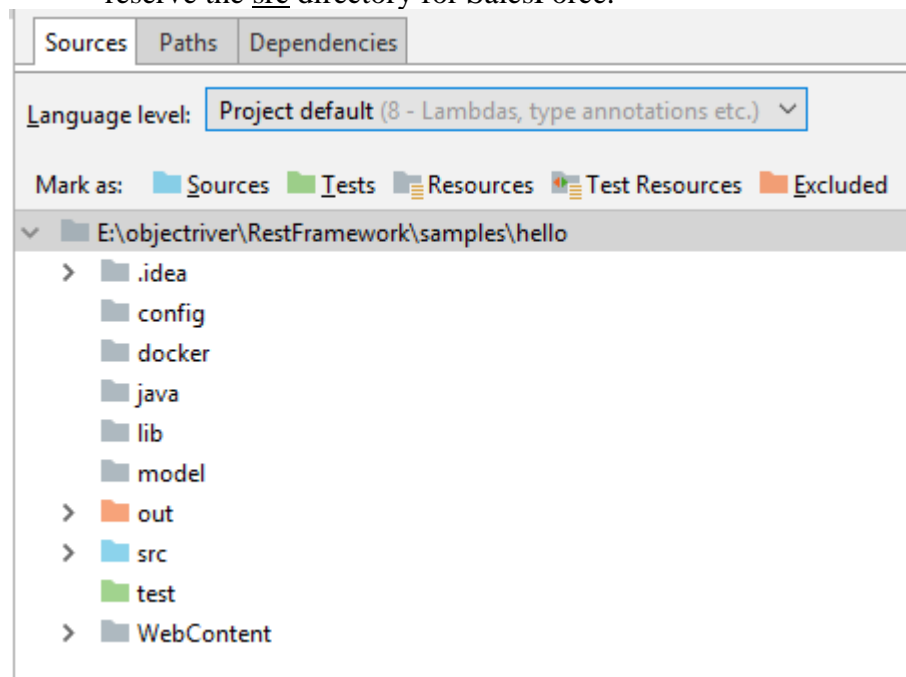
### Build and Execute Sample

- New Project
  - File → New → Project...
  - Create a new IntelliJ project in a sample directory.
- Run → Edit Configurations...
  - [+] → CloudCompiler
  - Fill in dialog below

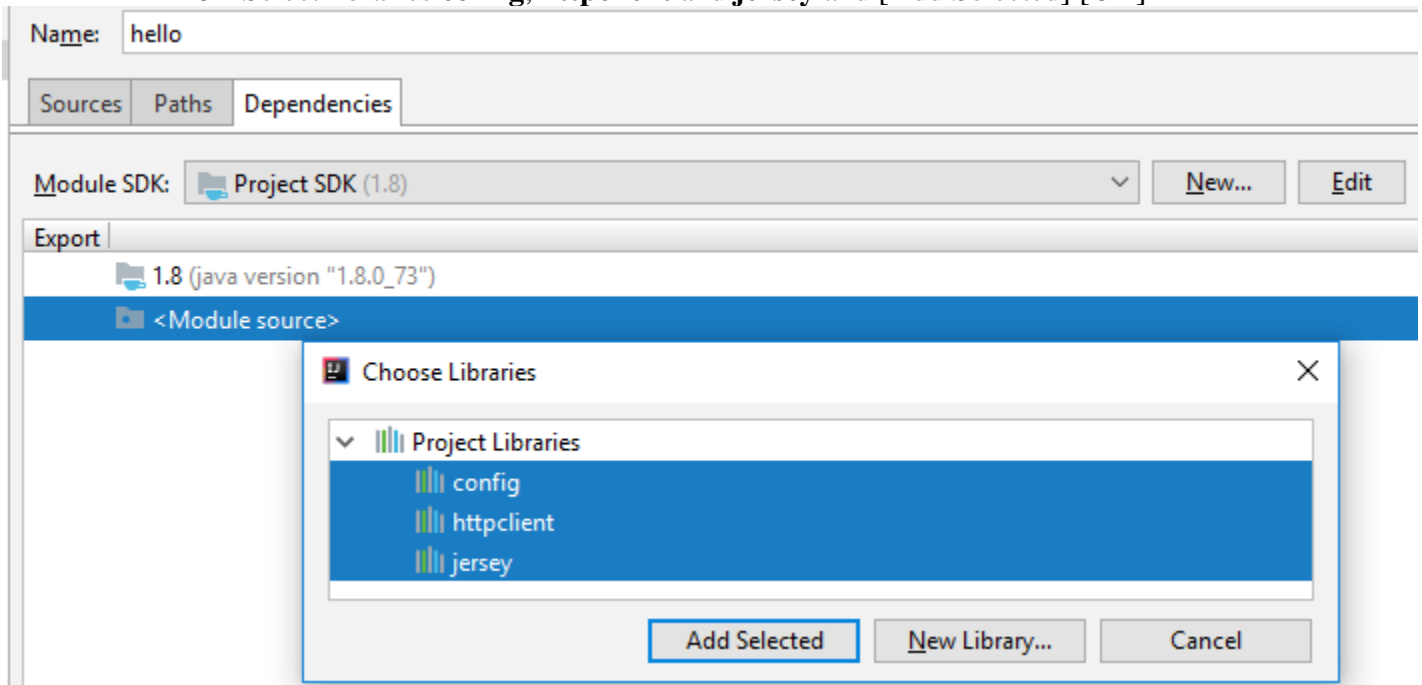


- Press (OK)
- Run CloudCompiler “Hello Java”
  - 
  - This will generate source code from the Hello.rest model.

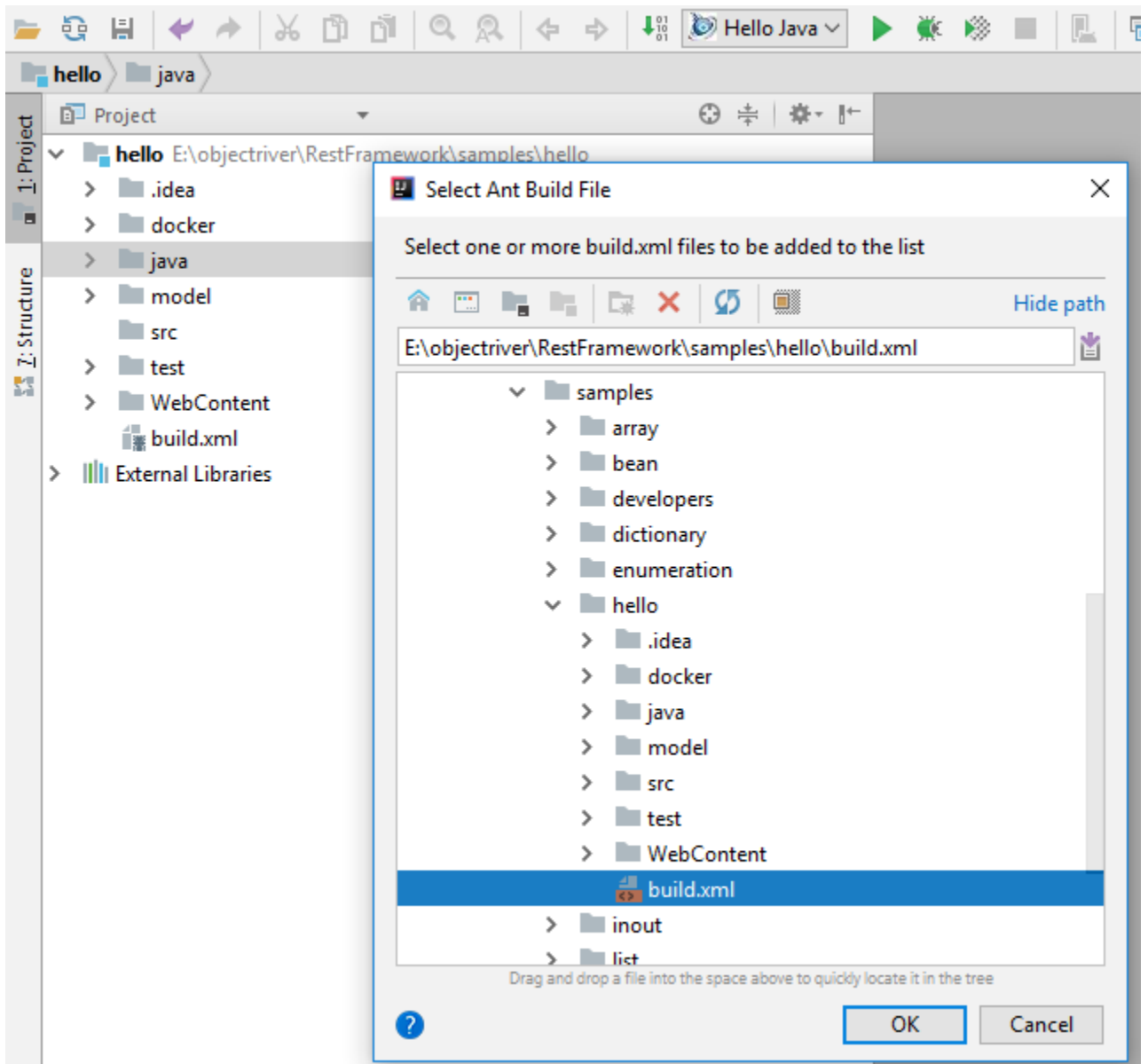
- File → Project Structure
  - [Modules]
  - Select Source Tab
    - Highlight sources and test directory's. This example chooses java for Java source to reserve the src directory for Salesforce.



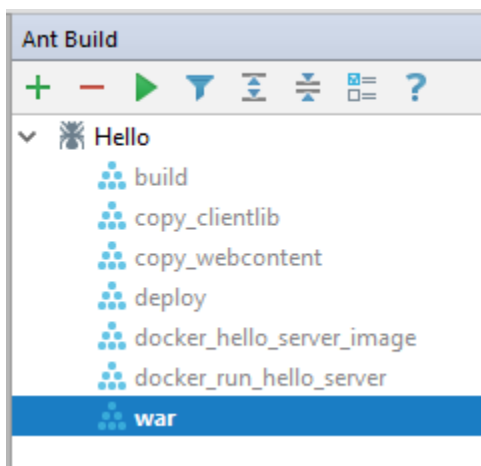
- Select Dependencies Tab
- [+] → Libraries
- Select libraries **config**, **httpclient** and **jersey** and [Add Selected] [OK]



- View → Tools Window → Ant Build
  - [+]

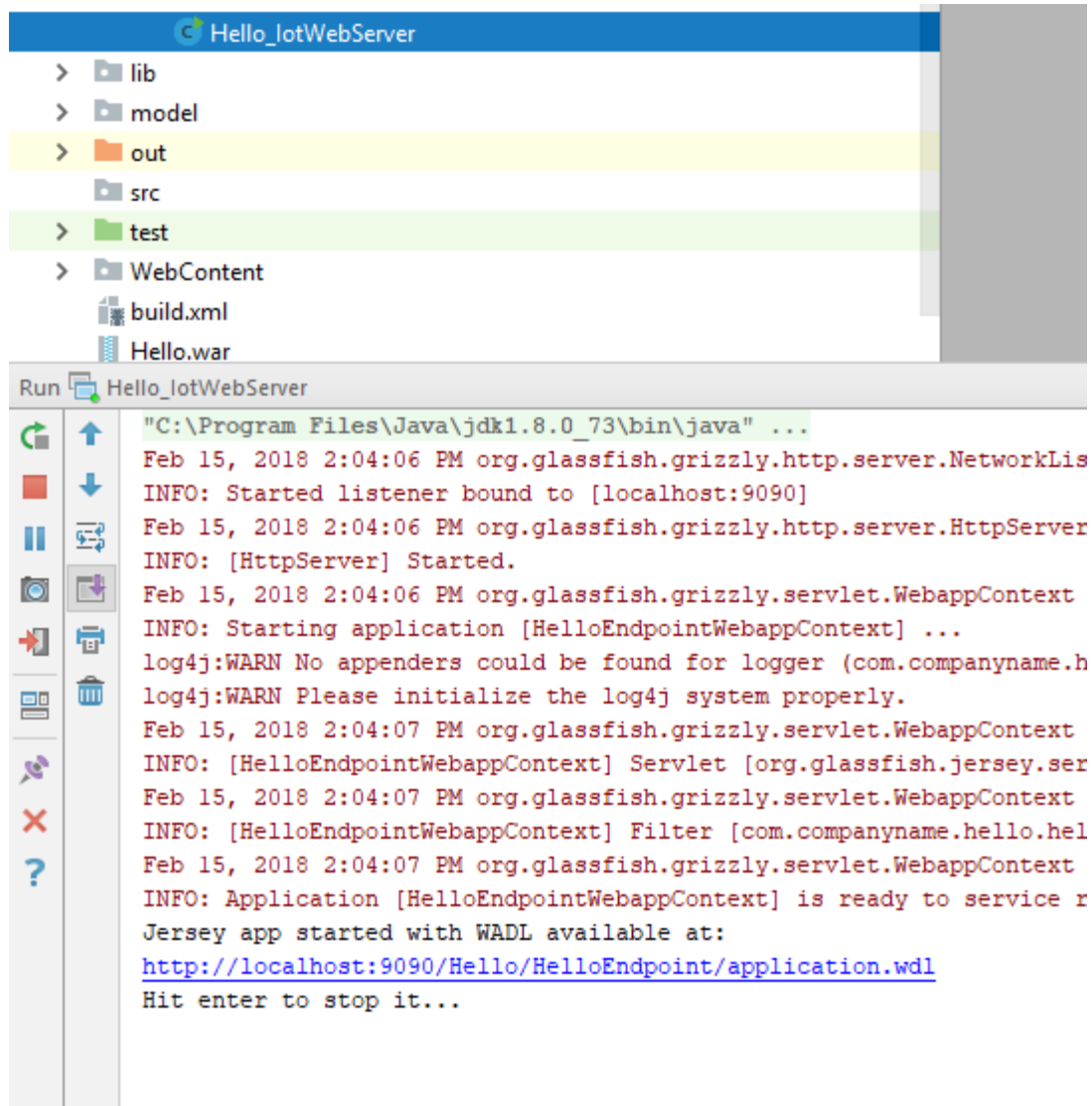


- Execute Ant war target.



- This will create a Hello.war file.
- Execute Hello\_IotServer
  - Select the Hello\_IotServer, <RightClick>→Run
    - Server run on port 9090. See config directory.

- By default the server will respond with random data. Ie. Server Mocking.
- This server can be directly debugged.
- This Server uses Oracle Grizzly for deploying to lightweight Internet Of Things servers. Docker/Alpine and Tomcat server uses JAXRS2 bundled in Tomcat



- Execute client test(s)
  - Select the HelloEndpointClientTest, <RightClick>→Run
    - Client uses on port 9090. See config directory.
    - By default the client will send random data. Ie. Client Mocking.
    -

Run: Hello\_IotWebServer HelloEndpointClientTest

```

"C:\Program Files\Java\jdk1.8.0_73\bin\java" ...
Executing request GET http://localhost:9090/Hello/HelloEndpoint/hello_query?hello=OVQJUGN HTTP/1.1
log4j:WARN No appenders could be found for logger (org.apache.http.client.protocol.RequestAddCookies).
log4j:WARN Please initialize the log4j system properly.
String result = [
    "WKQJ"
]
Executing request GET http://localhost:9090/Hello/HelloEndpoint/hello_path/WPRTANNAYS/CPNIXCB? HTTP/1.1
String result = [
    "ODFBTJF"
]
Executing request POST http://localhost:9090/Hello/HelloEndpoint/hello_body? HTTP/1.1
String result = [
    "XALTMKTVDJ"
]

```

- Stop Hello\_IotServer

## Build and Execute Docker Alpine-Tomcat Server

- Execute Ant Target `docker_hello_server_image`
  - This will build a docker hello server image
- Execute Ant Target `docker_run_hello_server`
  - This will load the hello docker image into a running docker container.
- Re-run client test(s)
  - Select the HelloEndpointClientTest, <RightClick>→Run
- Open CMD shell
- Execute **docker ps**

```

C:\Users\slemmo>docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
5430fcb0bd30       hello_server:latest "/opt/tomcat/bin/cat..." 6 seconds ago       Up 5 seconds        8080/tcp, 0.0.0.0:9090->80/tcp

```

- Enjoy!