**Difference between jdk, jre and jvm:**

**jdk**: is a tool required to develop java applications, jre is used to run the java program. This tool includes compiler(to convert java code into byte code), java application launcher( opens the jre, loads the classes and invokes the methods) and appletviewer.

Jdk is need to write a java program, to run java u need jre, jre=jvm+java package classes(util,math,awt etc)+runtime libraries.jdk is mainly targeted for java development. i.e u can create java file, compile java file and run a java file.

**JRE**: is used to run the program which includes class libraries and other supporting files, it doesnot require any other tool like compiler and debuggers. Actual jvm contains class libraris and other supporting files to run the program, so you need jre installation to run the program.

**JVM:** when v compile a java file, the java.class consists byte code. Jvm interprests the byte code into machine code depending upon on underlying the operating system and hardware.it is responsible for all the things like garbage collecter and array bounds checking etc…

**Jvm** is called as **virtual** beacuase it provide machine interface that doesnot depend on underlying operating system and hardware architecture. There are different jvm implementations are there. They may differ in performance, speed and realibility

**ADVANCED JAVA:**

**J2SE/JSE/JAVA SE**: IS used to develop standardalone applications,and to cover the fundamentals of technology we use j2se. desktop applications(simple apps that doesnot require server support): the concepts that will cover under this are:

Introduction, oops, string manipulation, exception handling, iostreams, n/w, gui

**J2EE/JEE/JAVA EE**: JAVA 2 ENTERPRISE EDITTION : is like web application to cover server side programming: concepts that cover under are: **SERVLETS, jsp, ejb, middleware services, jndi jta and jca**

**J2me**: is for micro programming in order to design mobile embedded software

These are technologies provided by sun microsystem

**FRAMEWORKS**: is a semi implemented application can be design enterprise applications as per our convince,

is a free fabrication of software components that customers can reuse can share can customize, can extend as per our application and implication

framework is collection of tools and apis, which can be provide good environment to design enterprise application in a simplified manner

**Structs, jibernate and spring, jsf** are not technologies ARE NOT provided by sun microsystem.

**SERVLETS:** When a client send a request that is static page the web browser will manage that request and sends to client, but it request a dynamic page web server cannot handle that request so In order to handle that request web server uses servlets which will provide dynamic web pages.

In order communicate web server with the **servlets** there will be web container which also called as servlet engine , what ever request from client will get it as a http request so here servlets doesnot understand http request here web container will convert it into a request which is understand by servlets. Here **web container** is responsible for everything that is to create thread and calls the method like service() and doget(). Thre are many servlets web container lnows to which servlets that request should be send by using a separate file: that contains: apache has given a web container like tomcat.

<wb app>

<servlet>

<s-name>loginserv></s>

<s-class>com.login></>

>serv-mapping>

<s-name>loginserv</>

<url-pattern> /logon</url-pattern>

</servlet-mapping>**</sweb>a**

**THE SERVLETS HAS 3 NAMES IN ORDER TO WEB CONTAINER KNOWS TO WHICH IT HAS TO BE SEND TH REQUEST.**

1. Clint known URL name
2. Deployer known secret internal name
3. Actual file name

**The http request: will contain 3 parts**

1. http method(which action to be performed)
2. url, the page to acesses
3. form parameters(username and passssowrds)

**the http response:**

1. status code(wheather the rquest is valid or not)
2. the content type(images, video etc..)
3. thae actual content.

Program to write functionalities of servlets:

**How to write a servlet class:**

There are 3 ways to create a class:

1. servlet (interface)
2. generic servlet class(it’s a abstract class, it’s a servlet child class)
3. http servlet(its fully concrete class it’s a abstract class)

if u use servlets interface there are 5 methods:

1)init() and returntype is void

2)service(servlet request, servlet response):void

3)destroy():void

4)get servletinfo: void

5)getservletconfig:void

If u goto http servlet if u want to implement class by servlet:

If u go with gneric servlet which is a child class for servlet interface, we use only abstract method that is

1)Service(servlet request, servlet response)

http servlet: don’t have any methods because all are concrete methods. They will recommednf to overwrite

1)doget(hhttp servletrequest, httpservletresponse)

and

2) dopost(httpsrvlet req, httpservlet response)

**If u want to create a class using servlet interface:**

If the application is helloworld:

See the notes

**MVC:**

**MVC Architecture has 3 parts:**

1. **view:** is the presentation layer in order to build the layer we include http, css and for browser validation we use javascript and also we can use JSP and adobe flex and tag library. When user sents the login the view will transfer to the controller.
2. **CONTROLLER:** controller will that request by using the servlets, jsp, and filters, frameworks like structs, jsf, spring mvc.
3. **MODAL**: modal willhave 2 parts **business** class and **dao** class when request is transferred into business class it uses java code, java transactions Api and java server API, spring security and EJB HERE THE REQUEST DETAILS WILLL SENT TO DAO the dao(data access object) contains jdbc in order to connect to database that dao will check weather the user login details in data base which will transfer into database then db will send the stored credintials to dao and dao to business here business class will compare the details and gives back to the user using controller and views.

Here business class are responsible for updating, creating using spring [transaction. @transaction](mailto:transaction.@transaction) is used to maintain consistency we can use spring instead of EJB if we use EJB we neded to write lot of code. In these thy will us logers to store data for safety.

**JDBC:** jdbc is used to connect the java application to the database. Jdbc is used to retrieve, delete. Add and update the data. The jdbc architecture consists 2 layers.

1)application layer: in these java application and jdbc will be there the java application sents java statements to jdbc.

2)database layer: here the drivers and database like sql, oracle, dbi, here jdbc will sent java stmnts to drivers here drivers will convert java stmnts to sql stmnts which can be understand by database oracle or sql.

There are four types of drivers:

1)jdbc –odbc bridge

2)native – api

3)network –thirdparty

4)pure java driver(thin)

1)

If we use 1st driver jdbc-odbc bridge driver:

Here it converts java application sendts dtata to jdbc and jdbc sents data to odbc which converts java appltn stmnts into opn database conectivity calls which can be understand by any database.

Note: here we need java, database and odbc applications install

Jdbc odbc bridge will use for only j2se applications.

2) native api

Here the java app will send data to jdbc and jdbc will send to driver which will talk directly to the data base functions and sends data to database. These is for j2se and for remote db access.

How to use odbc:

In order to use odbc 1st we need to create data source in odbc and load that same data source in java application. When you crearte a data source it will have accesses to database In order to load the same data sourcewe need to write a code on java application that is:

Example:

Public Db{

P s v m(string[] args)

{

//load driver

Class forname(“sun.jdbc.odbc.jdbcodbc.driver”);

//get connection

Connection conn = driver,get connection(“jdbc:odbc:ds-name”,”username”.”password”);

//execute the stmnts before thid crate a stmnt

Stmnt st = con.createstmnt();

Int I = St.executeupdate(“insert into student name(1, “dn”, “[dn@gmail.com](mailto:dn@gmail.com)”, 120);

S.out.pln(“th ffected rows are”, +i);

}