

Tinkiet Internship - Technical Test

Exercise 1: Algorithmic

Write a code in **two** programming languages that prints the numbers from 1 to 100, however, for every multiple of '3' it should print "Hello" instead of the number and for the multiples of '5' it prints "World" and for the multiples of '7' it prints "Yoo".

Example:

1, 2, Hello, 4, World, Hello, Yoo, 8, Hello Yoo, World, 11, Hello, 13, Yoo, Hello World, 16, 17, Hello, 19, World, Hello, Yoo, 22, 23, Hello, World, 26, Hello, Yoo, 29, Hello World, 31, 32, Hello, 34, World, Yoo, Hello, ...

Programming Language:

At least one of the programming languages should be chosen from the following list:

- Java
- JavaScript
- Python
- C++

Exercise 2: Logic

You are in the highway with your car (the green car), your friend calls you telling you that his car (a red car) has broken down, and you need to come and get him. However, there is a problem: you both don't know in which way he is (should you go straightforward or go back in the highway to find him).



Explain in the clearest way, how would you proceed to find your friend's car **in a finite amount of time**. Remember, all you can do is driving.

Assumptions & limitations:

- You can't leave the highway.
- You can't call your friend.
- The highway is bidirectional, meaning you can move with your car in any direction you want.

- The highway is infinite, meaning if you move in the same direction, you will never get to an end.
- Your car has infinite gas to run until you find your friend.

Exercise 3: Coding

We would like to get information/data (First name, Last name, Email, Description) from a job candidate. Your task is to create a web app frontend* (Example 1), where the candidate fills the form and clicks on the submit button then the data is sent over a REST API (POST Request) to the backend

Bonus question: add CV upload [file: PDF or DOCX] to the form.

The backend** exposes the resource "candidate" through a specific route as follows:

<http://localhost:3000/candidate>

When you receive the data in this resource, you need to store it as a JSON File (Example 2).

Bonus question: store the CV as file in the backend.

*frontend : you can use React or Vanilla JS (HTML, CSS, JS without framework) or Angular.

**backend: highly recommended to use JavaScript (Node.js), But you can use other Languages such as Python (flask, Django, FastAPI) or Java (Spring), PHP.

Example 1:

Example 2:

```
{
  "firstName": "Hamid",
```

```
"lastName": "Zahir",  
"Email": "hamid.zahir@gmail.com",  
"Description": "frontend developer."  
}
```