James Quach

Elizabeth O’neil

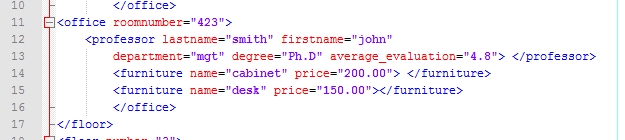
CS436

9/28/2016

Homework 2

1. XML. Read the tutorial linked from the class web page. Find the error in $cs636/campus.xml and describe it. Hint: try to display it in a browser. Fix the error and show a snippet of XML around your fix in your homework paper.

It is said to be “error on line 8 at column 148: Opening and ending tag mismatch: furniture line 0 and office”



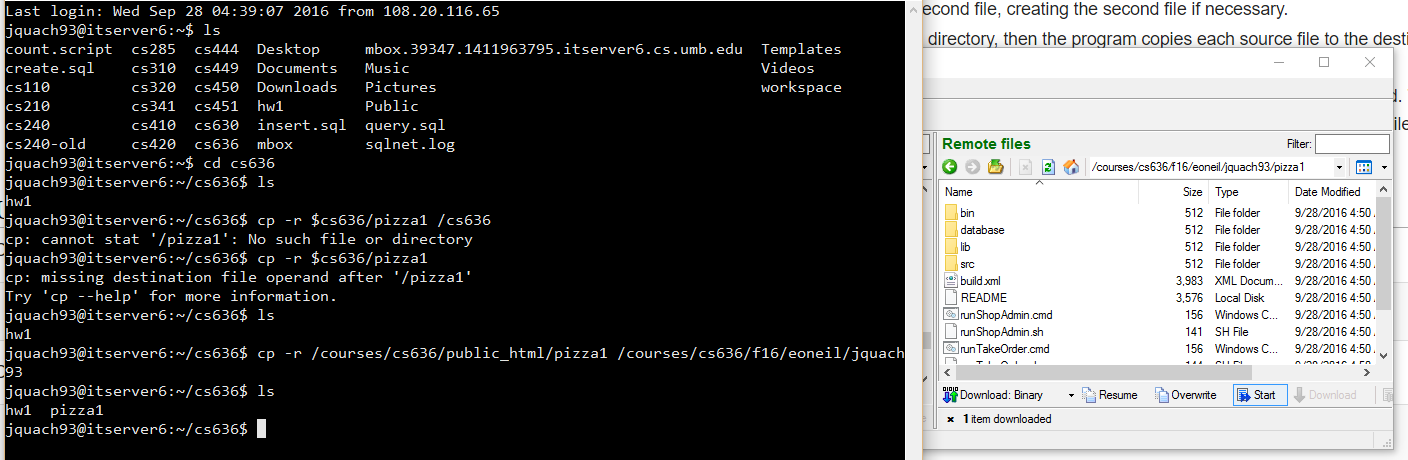


2. Command line development, environment variables, UNIX case. Do this before problem 5. This problem uses ant as a "black box."

a. Set up your UNIX/Linux account at cs.umb.edu as specified in DatabaseSetup, which also covers setting up environment variable definitions (both at cs.umb.edu and on your home machine) for ORACLE\_USER, ORACLE\_PW, ORACLE\_SITE, MYSQL\_USER, MYSQL\_PW, and MYSQL\_SITE, and also cs636 for /courses/cs636/public\_html, the directory for our class home page.

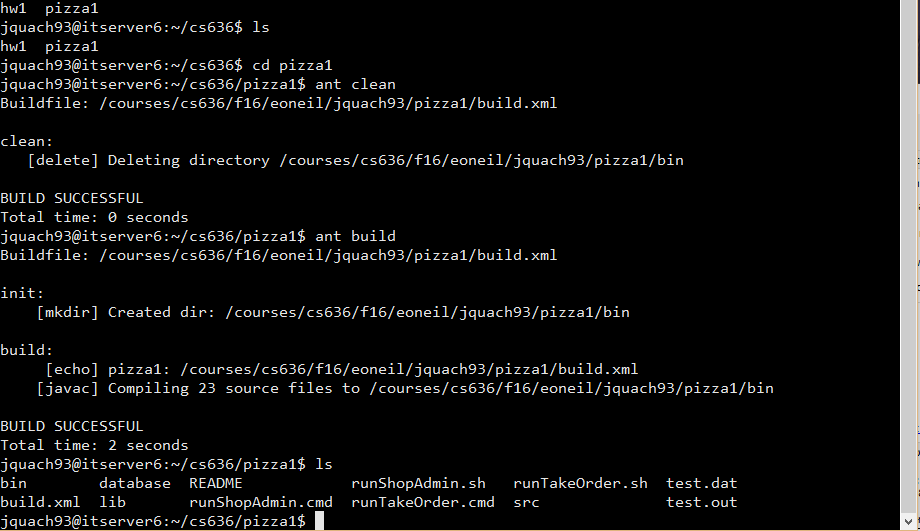
ssh -N -L1521:dbs3.cs.umb.edu:1521 -L3333:topcat.cs.umb.edu:3306 jquach93@topcat.cs.umb.edu

b. Give the UNIX/Linux command to recursively copy $cs636/pizza1 to your own pizza1 under your cs636 directory, when you are cd'd to your cs636 directory. Use $cs636 in the command. Do the copy.



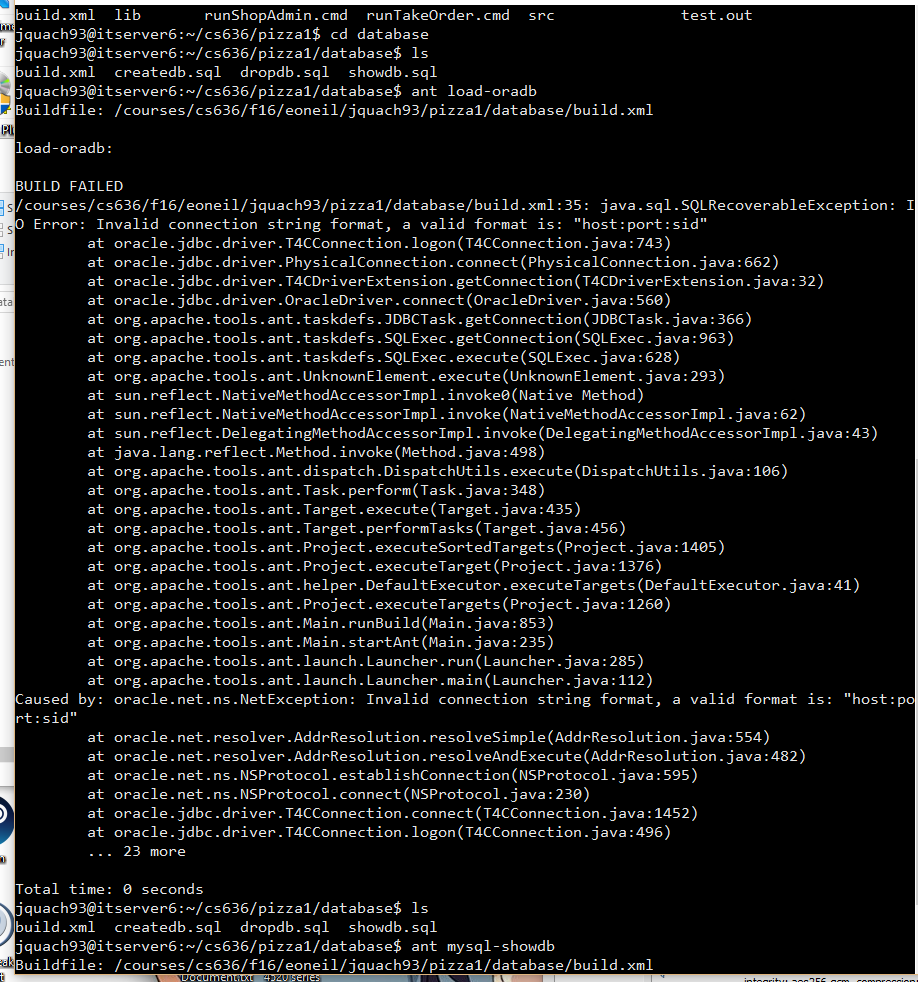
c. Download pizza1.zip to your home machine and unzip it there to a convenient directory. Show the whole directory structure of pizza1 after “ant clean” and then after "ant build" while cd'd to your copy of pizza1, both on cs.umb and at home. I'd use du for this on UNIX/Linux, tree on PC.

Unix/cs.umb:



Home://not successful

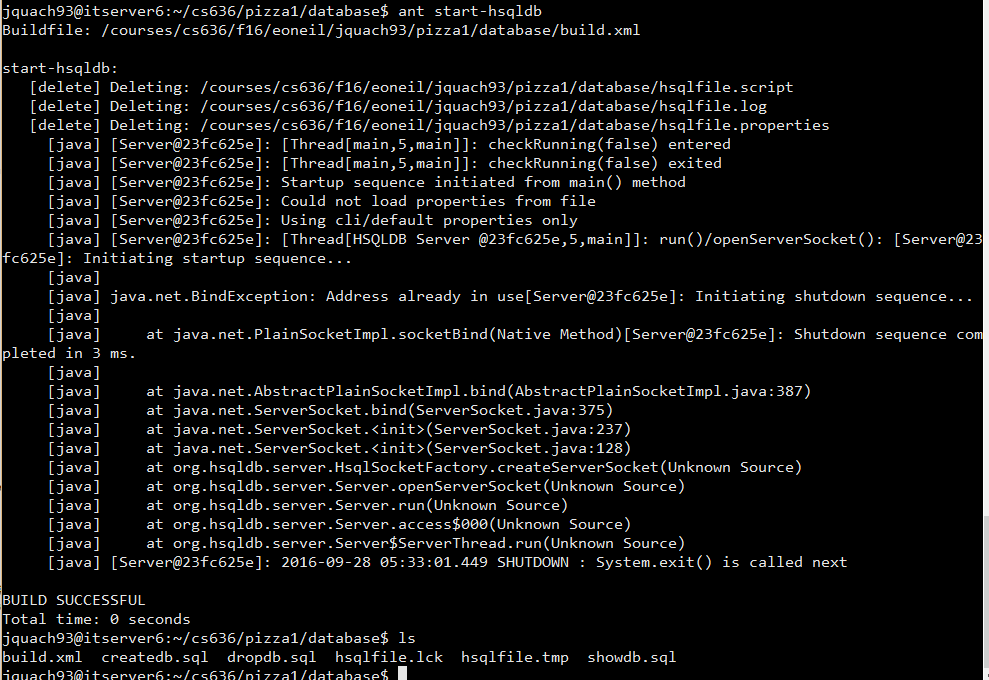
d. Cd to the pizza1/database directory (on either platform) and use “ant load-oradb” to load the Oracle tables, and “ant show-oradb” to display them. Note that you do not need to be logged into topcat to do this. You can do it anywhere connected to cs.umb.edu by the Internet, with a running tunnel if off-campus. Similarly load and show mysql, after looking at build.xml just to see the target names (grep target build.xml on UNIX, find “target” build.xml on Windows). Report success or any problems.



e. After loading the database, try "ant oraSysTest" to see it run the pizza1 system test using the ORACLE\_ environment variables you set up. Also try "ant mysqlSysTest". Report success or any problems.

Can’t succeed in part d thus cannot do part e.

f. Startup HSQLDB by using "ant start-hsqldb" in the database directory. Load up its database and run its sys test. Bring it down again.



3. Review or learn about Java packages. There is a Java tutorial on packages. Your directory listing of the pizza project in problem 2 showed many subdirectories of src and bin. These are all related to the Java packages in use.

a. Find and quote the package directive in PizzaOrderDAO.java that places its code in its package. What is its directory under src? Under bin?



\CS636\pizza1\src\cs636\pizza\dao\PizzaOrderDAO.java

\CS636\pizza1\bin\cs636\pizza\dao\PizzaOrderDAO.class

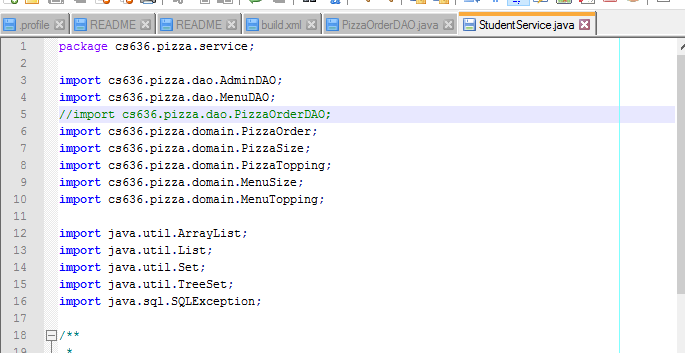
b. For StudentService.java, what is the package for this code? What methods of PizzaOrderDao are called from here? Find and quote the import statement that allows this StudentService code to access PizzaOrderDAO methods without long names.

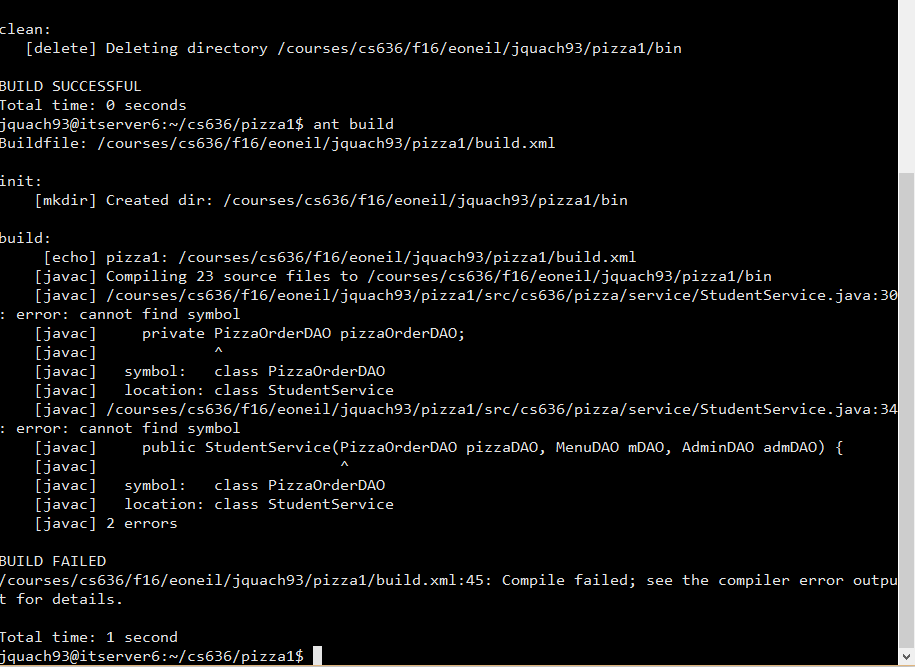






c. Try removing the import statement you found in b. and trying to compile the project (ant build). Report the resulting error.





d. What longer name could be used by StudentService to access the PizzaOrderDAO type without the import? Show the line of code that changes.



4. Start learning ant. Read the tutorial linked from the class web page..

a. Find and read the details on the ant delete task at the site where the tutorial resides. What does the line <delete dir="build"/> do when executed on UNIX/Linux? "rmdir build"? "rm -r build"? and from what directory?

“<delete dir=”build”/> would delete the respective file in the current directory.

“rmdir” will remove an empty directory on a Unix.

“rm –r” removes a directory and all its contents recursively if it is not empty.

b. Modify the second example in the tutorial for "oata.HelloWorld" to be for the same java file (except for the package declaration) but now made to be in package "com.oata", following the usual convention that the package name is the site name in reverse order. (Only one tiny change is needed, showing the ease of refactoring this way with ant.) What do you need to do with the source file other than change the package name (i.e. what change in its filesystem location)?

package com.oata;

public class HelloWorld {

public static void main(String[] args) {

System.out.println("Hello World");

}

}

What also changes is that ‘oata’ will be a subfolder of ‘com’ I suppose (not too sure of the question).

c. Read about the sql task used in database/build.xml and compose a target that executes “select \* from pizza\_orders” where this select statement is in the target, not in a separate file.

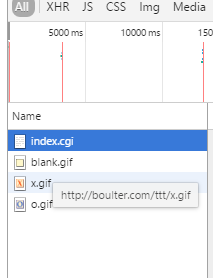
<target name=”select\*from pizza\_orders” description=”Select from Pizza Orders”> </target>

?

5. HTML forms. Read Murach on forms, pp. 114-123. Study this forms tutorial at http://www.w3schools.com (stop before the HTML5 Attributes part of the HTML Input Attributes chapter) and their tutorial on images. Look up <input> in their HTML reference, and look at the src attribute documentation. Visit http://boulter.com/ttt for a tic-tac-toe game with HTML user interface.

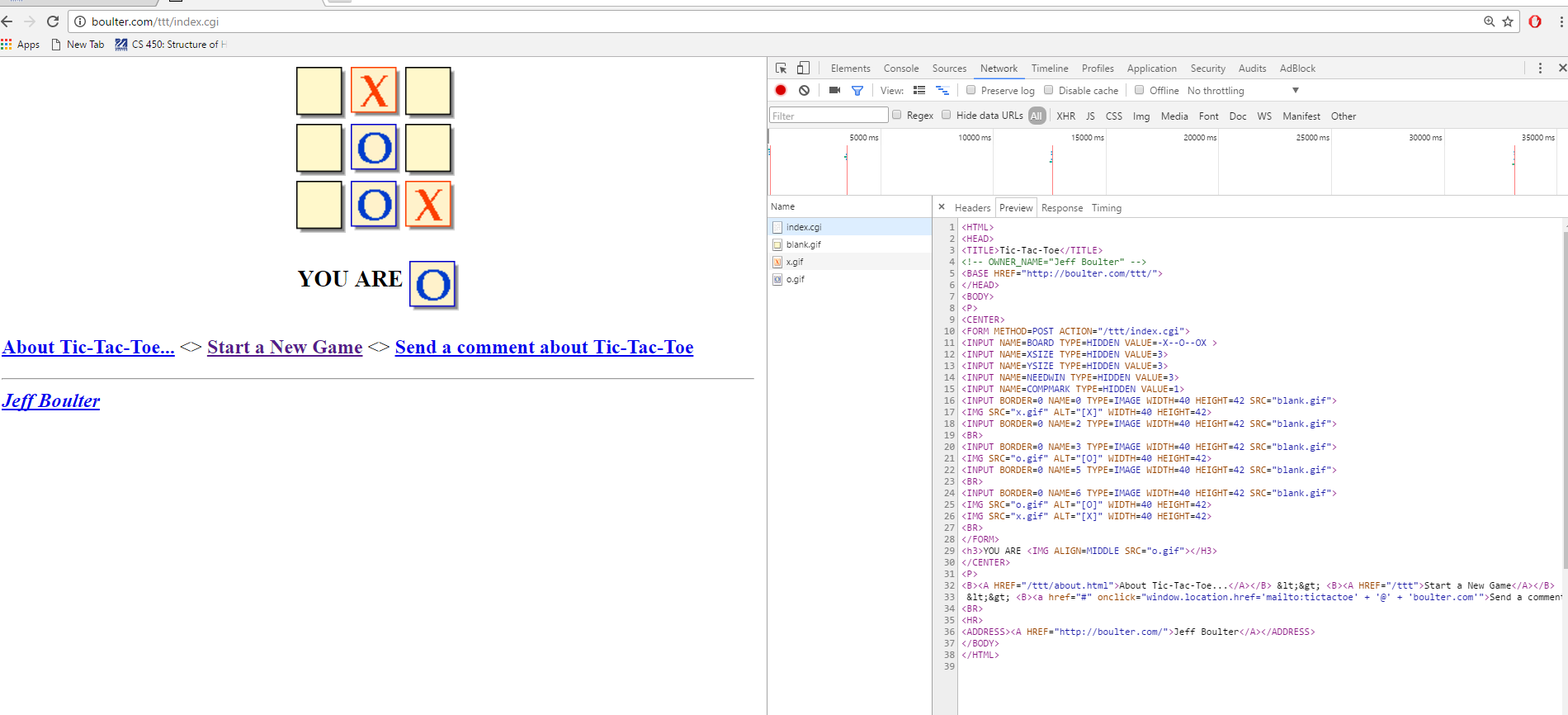
a. Start playing, and then look at the page source. Explain how the Xs and Os are displayed.

The X and O's are displayed as a form of a defined .gif image in .html code. X being x.gif and O being o.gif.



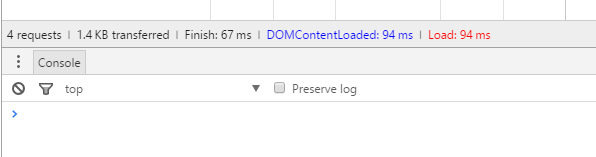
b. How does the server know which square you clicked on?

The server knows which square was clocked on due to the "<INPUT......>" of the respective square is defined in html. Then clocked in square’s blank.gif was replaced by a .gif of the respected O or X when clocked in as indicated by the <img src= ……>.



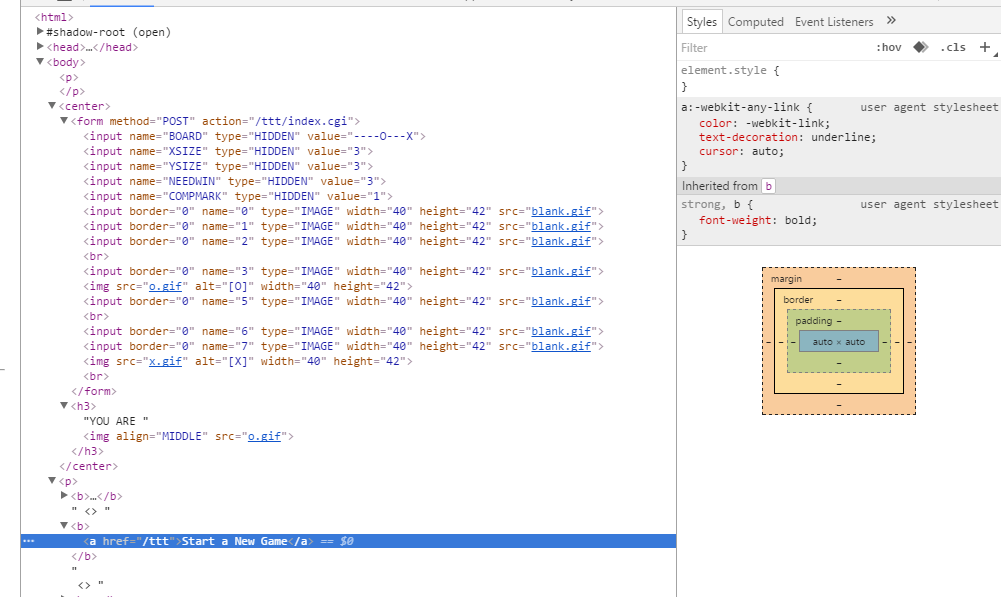
c. How many client-server round trips are needed to display this page?

There seems to be 4 server requests to display the page.



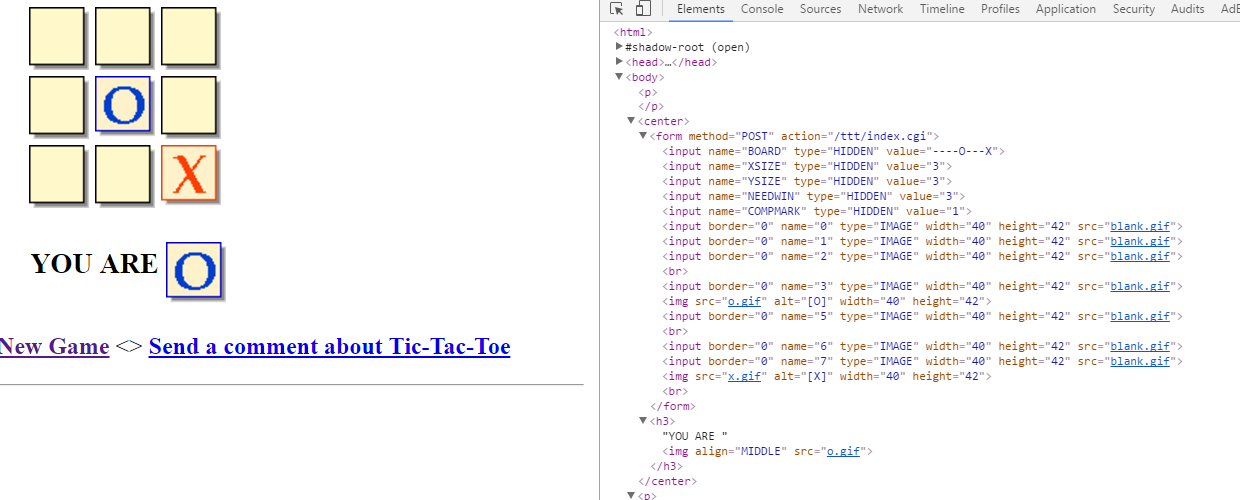
d. Observe how the cursor changes shape as you move the mouse around. What software is handling the moment-to-moment cursor painting? Look at the HTML source. How are the various areas for special cursors specified by the HTML? That is, what HTML elements correspond to what cursor shapes?

The software handling the changes of moment-to-moment cursor painting would be –webkit, the commands would be <input…> or <a…..>

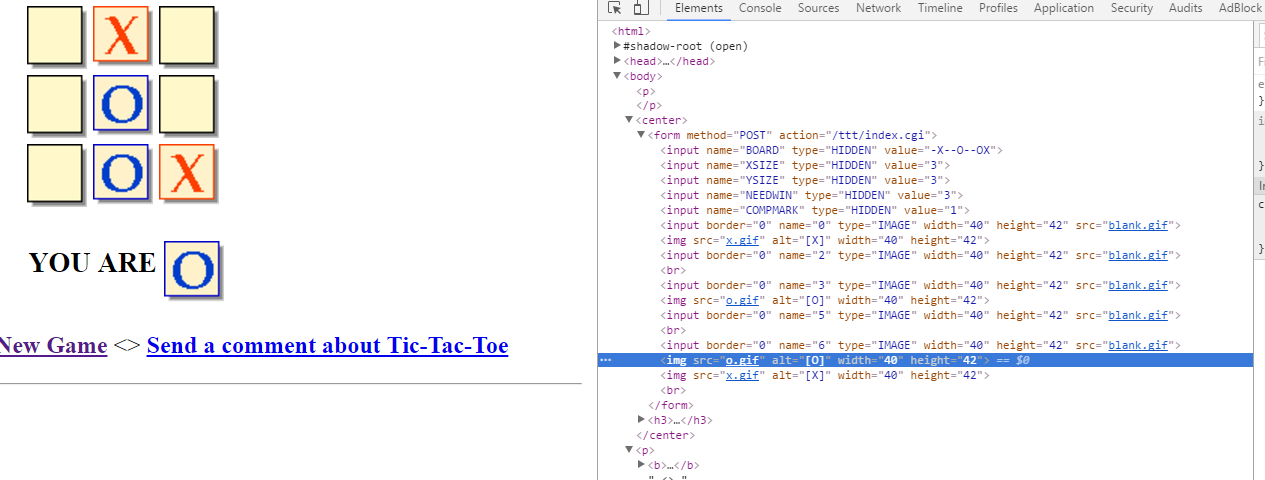


e. Do another move; note that the computer moves right away, but nothing ever happens after that until you do something. You can wait hours, and meanwhile hundreds of other users can be playing. Where is the knowledge of your current position (X's and O's) kept all this time? To answer this puzzle, look at the source, after a move and again after another move, and find out about hidden form fields.

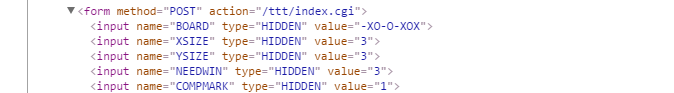
Before:



After:



The knowledge seems to be kept in one of the <input…….. type=”HIDDEN” value = “……..”> Where information of your current position is stored.



6. Use a plain editor like emacs or Wordpad to compose a web page register.html with page title "User Registration", contents entitled "Register for This Website", and a form, specifying name and email addresses in a text boxes, one above the other, with text to the left of each box "Name" and "Preferred Email", and a having a submit button. Include the text of register.html and its display from a browser in your homework submission. Note that we can't make this form work yet, just see it display itself. Don't use CSS here, so you can see the problem with alignment caused by the fact that we can't specify the width of the text even if we put it in a <span> or <label> because these elements can't have width specified. This is a well-known problem. See Duckett example. Later, with CSS, we should be able to fix this.

