# Setup the following class

Employee (is a person class)

* Attributes (id, email, workMood, salary, isManager)
* Methods (work)

Office:

* Attributes (name, employees)
* Methods (getAllEmployees, getEmployee, fire,hire)

# Implement Employee methods

sleep(hours): Method in Person class (7→ happy, <7 → tired, >7 → lazy)

eat(meals): Method in Person class (3 meals →100 health rate, 2

meals→75 health rate , 1 meal→ 50 health rate)

buy(items): Method in Person class ( 1 Item→decrees Money 10 LE)

work(hours): Method in Employee class ( 8→ happy, > 8 →tired, > 8 → lazy)

• Salary: Property must be 1000 or more

• Health rate: Property must be between 0 and 100

# Implement Office methods

getAllEmployees(): Method in Office class get all current employees.

getEmployee(empId): Method in Office class get employee data of given

employee id, and if he is a manager display all info except salary.

hire(Employee): Method in Office class hires the given employee.

fire(empId): Method in Office class fires the given employee id.

Let the program be user prompt

Print a menu with the functionalities allowed.

For example:

For adding new employee enter “add”

If manager press “mngr”

if normal employee press “nrml”

Enter your data:

> Name:

> age:

The final menu option is “q” to quit the application.

Hint:

* To store employee, you can use array or array of object
* Emp Id: you can use email instead of generating random id

Part 2:

Report:

1. Abstract VS interface
2. Inheritance in function constructor

For yourself:

This article talking about classes

<https://everyday.codes/javascript/please-stop-using-classes-in-javascript/>

https://www.toptal.com/javascript/es6-class-chaos-keeps-js-developer-up