Web Development Basics

Content

- HTTP Protocol
- Java EE Introduction
- Web Application Structure
- Web Servers
- Apache Tomcat

HTTP Protocol

Internet

World Wide Web (WWW)

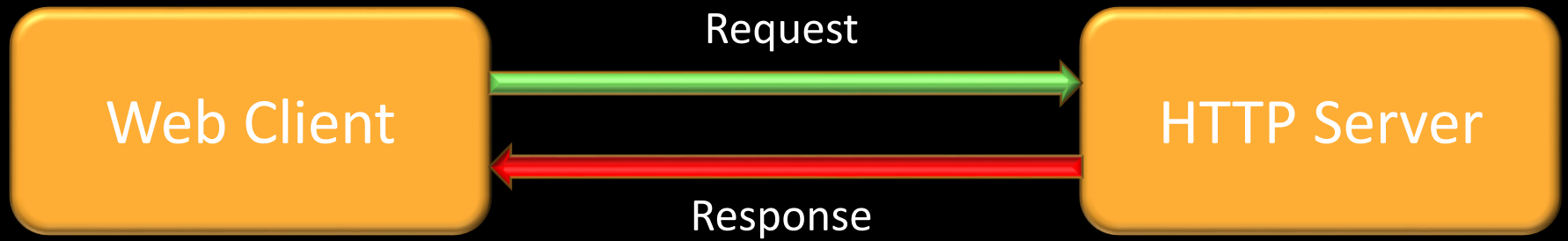
- **WWW** = **World Wide Web** = **Web** != **Internet**
- Internet is a global system of interconnected computer networks
- WWW is one of the services running in these networks
  - Global distributed information system in Internet (like E-mail, DNS, ...)
- Consists of set of resources (documents, images and others)
  - Located at different Internet servers, identified by URL
  - Accessed through standard protocols (like HTTP, HTTPS, FTP) by URL

Hypertext Transfer Protocol (HTTP)

- The HTTP is designed to enable communications between clients and servers.
- HTTP works as a request-response protocol between a client and server.
- Web servers provide Web content through the HTTP protocol
- Web clients access the Web content and display it

HTTP communication

- The web client sends HTTP request to the server



- The HTTP Server returns a response to the request

HTTP Request Methods

- Two commonly used methods for a request-response between a client and server are: GET and POST.
  - GET - Requests data from a specified resource
  - POST - Submits data to be processed to a specified resource

The GET Method

/demo/demoForm.jsp**?name1=value1&name2=value2**

- GET requests can be cached
- GET requests remain in the browser history
- GET requests can be bookmarked
- GET requests should never be used when dealing with sensitive data
- GET requests have length restrictions
- GET requests should be used only to retrieve data

The POST Method

```
POST /demo/demoForm.jsp
Host: java.ee
name1=value1&name2=value2
```

- POST requests are never cached
- POST requests do not remain in the browser history
- POST requests cannot be bookmarked
- POST requests have no restrictions on data length

Universal Resource Locator (URL)

- URL is used to locate the server and resource. Every resource on the web has it's own unique address.

```
http://localhost:8080/demo/demoForm.jsp
```

- http:// – Provides the communication protocol to be used in server-client communication.
- localhost – The unique address of the server, most of the times it's the hostname of the server that maps to unique IP address.
- 8080 – This is the port on which server is listening, it's optional and if we don't provide it in URL then request goes to the default port of the protocol.
- demo/demoForm.jsp – Resource requested from server. It can be static html, pdf, JSP, servlets, PHP etc.

Java EE Introduction

Java Enterprise Edition

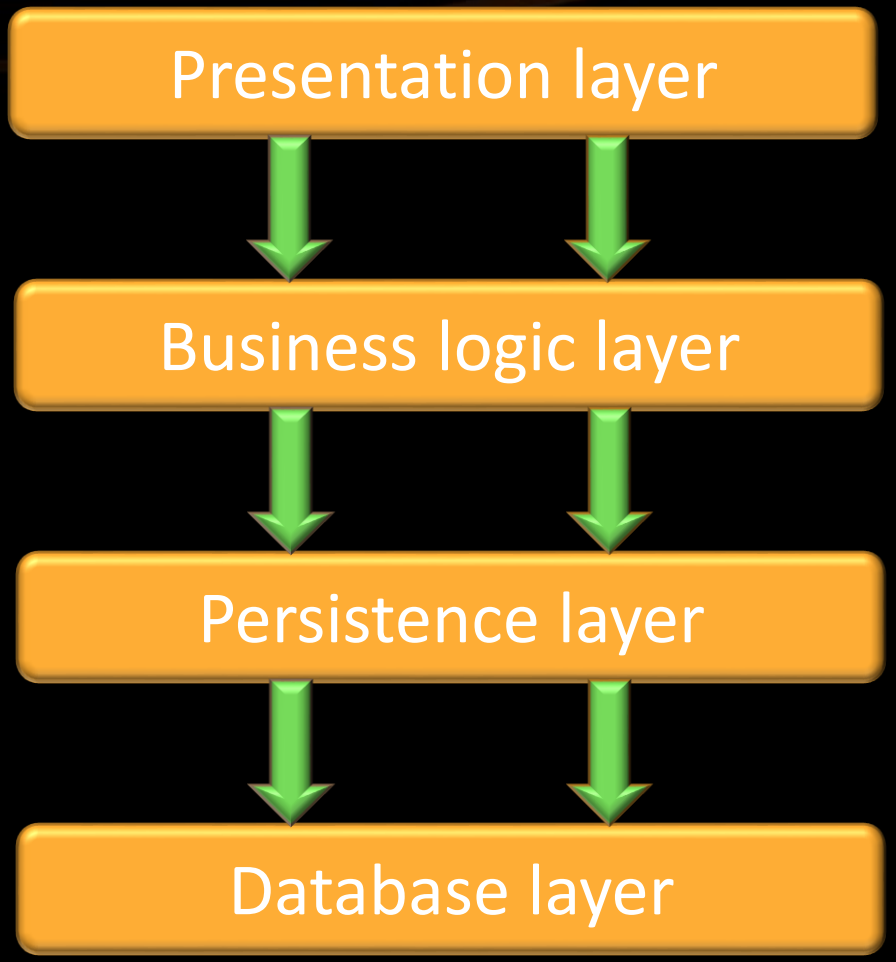
- Extension of Java SE
- Provides wide range of Application Program Interfaces (API) for development of enterprise applications
- Designed to provide multi layered architecture
- Follows convention over configuration principle

Multi Layered Architecture

- Standard pattern for server side applications
- Common purpose components are grouped together into layers
- Each layer performs it's function and delegates the rest of the work to the next layer

Traditional 4-Layered Architecture

- Presentation layer is responsible for the Graphical User Interface (GUI)
- Business logic layer contains the core application logic
- Persistence layer contains interface to access the Database
- Database layer is the actual Relational Database Management System (RDBMS) – Oracle, DB2, etc.



Java EE components

- presentation layer:
  - servlet, jsp, facelet, listener, filter, custom tag
- business logic layer:
  - ejb (session, message-driven), interceptor, timer
- persistence layer:
  - entity, listener
- database layer – out of Java scope

MusalaSoft

SOFTWARE UNIVERSITY FOUNDATION

Web Application Structure

Web Application Structure

- Web applications use a standard directory structure defined in the JavaEE specification.
  - You can deploy a Web application as a collection of files that use this directory structure, known as exploded directory format.
  - Another way is to deploy the Web application as an archived file called a WAR file.
  - It is recommended that you package and deploy your exploded Web application as part of an Enterprise application know as EAR file.

Directory Structure

- All servlets, classes, static files, and other resources belonging to a Web application are organized under a directory hierarchy.
  - DefaultWebApp/

    Place your static files, such as HTML files and JSP files in the directory that is the document root of your Web application.
  - DefaultWebApp/WEB-INF/web.xml

    The Web application deployment descriptor that configures the Web application.
  - DefaultWebApp/WEB-INF/server.xml

    The Server-specific deployment descriptor.
  - DefaultWebApp/WEB-INF/classes

    Contains server-side classes such as HTTP servlets and utility classes.
  - DefaultWebApp/WEB-INF/lib

    Contains JAR files used by the Web application, including JSP tag libraries.

Directory Structure Example

- The following is an example of a Web application directory structure:

```
myWebApp/
        WEB-INF/
                web.xml
                server.xml
                lib/
                        MyLib.jar
                classes/
                        MyPackage/
                                MyServlet.class
        index.html
        index.jsp
```

Web Servers

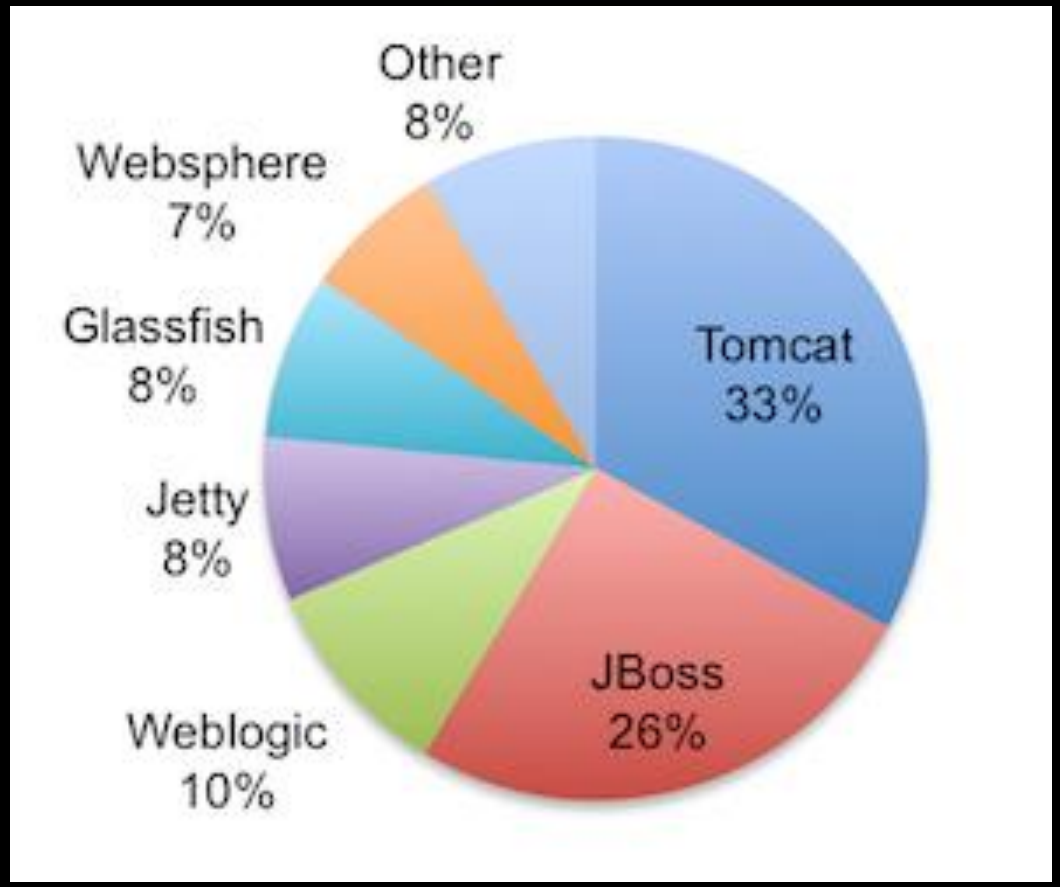
Web Server Definition

- A Web server is a program that uses HTTP to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computers' HTTP clients.

- Dedicated computers and appliances may be referred to as Web servers as well.

- Every Web server has an IP address and possibly a domain name

Popular Java EE Web Servers




MusalaSoft
SOFTWARE UNIVERSITY
FOUNDATION
Apache Tomcat

Apache Tomcat

- Apache Tomcat™ is an open source software implementation of:
  - Java Servlet
  - JavaServer Pages
  - Java Expression Language
  - Java WebSocket technologies
- Apache Tomcat powers numerous large-scale, mission-critical web applications across a diverse range of industries and organizations.

Apache Tomcat Versions

Servlet Spec	JSP Spec	EL Spec	WebSocket Spec	Apache Tomcat version	Actual release revision	Support Java Versions
4.0	TBD (2.4?)	TBD (3.1?)	TBD (1.2?)	9.0.x	9.0.0.M1 (alpha)	8 and later
3.1	2.3	3.0	1.1	8.0.x	8.0.30	7 and later
3.0	2.2	2.2	1.1	7.0.x	7.0.67	6 and later (WebSocket 1.1 requires 7 or later)
2.5	2.1	2.1	N/A	6.0.x	6.0.44	5 and later
2.4	2.0	N/A	N/A	5.5.x (archived)	5.5.36 (archived)	1.4 and later
2.3	1.2	N/A	N/A	4.1.x (archived)	4.1.40 (archived)	1.3 and later
2.2	1.1	N/A	N/A	3.3.x (archived)	3.3.2 (archived)	1.1 and later

Environment Setup

Demo

Resources

- http://www.journaldev.com/1854/java-web-application-tutorial-for-beginners
- https://docs.oracle.com/cd/E13222_01/wls/docs90/webapp/configurewebapp.html
- http://tomcat.apache.org/
- http://www.aicomputin.ch/