# Introduction

## Solving Coding Console Problems

While going through the program, you will find questions where you would have to write code to solve the questions. These questions are spread across the program and would help develop your problem solving and coding abilities. Hence, it is crucial for you to spend time answering these problems as these would help you get the best out of the UpGrad learning experience.

Now before you start solving the coding questions, it is necessary for you to go through the tutorial for the coding console that will be used for answering the questions on UpGrad platform. The videos below will help you understand how to use the UpGrad Coding Console.

First, let’s look at what the editor looks like and also learn about some relevant terms that you need to familiarise yourself with.

Here is a quick list of specific keywords along with their meaning

- **Problem Stub** – A problem stub (also known as 'language stub') is the skeleton code which the teacher has provided for a particular language; you have to solve the coding problem by adding your code to the given skeleton code.
- **Execution Time Limit** – Time-taken for the code to execute from start-to-finish for the given input on the server. Note: This time is independent of your internet speed. In case, your code exceeds the time limit for a particular input, you will receive an error message such as – "Time limit exceeded. The execution took more than 5.00s."
- **Load Language Stub** – This is the action which you can find in the overflow menu, on clicking this you will be able to go back to the problem stub. Note: Your current solution code will be replaced by the problem stub. This action will help you to clear your current solution code with the problem stub or the skeleton code.

## Demystifying Test Cases

Test cases are a tool to evaluate your code. A test case determines if the solution code (code written by you) works for a particular scenario or not. In most questions, your code is checked against multiple test cases to check if it meets the requirements specified by the teacher or not.

A test case has two major components – an input, i.e. the scenario which your code is being tested for, and an expected output, i.e. the expected results for that particular input. If your code is returning the same output as the expected output given a specific input, then the test case is deemed as "Passed". Otherwise, it is deemed as "Failed".

Let us consider a problem we saw in the above video, an example of a test case for this problem would be something like this:

```
Input :
2
87 98
87 89 67 56

Expected Output:
2
4
```

In this example, the test case includes the input of two arrays with lengths 2 and 4 respectively. If the output of your code for these test cases matches the expected output which is 2 and 4, then your code will pass the test case.