

Java Internal Web Server using Jersey and Jetty

- [Sample Code](#)
 - [Gradle Dependencies](#)
 - [Main Code](#)
 - [Resource](#)
- [References](#)

Sample Code

Gradle Dependencies

```
dependencies {  
  
    implementation "org.glassfish.jersey.inject:jersey-hk2:3.0.2"  
    implementation "org.glassfish.jersey.containers:jersey-container-jetty-http:3.0.2"  
    implementation "jakarta.servlet:jakarta.servlet-api:5.0.0"  
  
    ...  
}
```

Main Code

```
package com.company.test;  
  
import org.eclipse.jetty.server.Server;  
import org.glassfish.jersey.jetty.JettyHttpContainerFactory;  
import org.glassfish.jersey.server.ResourceConfig;  
  
import java.net.URI;  
  
public class TestServer {  
  
    public static void main(String args[]) {  
  
        // scan packages  
        final ResourceConfig config = new ResourceConfig().packages("com.company.test");  
  
        // Start Jetty Server  
        final Server server =  
            JettyHttpContainerFactory.createServer(  
                URI.create("http://localhost:8080/"), config);  
    }  
}
```

Resource

```
package com.company.test;

import jakarta.ws.rs.GET;
import jakarta.ws.rs.Path;
import jakarta.ws.rs.Produces;
import jakarta.ws.rs.core.MediaType;

@Path("/hello")
public class TestResource {

    @GET
    @Produces(MediaType.TEXT_PLAIN)
    public String hello() {
        return "Jersey Jetty example.";
    }
}
```

References

Reference	URL
Jersey and Jetty HTTP Server examples	https://mkyong.com/webservices/jax-rs/jersey-and-jetty-http-server-examples/
Jersey Documentation	https://eclipse-ee4j.github.io/jersey.github.io/documentation/latest3x/index.html