**Things to remember while doing JSP Beans program, taglib and servlet programs. (Windows User)**

**JSP Beans –**

* Put your jsp file in the webapps folder of your Tomcat.
* Create a ‘WEB-INF’ folder in the folder where you store your JSP files usually the webapps folder, if the folder already exists then follow further steps.
* Create a ‘classes’ folder in the WEB-INF folder.
* Inside the classes folder create a folder with your package name (package name that you give in your program like for eg. Mypack).
* Store all your java files inside folder with your package name.
* You need to compile your java file before you run your jsp program in the browser.
* Remember, you need to just compile and not run this java file, running it doesn’t make sense because it doesn’t have a main function.
* Try these steps with programs – register.jsp and setUserPass.java, programs are there in the repo itself.

**Taglib –**

* Put your jsp file in the webapps folder of your Tomcat.
* Create a ‘WEB-INF’ folder in the folder where you store your JSP files usually the webapps folder, if the folder already exists then follow further steps.
* Create a ‘classes’ folder in the WEB-INF folder.
* Inside the classes folder create a folder with your package name (package name that you give in your program like for eg. Mypack).
* You need to have your tld file in the WEB-INF folder and java file inside the folder with your package name.
* You need to compile this java file, in order to do that you need to additionally use the jsp-api.jar file while compilation to do that use the following command while compiling the java file – javac -cp .;C:/xampp/tomcat/lib/jsp-api.jar hello.java
* While giving the jsp-api.jar file path check the path you’re giving after ; since I’m using Xampp I’ve that path, you’ll find all the jar files in your tomcat’s lib folder, you’re required to give that path according to your system.
* Remember, once again you’re not required to run the program after compiling, once this is done, go to your browser after starting the Tomcat ofcourse and run the jsp file.
* Use taglibeg.jsp, hello.tld and hello.java files while performing above steps.

**Servlet Programs –**

* Create a folder in your webapps folder and name it anything you want, for example servlet.
* Now, inside this folder create a WEB-INF folder.
* Inside WEB-INF folder create 2 folders named ‘classes’ and ‘lib’.
* You need to place your web.xml file in the WEB-INF folder.
* You need to place your java files(servlets) in the classes folder directly. You can also a create a package if you want but not necessary.
* You need to compile this java file using the following command - javac -cp .;C:/xampp/tomcat/lib/jsp-api.jar hello.java
* While giving the jsp-api.jar file path check the path you’re giving after ; since I’m using Xampp I’ve that path, you’ll find all the jar files in your tomcat’s lib folder, you’re required to give that path according to your system.
* If you’re using the database in the servlet then you need to keep a mysqlconnector file in the lib folder, maybe some of you won’t require this but still in case it shows error, do this step.
* Now, one thing you may wonder that we don’t have a jsp file here so what are we suppose to run in the browser, well good thinking. Remember, the web.xml file in the WEB-INF folder. It has the <url-pattern> tag, this tag is where you have a mentioned the name that you’re required to run in the browser, suppose this name is ‘genericserv’.
* Then you’ll be running the program in browser has -localhost:8080/servlet/genericserv
* Perform these steps with the genericserv.java and web.xml file.

**Some additional things to remember –**

* This may not be the most accurate description but here is an observation.
* Apache has a certain thing like cache so when you make a change run the program, it is possible that it may show you the same output has previous, this happens because it is not running the entire computation it displays the result from cache.
* So, whenever you make a change in your program, you’re suppose to restart your tomcat server.
* Well, it is kind of tedious and frustrating but it is what it is.
* Happy Coding!