Homework

Question 1:

Answer below questions based on `Clock` constructor function(in class exercise) and `Clock` class.

> ➢ Also add a line of code that will stop the clock after 10 ticks
> ➢ What is the inner function of the constructor function?
> ➢ What is the local variable of the constructor function?
> ➢ What is the clock "interface" returned by the constructor function?
> ➢ What are the closures?
> ➢ What are the private variables and functions?
> ➢ What are the public methods?
> ➢ How does this example illustrate that a JavaScript class is really a function and not an object?

```
function Clock({ template }) {

  let timer;

  function render() {
    let date = new Date();

    let hours = date.getHours();
    if (hours < 10) hours = '0' + hours;

    let mins = date.getMinutes();
    if (mins < 10) mins = '0' + mins;

    let secs = date.getSeconds();
    if (secs < 10) secs = '0' + secs;

    let output = template
      .replace('h', hours)
      .replace('m', mins)
      .replace('s', secs);

    console.log(output);
  }

  this.stop = function () {
    clearInterval(timer);
  };
```

```
class Clock {
  constructor({ template }) {
    this.template = template;
  }

  render() {
    let date = new Date();

    let hours = date.getHours();
    if (hours < 10) hours = '0' + hours;

    let mins = date.getMinutes();
    if (mins < 10) mins = '0' + mins;

    let secs = date.getSeconds();
    if (secs < 10) secs = '0' + secs;

    let output = this.template
      .replace('h', hours)
      .replace('m', mins)
      .replace('s', secs);

    console.log(output);
  }

  stop() {
    clearInterval(this.timer);
```

```
    this.start = function () {
      render();
      timer = setInterval(render, 1000);
    };

}

let clock = new Clock({ template: 'h:m:s' });
clock.start();
```

```
    }

    start() {
      this.render();
      this.timer = setInterval(() => this.render(), 1000);
    }
  }
let clock = new Clock({ template: 'h:m:s' });
clock.start();
```

Question 2: Use class syntax to create LinkedList. Methods below:

1) add(value)

2) remove(value)

3) print()

You're allowed to add other methods if it's necessary.

After completion, we can call methods as below and see the results in console if call print().

```
linkedlist.add(1);
linkedlist.add(2);
linkedlist.add(3);
linkedlist.print(); //in the console, you should see: LinkedList{1,2,3}
linkedlist.remove(3);
linkedlist.print(); //in the console, you should see: LinkedList{1,3}
```

Question 3:

This is a quiz system which allows students to take quiz, get each student's quiz score and compute average score of students.

You need to use constructor functions or class syntax to implement Student, Question and Quiz

constructor function/class Student:

properties:

studentId: a unique student id

answers: holds an array that records the student's answers for the questions.

method: addAnswer(question) - add student's question(id, answer) to answers array.

constructor function/class Question:

properties:

qid: unique quesiton id

answer: hold quiz correct answer or student's answer

method: checkAnswer(answer) - used to check if student's answer is correct

constructor function/class Quiz:

properties:

questions: It's a Map which holds question id and correct answers. The key is question id, the value is the correct answer for this question

students: an array holds all students

methods:

- scoreStudent(sid), computes the quiz score for this student
- getAverageScore(), computes the average score over all students

After you complete the Question, Student and Quiz constructor functions, we may use the system as below:

Your system should return the correct result for getAverageScore() and scoreStudent(sid) as the expected result.

```
const student1 = new Student(10);
student1.addAnswer(new Question(2, 'a'));
student1.addAnswer(new Question(3, 'b'));
student1.addAnswer(new Question(1, 'b'));
const student2 = new Student(11);
student2.addAnswer(new Question(3, 'b'));
student2.addAnswer(new Question(2, 'a'));
```

```javascript
student2.addAnswer(new Question(1, 'd'));

const students = [student1, student2];

const questions =[new Question(1, 'b'), new Question(2, 'a'), new Question(3, 'b')];

const quiz = new Quiz(questions, students);

let scoreforStudent10 = quiz.scoreStudentBySid(10);

console.log(scoreforStudent10); //Expected Result: 3

let scoreforStudent11 = quiz.scoreStudentBySid(11);

console.log(scoreforStudent11); //Expected Result: 2

let average = quiz.getAverageScore();

console.log(average); //Expected Reuslt: 2.5
```