**Problem**

Using object-oriented concepts, create a CLI tool that accepts a string and does the following to it:

- converts the string to uppercase and outputs it to stdout.

- converts the string to alternate upper and lower case and outputs it to stdout.

- creates a CSV file from the string by making each character a column in the CSV and then output "CSV created!" to stdout.

Make sure the code unit-testable and extendable. We would like to see a piece of code as you would do on your daily task.

**Sample Input**

hello world

**Sample Output**

HELLO WORLD

hElLo wOrLd

CSV created!

**Note:** a CSV file is created on the root directory which containing the following:

h,e,l,l,o, ,w,o,r,l,d

FAQ   
**1. What do you need to do?**

Write a program or application to solve the given problem. Prepare setup guide on how to execute the solution and unit test in the ReadMe file.

**2. How complicated is this assessment?**

Generally, this assessment can be solved within 4-8 hours for an experienced developer.

**3. When is the deadline to send the solution?**

Please complete and send us the solution for this problem within 1-week time. If you are on tight schedule and unable to make it, please inform us before the deadline.

**4. How to submit the solution?**

You may send us via Github, cloud storage like Google Drive, Dropbox, One Drive etc or email attachment.

**5. Any restriction on the programming language, tools, library, or framework?**

Please feel free to use any third-party library, framework, or tools. Our preferred language is PHP, but you may use any languages to showcase your problem-solving skills.

**6. What are we looking for in the solution?**

• Problem solving skills – Logical & critical thinking, the ability to apply the objectoriented concept or design pattern if needed.

• Testable code - the application of unit test and different test cases

• Maintainable code – easy to understand, extendable code, clean code with best practise implementation.

• Ability to handle unforeseen circumstance - exception handling, defensive programming

• Paying attentions to details

• Freedom to impress