1. A webpage on www.fropjop.com has a form that allows user input to enter data. Entering data results in a script being executed that depends on this data. The script loops forever (infinite loop) if the user provides bad input. Who gets harmed by the bad input: the user who entered the data, or the webserver owner of the www.fropjop.com, if the script is ...  

a. ... "client-side"? Circle one: user webserver owner  
b. ... "server-side"? Circle one: user webserver owner

2. Fill in the missing pieces of the HTML below, based on the following: You go to the page, enter John in the first box, enter Doe in the second box, then press the "Look Up Name" button, which results in the following URL being visited:


```html
<html>
  <head></head>
  <body>
    <form name="lookup" onsubmit="return false"
      action="http://rona.cs.usna.edu/gungnam.jsx">
      First Name: <input type="text" name=""/>
      Last Name: <input type="text" name=""/>
      <input type="button" onclick="submit()" value="Look Up Name">
    </form>
  </body>
</html>
```

3. Go to the URL: `http://rona.cs.usna.edu/~si110/lec/l13/stfin.html`, enter some numbers, and click the “process” button. Normally, you get a nice message that your input has been processed. It so happens that the server script that the form submits to will crash and cause a server error if it receives -999 for both input values. For that reason, stfin.html does some client-side input validation to ensure that it never submits -999 for both values. Find a way to submit -999 to this server side script none the less!

a. What is displayed in your browser when this happens.

b. Describe exactly what you did to send this bad input to the server despite stfin.html’s input validation.
4. Consider this HTML file:

```html
<html>
  <head></head>
  <body>
    <form name="login" onsubmit="return false" action="http://rona.cs.usna.edu/cgi/validate.jsx" method="get">
      Username: <input type="text" name="usrn" >
      Password: <input type="password" name="pswd">
      <input type="button" value="login" onclick="submit()">
    </form>
  </body>
</html>
```

a. You enter username "jonesy", the password "egbdf88", and press the "login" button. What URL gets visited as a result?

b. Even assuming there’s nobody and no camera looking over your shoulder, why is it that submitting a password to a website using forms and the "get" method like this puts your password in danger?

5. The URL http://rona.cs.usna.edu/~wcbrown/hwss.cgi is a server-side script. Unlike what we saw in class, rona won’t show you this script (you get an error message when you click on the above link), though it will let you execute it. The script expects a single input named alpha that, not surprisingly, should be your alpha code. If you call the script successfully, it will return a short message for you. Get the secret message for your alpha code!