Servlets and Application Servers

Managed Server Architecture

What Are Application Servers?

- Middleware for server centric software.
- Acts as a container for your java application.
- Helps to manage persistence, security, and session among other things.

When to Use Application Servers

- Need to integrate with other existing systems and databases.
- Any web-integrated collaboration.
- Component oriented systems.

Java EE

All Java EE applications using an application server as a container.

There are many types of services in Java EE but we will focus on servlets for now.

Java EE Cont’d

Features of Java EE:

- Servlets
- Websockets
- Enterprise Java Beans
- Java Messaging Service

Tomcat Server

Application container for java servlets, websockets, and java server pages.

Tomcat Components

- Catalina - Servlet Container
- Coyote - Allows Catalina to act as a plan webserver.
- Jasper - Tomcat’s JSP Engine

TomEE

Provides Enterprise Java Bean support.

Configuration

The two most important configuration files to get Tomcat up and running are called server.xml and web.xml.

By default, these files are located at TOMCAT-HOME/conf/server.xml and TOMCAT-HOME/conf/web.xml, respectively.

server.xml

The elements of the server.xml file belong to five basic categories - Top Level Elements, Connectors, Containers, Nested Components, and Global Settings.

web.xml

This is mainly used to define your servlets and their resource mappings.

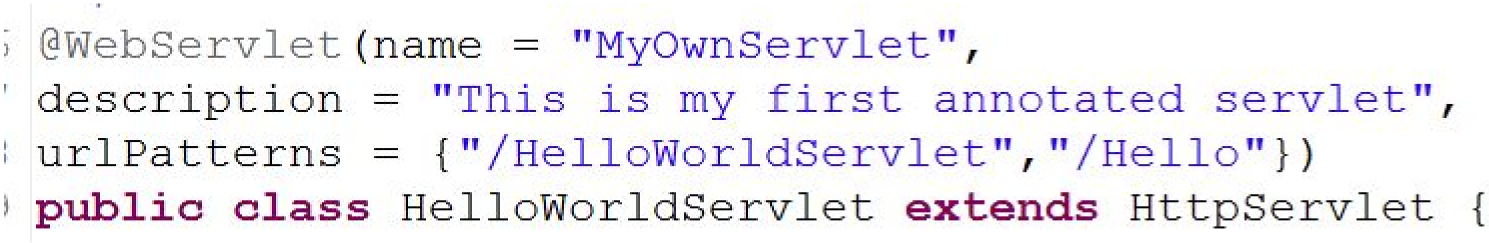
Every servlet you create will require an entry in web.xml.

Servlets

Servlets provide a high level API for handling HTTP calls.

WebServlet Annotation

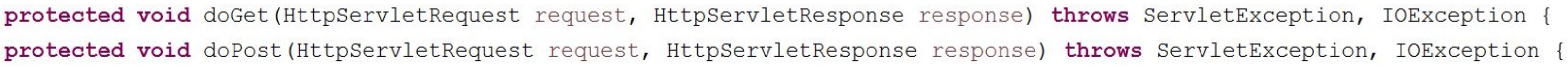
Used to drive Tomcat’s configuration



```
@WebServlet(name = "MyOwnServlet",
description = "This is my first annotated servlet",
urlPatterns = {"/HelloWorldServlet","/Hello"})
public class HelloWorldServlet extends HttpServlet {
```

HTTP Mappings

Servlets provide high level access to request and response streams as well as HTTP headers.



```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
```