J2EE

&

Web Development

Jatin Ambasana

SUBJECT:
Advanced Java Programming

TOPIC:
J2EE
&
Web Development

OUTLINE

1. JAVA PLATFORM
2. J2EE ARCHITECTURE TYPES
3. EXPLORE JAVA EE CONTAINERS
4. TYPES OF SERVERS IN J2EE APPLICATION
5. HTTP PROTOCOLS AND API
6. REQUEST PROCESSING IN WEB APPLICATION
7. WEB APPLICATION STRUCTURE
8. WEB CONTAINERS AND WEB ARCHITECTURE MODELS

JAVA PLATFORM

- The **Java platform** is the name given to the computing platform from **Oracle** that helps users to *run* and *develop* Java applications.
- The **platform does not just enable a user to run and develop a Java application**, **but also features a wide variety of tools** that can help developers work efficiently with the Java programming language.
- The platform consists of two essential pieces of software:

1. The **Java Runtime Environment (JRE)**, which is **needed to *run*** Java applications,
2. The **Java Development Kit (JDK)**, which is **needed to *develop*** those Java applications . If you have installed the JDK, you should know that it comes equipped with a JRE as well.

JRE

- The Java Runtime Environment (JRE) is a **set of software tools** for development of Java applications. **It combines the Java Virtual Machine (JVM), platform core classes and supporting libraries**.

  JRE is part of the Java Development Kit (JDK), but can be downloaded separately.
- **JRE** was originally **developed by Sun Microsystems Inc**., a wholly-owned subsidiary of Oracle Corporation.

  Also known as **Java runtime**

JDK

- The **Java Development Kit**(**JDK**) is an **implementation** of either one **of the Java Platform**, **Standard Edition**, **Java Platform**, **Enterprise Edition**, or Java Platform, **Micro Edition** platforms **released** by **Oracle Corporation** in the form of a binary product aimed at Java developers on **Solaris, Linux, macOS or Windows**.
- The JDK includes a private JVM and a few other resources to finish the development of a Java Application.
- Since the introduction of the Java platform, it has been by far the most widely used Software Development Kit

JDK contents

- **appletviewer** – this tool can be used to run and debug Java applets without a web browser
- **extcheck** – a utility that **detects JAR** file conflicts
- **java – the loader** for Java applications. This tool is an interpreter and can interpret the class files generated by the javac compiler.
- **javac** – the Java compiler, which converts source code into Java bytecode
- **javadoc** – the documentation generator, which automatically generates documentation from source code comments
- **jdb** – the debugger
- **jstack** – utility that prints Java stack traces of Java threads

THE 3 DIFFERENT JAVA PLATFORM EDITIONS

1. **Java 2 Platform, Standard Edition (J2SE)** :- J2SE is used to develop **stand alone / desktop** portable java applications. **J2SE consists of two components. Core component and desktop component. Core component provides back end functionality. Desktop component provides GUI functionality.**
2. **Java 2 Platform, Enterprise Edition (J2EE)** :- J2EE is a platform to develop **multi-tier enterprise applications**. JEE includes Servlets API, Java Server Pages and Enterprise Java Beans. It was built on J2SE technology.
3. **Java 2 Platform, MicroEdition (J2ME)** :- JME is the technology for **micro electronic devices like PDAs, Mobile phones** etc. JME supports the flexibility for user interfaces, provides the security and uses the built-in network protocols for networked applications. J2ME applications can be ported across different devices.

J2EE ARCHITECTURE TIER

- The following are the tiers in J2EE application

Client Tier :

- The client tier includes the web components such as Servlets, JSP or standalone Java Desktop applications.

- This tier provides dynamic interfaces to the middle tier.

Middle Tier:

- This is also called as the server tier.

- In the middle tier enterprise beans and web services encapsulate distributable business logic for the applications which are reusable.

- The JEE application server contains the server-tier components which provides the platform for these web components for actions to be performed and data to be stored / persisted.

Enterprise data tier :

- The enterprise level data is stored / persisted preferably or typically in a relational database.

- In this tier, the JEE applications comprises of components, containers and services.

- All the web components (Servlets, JSP) provide dynamic requests and responses from a web page

EXPLORE

JAVA EE CONTAINERS

- **Containers** are the **interface between a component and the low-level platform**-specific functionality that supports the component.

Container Types

- **Java EE server**: The runtime portion of a Java EE product. A Java EE server provides EJB and web containers.
- **Enterprise JavaBeans (EJB) container**: Manages the execution of enterprise beans for Java EE applications. Enterprise beans and their container run on the Java EE server.
- **Web container**: Manages the execution of JSP page and servlet components for Java EE applications. Web components and their container run on the Java EE server.
- **Application client container**: Manages the execution of application client components. Application clients and their container run on the client.
- **Applet container**: Manages the execution of applets. Consists of a web browser and Java Plug-in running on the client together

APPLICATION SERVERS IN J2EE

- An **application server** is a software framework that provides both facilities **to create web applications and a server environment to run them**.

Application servers consist of

1. Web server connectors,
2. Computer programming languages,
3. Runtime libraries,
4. Database connectors.

TYPES OF SERVERS

- **Java application servers** : Java Platform, Enterprise Edition or Java EE (was J2EE) defines the core set of API and features of Java Application Servers.
- There are many open source Java application servers that support Java EE including

1. **WildFly** (formerly JBoss AS) from **JBoss** (division of RedHat)
2. **Geronimo** from **Apache**,
3. **TomEE** from **Apache**,
4. **GlassFish** from **Oracle**

- **.NET Framework :** Developed by Microsoft and is based on .NET framework.
- **Example:**

1. **Mono** (a cross platform open-source implementation of .NET supporting nearly all its features, with the exception of Windows OS-specific features), sponsored by Microsoft and released under the MIT License
2. **Base4** Application Server, an open source project
3. **TNAPS** Application Server, freeware application server, developed by TN LL

TYPES OF SERVERS

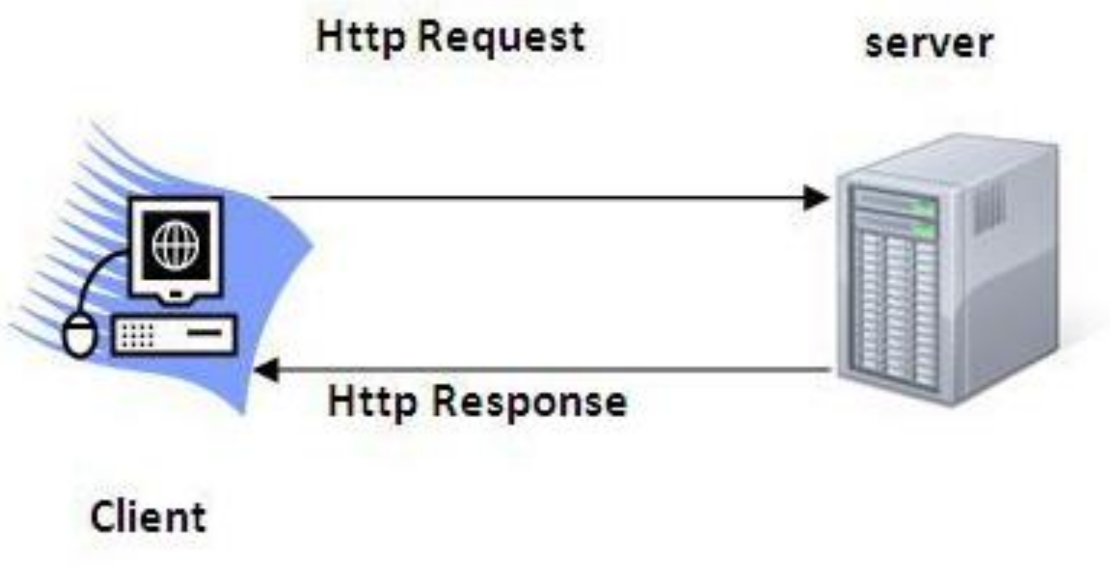
PHP application servers :

PHP application servers are used for running and managing PHPapplications.

1. **Zend Server**, built by **Zend Technologies**, provides application server functionality for the PHP-based applications.
2. **Appserver.io**, built by **TechDivision GmbH** is a multithreaded application server for PHP written in PHP

HTTP PROTOCOLS

1. The Hypertext Transfer Protocol (**HTTP**) is **application-level protocol**.
2. It is the **data communication protocol** used to establish communication between client and server.
3. HTTP is TCP/IP based communication protocol, which is used to deliver the data like **image files, query results, HTML files etc** on the World Wide Web (WWW).
4. The default port is **80**



Characteristics of HTTP

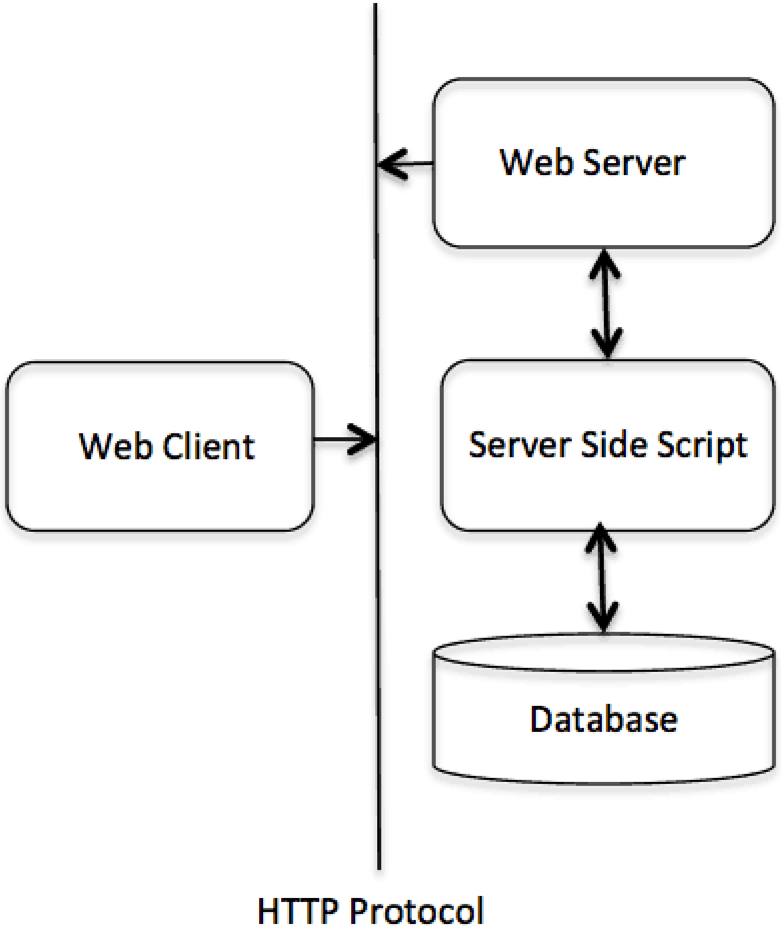
1. It is the protocol that **allows web servers and browsers to exchange data** over the web.
2. It is a **request response protocol**.
3. It uses the reliable TCP connections by default on TCP **port 80**.
4. It is **stateless** means each request is considered as the new request. In other words, server doesn't recognize the user by default.

The Basic Features of HTTP

- **HTTP is media independent:** It specifies that any type of media content can be sent by HTTP as long as both the server and the client can handle the data content.
- **HTTP is connectionless:** It is a connectionless approach in which HTTP client i.e., a browser initiates the HTTP request and after the request is sent the client disconnects from server and waits for the response.
- **HTTP is stateless:** The client and server are aware of each other during a current request only. Afterwards, both of them forget each other. Due to the stateless nature of protocol, neither the client nor the server can retain the information about different request across the web pages.

The Basic Architecture of HTTP (Hyper Text Transfer Protocol):

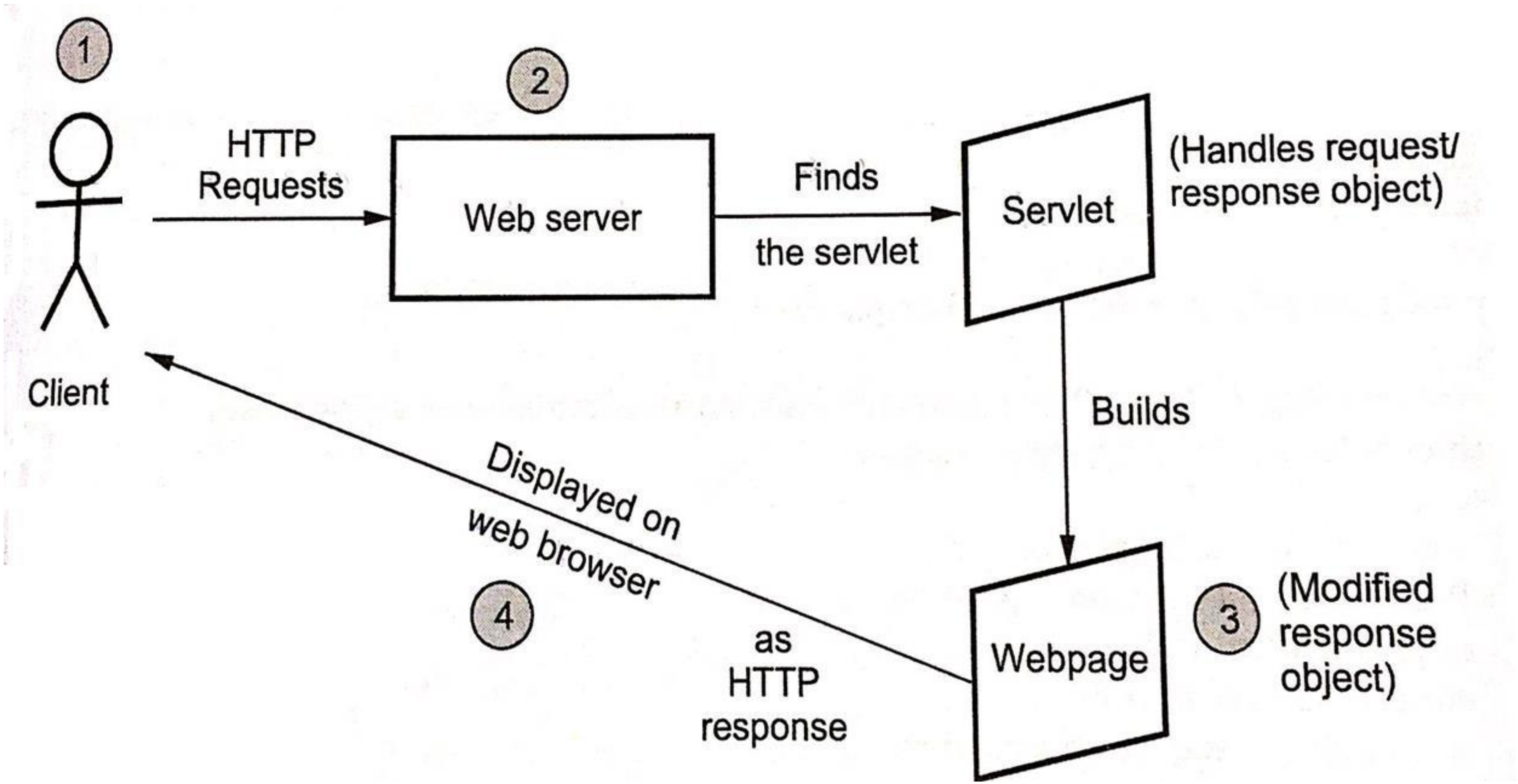
- HTTP is request/response protocol which is based on client/server based architecture.
- In this protocol, web browser, search engines, etc. behave as HTTP clients and the Web server like Servlet behaves as a server



HTTP Requests : Get vs Post

GET	POST
In case of Get request, only **limited amount of data can be sent** because data is sent in header.	In case of post request, **large amount of data can be sent** because data is sent in body.
Get request is **not secured** because data is exposed in URL bar.	Post request is **secured** because data is not exposed in URL bar.
Get request **can be bookmarked**.	Post request **cannot be bookmarked**.
Get request is **more efficient** and used more than Post.	Post request is **less efficient** and used less than get.

REQUEST PROCESSING IN WEB APPLICATION

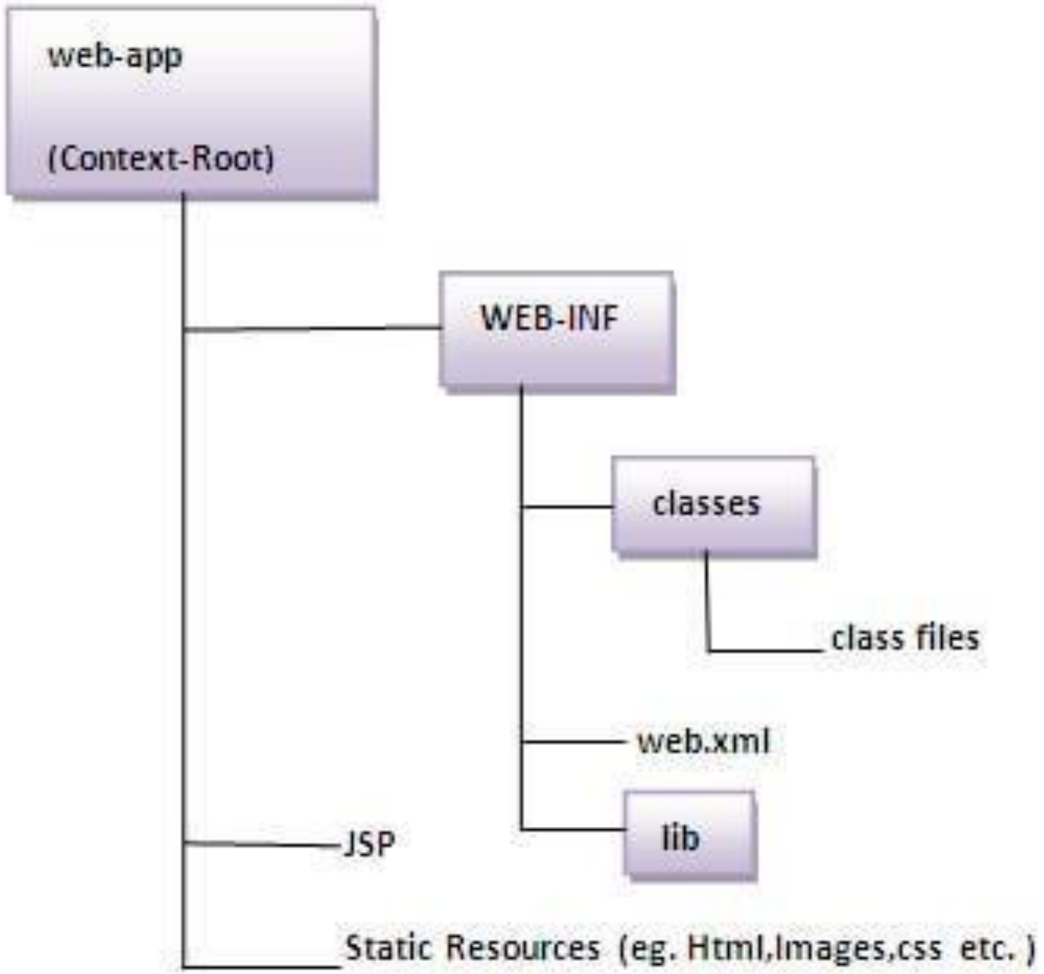


REQUEST PROCESSING IN WEB APPLICATION

1. When a client make a request for some servlet, he/she actually uses the Web browser in which request is written as a URL.
2. The web browser then sends this request to Web server. The web server first finds the requested servlet.
3. The obtained servlet gathers the relevant information in order to satisfy the client's request and builds a web page accordingly.
4. This web page is then displayed to the client. Thus request made by client gets satisfied by the servlet.

WEB APPLICATION STRUCTURE

- The directory structure of JSP based web application.
- It contains the jsp page outside the WEB-INF folder or in any directory.



WEB CONTAINERS

- A Web application runs within a Web container of a Web server.
- The Web container provides the runtime environment.
- Some Web servers may also provide additional services such as security and concurrency control. A Web server may work with an EJB server to provide some of those services.
- A Web server, however, does not need to be located on the same machine.
- Web applications are composed of web components and other data such as HTML pages. These components typically execute in a web server and may respond to HTTP requests from web clients.

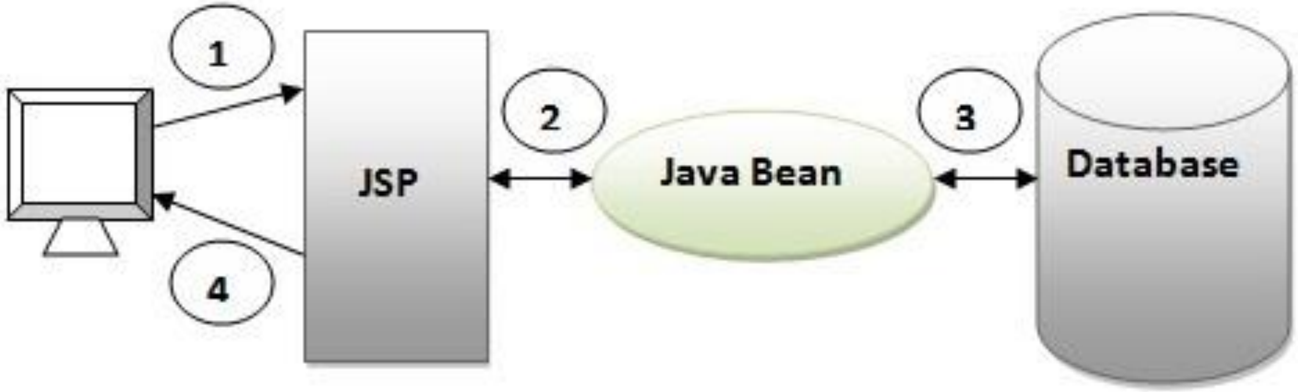
WEB ARCHITECTURE MODELS

- Before developing the web applications, we need to have idea about design models. There are two types of programming models

1. Model 1 Architecture
2. Model 2 (MVC) Architecture

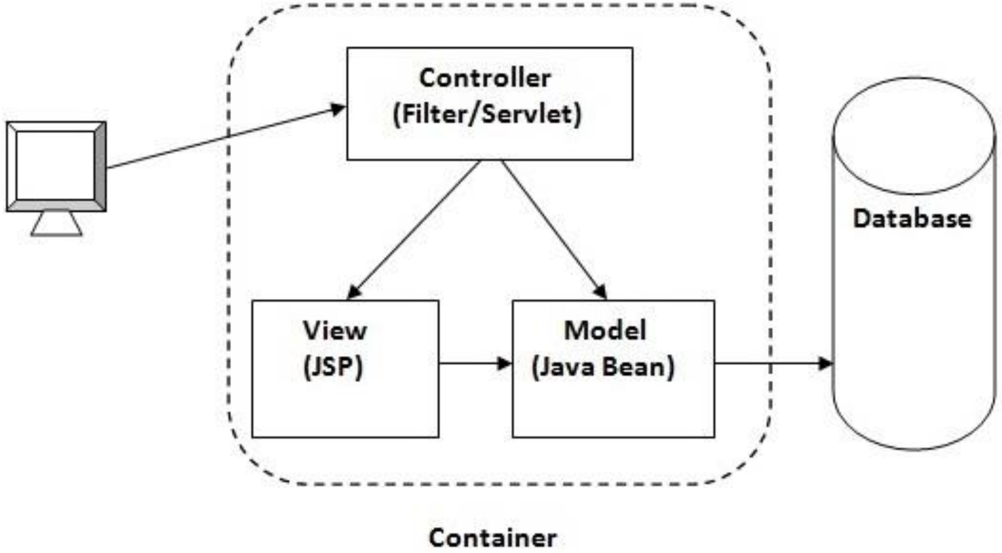
Model 1 Architecture

- As you can see in the figure, there is picture which show the flow of the model1 architecture.
- Browser sends request for the JSP page
- JSP accesses Java Bean and invokes business logic
- Java Bean connects to the database and get/save data
- Response is sent to the browser which is generated by JSP



Model 2 (MVC) Architecture

- Model 2 is based on the MVC (Model View Controller) design pattern. The MVC design pattern consists of three modules model, view and controller.
- **Model** The model represents the state (data) and business logic of the application.
- **View** The view module is responsible to display data i.e. it represents the presentation.
- **Controller** The controller module acts as an interface between view and model. It intercepts all the requests i.e. receives input and commands to Model / View to change accordingly.


Controller
(Filter/Servlet)
View
(JSP)
Model
(Java Bean)
Database
Container