# Sections

# **Preliminaries**

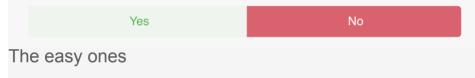
Let's do it. This is the scale of the first day of PHP. There is a lot of exercises, don't stay to long on each exercises else you'll never finish. Execute the simplest, fastest and most efficient tests. You should have executed a lot of tests for your own exercises just redo those. BE CAREFUL: this scale will not tell you what tests to do. Yes, some will maybe execute more(+) strange tests, that might end end break you programs (and stop the evaluation). Solution: multiply defense to avoid this problem. Edit: If it's not clear for everybody. As soon as an exercise is wrong, the evaluation stops the next exercises won't be graded. You can still discuss exercises but can't grade them. It's also explained in the 7th point of the subject.

#### The basics

- Is everybody present?
- Only grade the work that is in the student or group's GiT repository.
- -Chrome must be used for this defence.

If there is nothing in the repository put 0 and stop evaluation.

Discuss what went wrong (not necesseraly on the technical side) and how to avoid same situalion tomorrow.



The first exercises are rather simple...

HW

If the program displays Hello World, it's good.

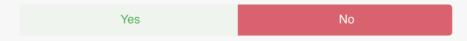
Else the evaluation stops.



### mIX

Execute Is to check that the php script iw under 100 characters (the whole file, everything included). Also check the number of X, especially the fact that there isn't 999 or 1001. It's a make-or-break.

(reminder: if it breaks the evaluation stops).



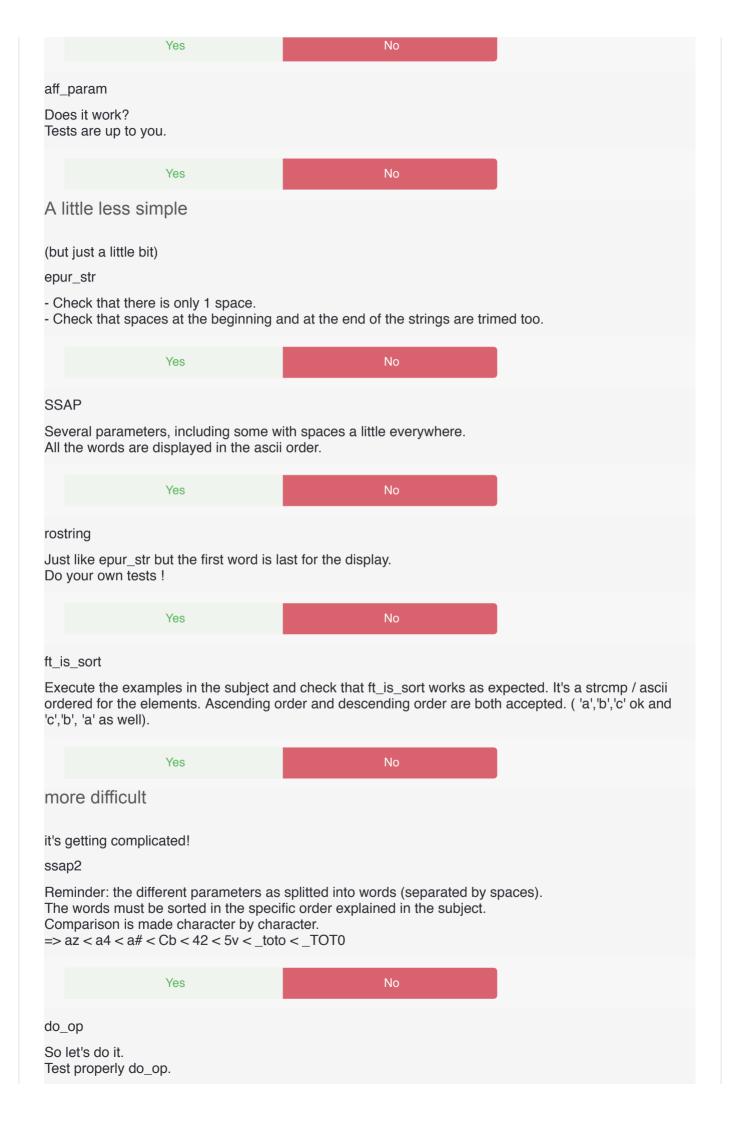
# To the divine

The program must ask for a number in the command line and say it's odd or even. - The result must be ok. - The message display must be accurate. - The characters other that numbers must be detected. - Test at least the examples given in the subject more are better. - No test with float numbers.



# ft\_split

Execute the examples in the subject and check that it works as expected. Execute extra tests.



No error management required except for the number of arguments. So you have to test with 3 arguments, put 3 correct arguments, eventually with spaces.

No division by 0.



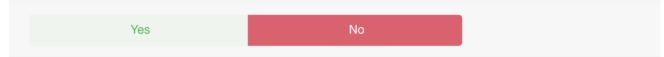
do op 2: the ++ version

this time just one argument. If there is more or not enough, the must be the same error message a last exercise.

Check that something that doesn't look like a calculation displays a syntax error.

If you test the last example (the one with rm -rf between backquotes), do it on the other student account. Not yours But it's totally up to you.

No division by 0.



search it!

Who seeks finds!

At least the subject example must be tested.

Don't hesitate to do more.



The hesitating agent.

More than just the examples in the subject you can also do the following tests:

\$> cat peer\_notes\_2.csv | ./agent\_stats.php moyenne 9.9787234042553

\$> cat peer\_notes\_2.csv | ./agent\_stats.php moyenne\_user

adam\_e:9.5263157894737

bertrand\_y:7.45 bruce\_w:12.45

clark\_k:11.416666666667 david\_a:8.5714285714286

dexter\_m:11.75

didier\_f:10.761904761905 francois\_h:10.045454545455 georges\_b:9.5263157894737 hans\_z:10.666666666667

[.....]

nicolas\_s:9.83333333333333 olivier\_z:8.4285714285714 roxane\_e:11.545454545455

sandra\_n:10.88

steve\_j:8.695652173913 trevor r:8.8571428571429

\$> cat peer\_notes\_2.csv | ./agent\_stats.php ecart\_moulinette

adam\_e:3.5263157894737

bertrand\_y:-12.55 bruce w:-0.55

clark\_k:-7.58333333333333 david\_a:-3.4285714285714 dexter\_m:-1.25
[...]
ludovic\_k:-3.4736842105263
maelle\_m:8.0454545454545
marcel\_j:10.95
marie\_c:4.5294117647059
mathieu\_c:1.9411764705882
melissa\_v:-8
michel\_r:5.4
nicolas\_s:-9.16666666667
olivier\_z:5.4285714285714
roxane\_e:6.5454545454545
sandra\_n:0.88
steve\_j:2.695652173913
trevor\_r:-2.1428571428571