Advance java

1. It is a set of multiple technologies.
2. Using these technologies, you can develop a web application.
3. Using JSP and servlet technology you can build the Web Application.

**Web Application**

1. Web Applications are also known as client and server application.
2. In this application client and server communicate with each other through a network.
3. For communication request and response are used.



1. There are 2 type of web application
   1. Static Web sites
      1. The content of the application is same for every user.
      2. This web sites are used for an informative site.
      3. To create this application you will required client side technologies like HTML, CSS, JS etc.
   2. Dynamic Web Application
      1. The content of the application will be change use to user.
      2. These are the Baking app, E-commerce, social sites etc.
      3. To develop these application you should know client and server side technologies such as HTML, CSS, JS, Servlet, JSP etc.

**Technologies**

**Client Side Technologies**

The code executes at client/bowser side. Such as HTML, CSS, JS etc.

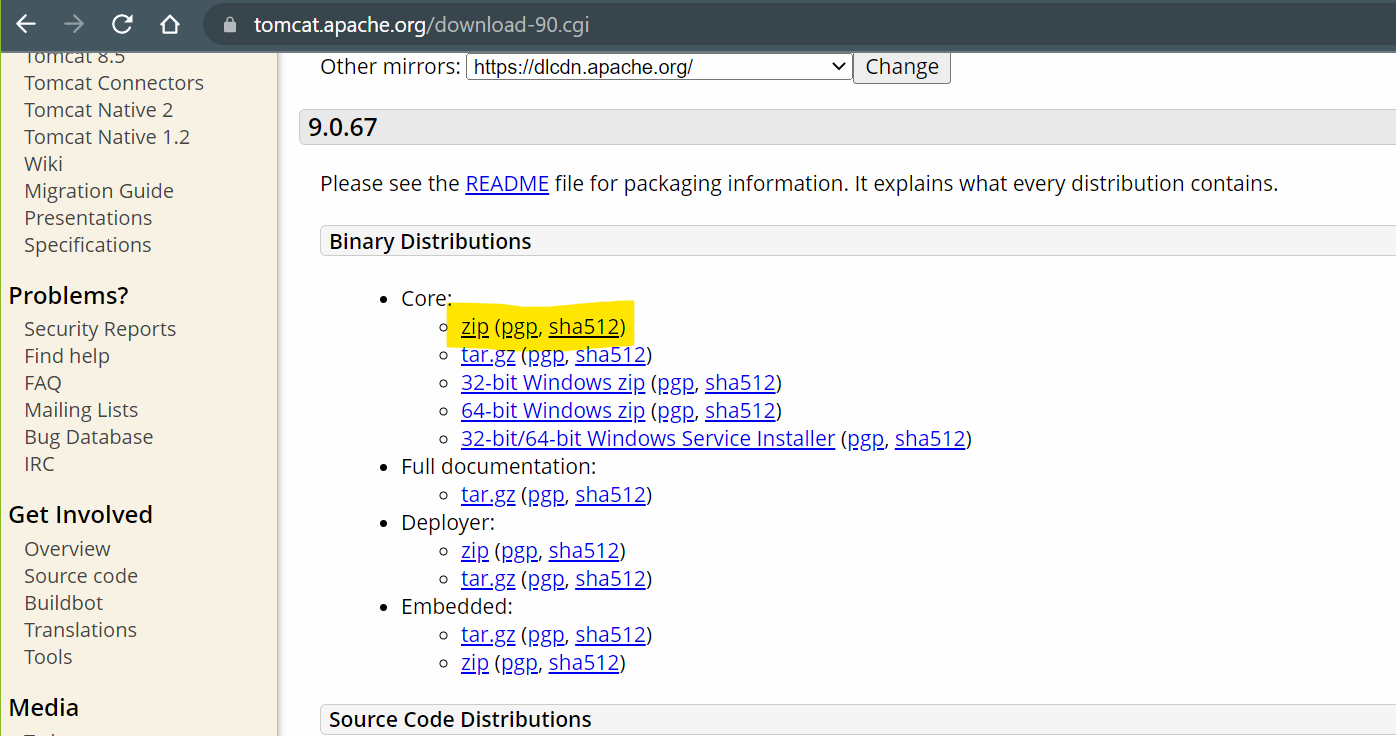
**Server Side Technology**

The code executes at server side. Such as JSP, Servlet, PHP, Asp.Net, Python.

**Environment Setup**

1. Download Tomcat server

[**https://tomcat.apache.org/download-90.cgi**](https://tomcat.apache.org/download-90.cgi)



1. Extract Server and Setup server into eclipse
   1. Copy and paste zip fine into an appropriate location.
   2. Extract the ZIP file.
   3. Server Setup in eclipse
      1. Set the eclipse perspective to “Java EE”
      2. Go to “servers” tab (at the bottom of the eclipse window)
      3. Click the link to add new Server.
      4. Expand the Apache option from the new Window
      5. Select the Tomcat version which is downloaded.
      6. Click on “Next”
      7. Browse for the tomcat path (path must the parent folder of the extracted tomcat in which bin, lib, config folders are present)
      8. Click on “Next” -> “Finish”

**Create Web Application in Eclipse**

1. “File” Menu -> “New” Option -> select “Dynamic Web Project”
2. Provide the Project Name (Make sure that target runtime is selected)
3. Click on “Next” -> Click “Next”
4. Make Sure that “Generate web.xml deployment descriptor” check box is selected
5. Click on “Finish”

**Dynamic Web Project Structure**



**Servlet**

1. Servlet is a java class.
2. This java class will not have a main method.
3. Servlet is use to create dynamic web pages.
4. Servlet is mainly used for accepting a request, process a request and generate response.
5. Servlet will be executed by the servlet container which is part of server.
6. Every servlet will have a unique URL. Using which servlet can be access/execute.
7. In servlet can write HTML language. Which has to add inside java which is also known as HTML in Java

**Return Response in Servlet**

1. Set the Content type of the response. That is you can specify which type of response you wanted to return to the client.
2. To set the type of Request and Response you can set the MIME type

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/MIME_types/Common_types>

1. Syntax to set Response type

**response.setContentType(“<MIME\_Type>”);**

1. Get the object of PrintWriter

**PrintWriter out = response.getWriter();**

1. Using Print method of the Printwriter class you can return a response.

**Client Request**

1. Client can send request to the server.
2. In a request there can be a data entered by client.
3. Request can be process at server side using HttpServletRequest object.
4. The data can be send from client to server in form parameter.

**Parameter**

1. It is a user data sends from the URL/Request
2. Parameters are added inside a URL after ‘?’
3. Parameters can be send internally which is known as request body(form data).
4. There can be a multiple parameters in a URL, every parameter separated by ‘&’
5. Every parameter has key and value. Were key is a name of the html component and value it the details provided by user.
6. Parameters are always in String type.
7. Parameters can be get using a request object.
8. Syntax:

**request.getParameter(“Key”) : String**

**Add Jar files into Web application**

1. Copy a jar file from a directory/folder.
2. Paste a jar file into src/main/webapp/WEB-INF/lib