Documentation for using domjudge for online verification

Step 1. Go to http://kumarslab.com/domjudge/public/

You get the following screen.

Step 2.

Click on login, then you get the following screen.

Step 3.

Register yourself with username as your roll number and some suitable password.
After login, you get the following screen.

In this screenshot, you see that a testuser2 (shown on extreme top right) is logged in. Instead of testuser, you should see your rollnumber. By clicking on the x beside your username will log you out.

Here you can click on the problems, to see the problems, you need to solve.

Step 4. Uploading your source code:

Never upload any executible or binary. You must only upload source code. For example, you may upload sum.py or sum.c or sum.cpp. Only Python (2, 3), C/C++ source files can be uploaded.

For trial, upload a C/C++/Python code that takes two numbers as input and produces the output. This program should read two numbers in a line and produce the result in next line in stdout. For all problems, specified sample input and outputs will be provided to you.
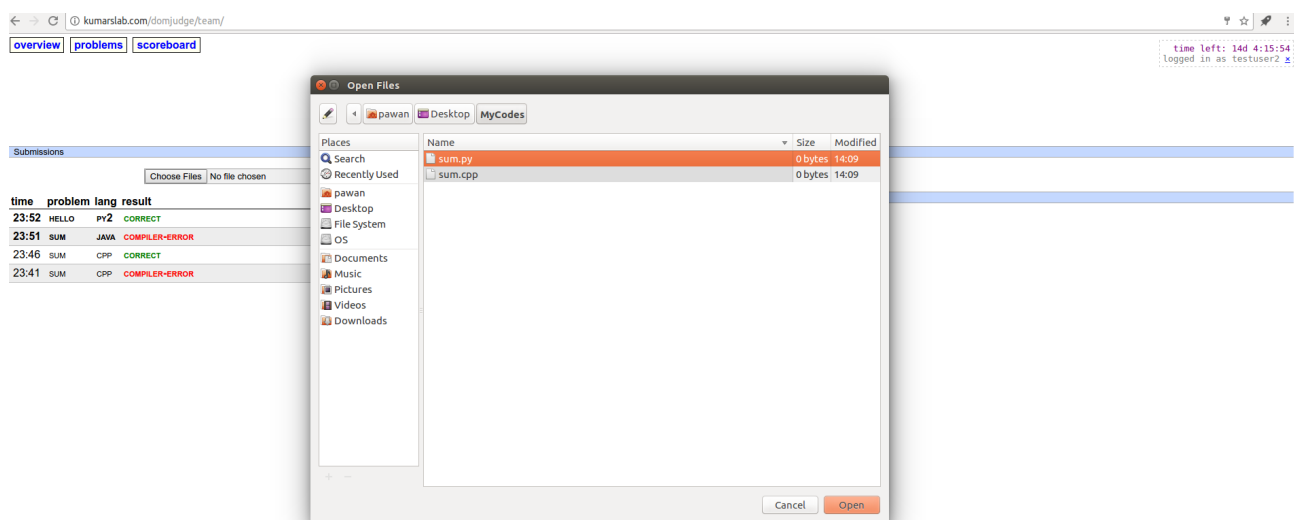
For example, for the sum problem:

23 34
57

Here 23 and 34 are inputs and 57 is the output.

After you have written a code to sum two numbers.

Step 5. Upload Sorce code for online judging

You select the file, in my case, I select sum.py. Select for which problem you are submiting your solution, also, select for which language your source code is written (C/C++/Python). In case of Python, you should select the appropriate version 2 or 3.

When you click on submit, you will be required to confirm, the screenshot is the following

Step 6. Waiting for confirmation of acceptance or rejection or error

Immediately after you submit, it will show the result as pending. After few seconds, it will tell you whether your code was accepted. If it is accepted it will show CORRECT, if not, then it will show other types of error. When you get error, click to see the error.

You can also ask for the clarification for the specific problem by clicking on request clarification.