

Revision ---

```
String not_working_sql create table ? ( ? Int, ? Varchar(25) )
```

```
pstmt.setString(1,"student" )
```

```
//create table 'student' ( ? Int, ? Varchar(25) ) } MySQL generates error
```

```
Scanner sc =
```

```
String tname = sc.next();
```

```
String working = "create table "+tname+ "(id int, name varchar(23))";
```

```
Pstmt = con.prepareStatement(working);
```

PreparedStatement interface extends Statement

Use when column values are not hard coded but they are taken from variables

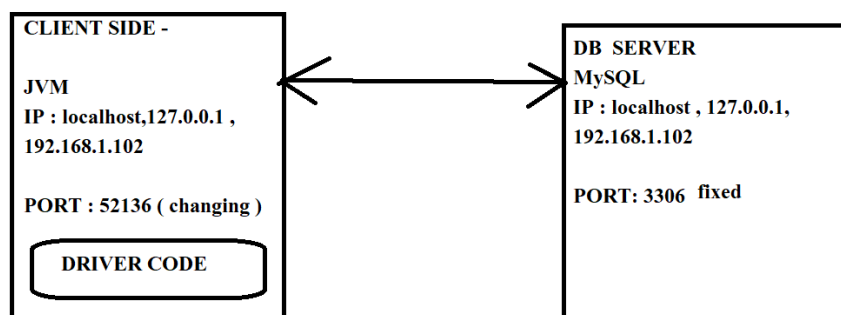
CallableStatement interface extends PreparedStatement

Use when we have to call stored procedures

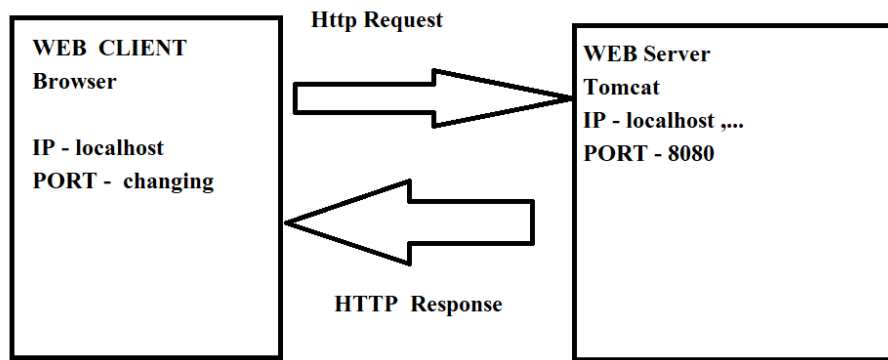
Jdbc Transaction ---- setAutocommit(false)

Commit, rollback

2-tier JDBC client application (SERVER type = DB Server)



JEE standard = Servlet -----for generating HTTP in Web Server !!!!



Web Server = **Http Server**

Examples --- **of Non JEE compliant !!!**

1. Apache Web Server
2. IIS web server Microsoft
3. WAMP/XAMP
4. Express
5.

JEE compliant Web Servers -----

Web Servers that have Java Web Containers that can manage Servlets!!

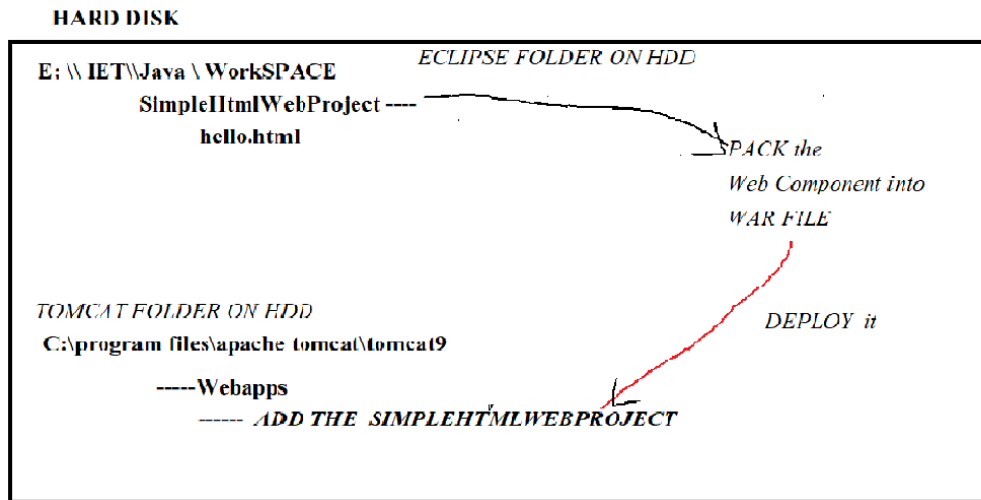
Example ----

1. Apache Tomcat
2. Jboss
3. Jetty
4. Glassfish
5. WildFly
6. IBM Websphere
7. Oracle Weblogic
8. Geronimo

-
1. Download and install Tomcat 9 !!!
 2. Check if JAVA_HOME = path upto jdk is present , if not then set it through environment variables!!!
 3. Start the tomcat server
 - Copy the Installation folder path upto bin
 - Open CMD
 - cd to above path
 - cd E:\Software\Install-Folder\tomcat\apache-tomcat-9.0.21\bin
 - Run the startup.bat on the prompt !!!
 - This should start your tomcat server .
 - You see the server window with this message once the server is started--
 - org.apache.catalina.startup.Catalina.start Server startup in [28,451] milliseconds
 4. Run the Client --- any browser
 - a. Type the following - URL window -----
 - Protocol : //host:port/service name /resource name

Create Web Application Using Eclipse !!!

Make sure your eclipse is JEE compliant



DEPLOYMENT ----- Submitting OUR COMPONENT to the WEB SERVER !!!!

---Copy the WAR file to tomcat webapps !!!

Deploying the web project on tomcat -----

Project - right click , EXPORT , WAR File , browse tomcat webapps and save it there!!!

If the tomcat server is running it will show the deployment message ---observe message.

GO to Browser -----

Access the hello.html

When html is accessed from Web server URL looks like this ----

<http://localhost:8080/SimpleHtmlWebProject/hello.html>

When html is accessed from File server URL looks like this -----

<file:///E:/batches/Core%20Java/workspace2/SimpleHtmlWebProject/WebContent/hello.html>

HW -----

Create the SimpleHtmlWebProject in Eclipse as discussed in class

Add the different HTMLs done in last week to that project's WebContent

Access those using web server url from browser.

Static Html = PRE Created html .

It cannot be changed .

Dynamic Html = HTML created ON THE FLY using latest data

Difficult to create too many static htmls on any website!!!

SimpleHtmlWebProject ----- Static Web project !!!

Compile time
Packaging time
Deployment time
Request time -----Html created on the fly

Servlet is a SERVER SIDE
JAVA
WEB
COMPONENT

This component is fixed into Java Web Container
Is inbuilt in
Java Web Server(Tomcat)

INTEGRATE Servlets with Tomcat Web Container

- JEE Servlet standards
Interface
1. Servlet interface
 2. ServletContext
 3. ServletConfig
 4. ServletRequest
 5. ServletResponse

Component 1 = Implement the interfaces
We implement Servlet interface
We call other interface method

Component 2 = calls the methods
Tomcat web container calls the servlet methods
Other interfaces are implemented by Tomcat

TASKS done by Servlet web component ----

1. It has the logic to Generate html on the fly (dynamic html)
 2. We can process data in the servlet(data processing)
 3. Session Management
-

Web.xml = DD = Deployment Descriptor
----- the file that **communicates** between
Programmer and tomcat web container

<http://localhost:8080/SimpleHtmlWebProject/hello.html>

<http://localhost:8080/SWP/>
This shows the fish.html

WHY ??

The tomcat CHECKED The welcome file list given by programmer

It found fish.html

Hence when no resource name is given still fish.html

Steps to write the first servlet

1. Create a dynamic web project
2. Also include web.xml
3. Add fish.html in webcontent
4. Add fish.html in welcome file list in web.xml
5. Deploy and test from browser as discussed in class
6. In eclipse src right click , new , servlet
7. You get a servlet class ---remove all unnecessary things as discussed in class
8. Add tomcat-installation/lib/servlet-api.jar this to build path
9. Make servlet and servlet-mapping entries in web.xml
10. Deploy the project
11. Run from browser

If u use <http://localhost:8080/SWP> you will see welcome page fish.html (static content)

If u use <http://localhost:8080/SWP/roes> you will see dynamic content generated from servlet
