

How to

To develop quickly (if you write code, then it is automatically reloaded):

In a console:

```
gradle build --continuous
```

and another console:

```
gradle bootRun
```

To build war and runnable/deployable war:

```
gradle clean build
```

To start by gradle command:

```
gradle bootRun
```

To start by runnable war:

```
java -jar ROOT.war
```

w/ JVM options:

```
-Xmx256m -XX:MaxMetaspaceSize=128m -XX:MaxDirectMemorySize=90m -  
Xloggc:logs/gc_%p_%t.log -XX:+UseSerialGC -XX:+PrintHeapAtGC -  
XX:+PrintGCDetails -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -  
XX:NumberOfGCLogFiles=5 -XX:GCLogFileSize=10M -XX:GCTimeLimit=15 -  
XX:GCHeapFreeLimit=50 -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=logs/  
-XX:ErrorFile=logs/hs_err_pid%p.log -XX:NativeMemoryTracking=summary -  
agentlib:jdwp=transport=dt_socket,server=y,address=9009,suspend=n -  
Dderby.stream.error.file=logs/derby.log -Dderby.infolog.append=true -  
Dderby.language.logStatementText=true -Dderby.locks.deadlockTrace=true -  
Dderby.locks.monitor=true -Dderby.storage.rowLocking=true -  
Dcom.sun.management.jmxremote -Dcom.sun.management.jmxremote.port=7900 -  
Dcom.sun.management.jmxremote.ssl=false -  
Dcom.sun.management.jmxremote.authenticate=false -ea
```

To change port and start by runnable war:

```
java -jar ROOT.war --port=9000
```

To start by another container, for example

Tomcat:

```
cp ROOT.war ${CATALINA_HOME}/webapps/  
${CATALINA_HOME}/bin/startup.sh
```

To attach a remote debugger:

- Connection type: Standard (Socket Listen)
- Host: localhost
- Port: 9009