

TP05: TDD avec l'exercice `fizzbuzz`

La fonction `fizzbuzz` reçoit en paramètre un nombre entier et doit retourner un tableau de `string`, dont chaque éléments seront évalués selon les règles suivantes:

- Si le nombre peut être diviser par 3, alors l'élément sera `'Fizz'`.
- Si le nombre peut être diviser par 5, alors l'élément sera `'Buzz'`.
- Si le nombre peut être diviser par 3 et 5, alors l'élément sera `'FizzBuzz'`.
- Si le nombre ne répond à aucune des règles précédentes, alors l'élément sera le nombre lui-même.

Fait en `Typescript` et testé avec `Jest`

Code source de `fizzbuzz`

```
export default function fizzbuzz(n: number){
  let rep: string[] = [];

  for(let i:number = 1; i <= n; i++){

    if(((i % 3) === 0 ) && ((i % 5) === 0)){
      rep.push('FizzBuzz');
    }
    else if((i % 3) === 0){
      rep.push('Fizz');
    }
    else if((i % 5) === 0){
      rep.push('Buzz');
    }
    else{
      rep.push(i.toString());
    }
  }

  return rep;
}
```

Code du test de `fizzbuzz`

```
import fizzbuzz from "../fizzbuzz";

describe('FizzBuzz tdd test', () => {
  let expectedValuesNone: string[] = ["1"];
  let expectedValuesFizz: string[] = ["1", "2", "Fizz"];
  let expectedValuesBuzz: string[] = ["1", "2", "Fizz", "4", "Buzz"];
  let expectedValuesFizzBuzz: string[] =
    ["1", "2", "Fizz", "4", "Buzz",
     "Fizz", "7", "8", "Fizz", "Buzz",
     "11", "Fizz", "13", "14", "FizzBuzz"];

  it('Should return the number if its not divisible by 3 or 5', () => {
    expect(fizzbuzz(1)).toEqual(expectedValuesNone);
  });

  it('should return Fizz if the number is divisible by 3', () => {
    expect(fizzbuzz(3)).toEqual(expectedValuesFizz);
  });

  it('should return Buzz if the number is divisible by 5.', () => {
    expect(fizzbuzz(5)).toEqual(expectedValuesBuzz);
  });

  it('shoule return FizzBuzz if the number is divisible by 3 and 5.', () => {
    expect(fizzbuzz(15)).toEqual(expectedValuesFizzBuzz);
  });
});
```