



# Discovery Piscine

## Cell 3-2

*Summary: In this cell, we see how javascript works.*

*Version: 2*

# Contents

<b>I</b>	<b>A word about this Discovery Pool</b>	<b>2</b>
<b>II</b>	<b>Introduction</b>	<b>3</b>
<b>III</b>	<b>General instructions</b>	<b>4</b>
<b>IV</b>	<b>Exercise 02: It's over 9000</b>	<b>5</b>
<b>V</b>	<b>Presentación y evaluación por pares</b>	<b>6</b>

# Chapter I

## A word about this Discovery Pool

Welcome !

You will begin the first cell of this discovery pool of computer programming. We want to both show you what the code is that makes up the software you use every day, and at the same time experience peer-learning, an educational model of 42.

Programming involves logic (not math). It provides you with elementary bricks, which you assemble as you wish. There is never THE solution to a problem. There will be your solution, there will be those of each of your neighbors. Slow or fast, ugly or beautiful, if that gets the job done, that's all it takes! This assembly of bricks will constitute a series of orders (calculation, display, ...) that the computer will perform, in the order you have chosen.

Rather than giving you a course with only one solution for each problem, and which will probably be outdated in a few years, we have chosen to put you in a peer-learning situation. You are going to look for the elements that could serve you for your challenge, sort out those that are actually interesting by testing and manipulating them, and create your own program. To do this, discuss with others, exchange your points of view, find new ideas together, and finally test for yourself even to convince you that it works.

Peer-evaluation is a key moment to discover other ways of doing things, as well as special cases that you have not thought of and that could undermine your program (think about your degree of nervousness with software which crashes). Like different clients who don't pay attention to the same things, each reviewer will be different from the last. And who knows, you might have made new acquaintances for later collaborations.

At the end of this pool, you will not have done the same things as the other participants, you will not have validated the same projects, you will have chosen to do one challenge rather than another ... and that's normal! It's both a collective and a personal experience. Everyone will benefit from what he or she experiences during this time.

Good luck to all, we hope you will like this discovery.

# Chapter II

## Introduction

What this cell will make you see :

- Discovery of how a web page works.
- Discovery of the Javascript.

# Chapter III


## General instructions

Unless explicitly specified, the following rules will apply every day of this Piscine.

- This subject is the one and only trustable source. Don't trust any rumor.
- This subject can be updated up to one hour before the turn-in deadline.
- The assignments in a subject must be done in the given order. Later assignments won't be rated unless all the previous ones are perfectly executed.
- Be careful about the access rights of your files and folders.
- Your assignments will be evaluated by your Piscine peers.
- All shell assignments must run using `/bin/bash`.
- You must not leave in your turn-in your workspace any file other than the ones explicitly requested By the assignments.
- You have a question? Ask your left neighbor. Otherwise, try your luck with your right neighbor.
- Every technical answer you might need is available in the `man` or on the Internet.
- Remember to use the Slack workspace dedicated to your piscine!
- You must read the examples thoroughly. They can reveal requirements that are not obvious in the assignment's description.
- By Thor, by Odin! Use your brain!!!

# Chapter IV

## Exercise 02: It's over 9000

	Exercise 02
It's over 9000	
Turn-in directory : <i>ex02/</i>	
Files to turn in : <i>calc.html</i>	
Allowed functions : HTML, CSS, JS	

For this exercise, we will create a basic calculator. No library allowed. The design doesn't matter as long as the exercise is still doable.

First, let's model our calculator. It will be composed as follows:

- A **text input** field that represents the left member of our calculation.
- A **select** field that will contain a list of the following operators as options:('+', '-', '\*', '/', '%').
- A **text input** field that represents the right member of our calculation.
- A **submit input** of value 'Try me!'.

When we click on the button 'Try me!', the calculation is executed and the result appears in an alert message. The result also needs to be displayed on the console of your browser (log).

Both inputs fields can only contain positive integer values( $\geq 0$ ) for the calculation to be executed. Otherwise display an alert message must appear with the following message 'Error :('.

Division and modulo by zero display an alert message with the following message: "It's over 9000!". The result also needs to be displayed on the console of your browser (log)

Every 30 seconds, an alert pop-up windows will appear saying 'Please, use me...'.

# Chapter V

## Presentación y evaluación por pares

- En la carpeta `discovery_piscine` de la raíz de su home, cree una nueva carpeta `cell103` y navegue hasta ella.
- A partir de ahora, todos los ejercicios deben estar en la carpeta correcta de renderización. El ejercicio 00 en la carpeta `ex00`, el ejercicio 01 en la carpeta `ex01`, etc ... usted consigue la lógica.



Tenga en cuenta que durante su defensa todo lo que no esté presente en la carpeta del día no será revisado.