**Practical task**

**1.** Протестувати додавання числа 0.1 до числа 0.2 із результатом 0.3.

**2.** Виконати тестування заданого масиву userList.

1. Протестувати чи є в масиві елемент “admin”
2. Протестувати чи перший елемент в масиві “Nick”
3. Протестувати чи останній елемент в масиві “new\_user\_2”
4. Протестувати твердження, що довжина масиву дорівнює 5
5. Протестувати твердження, що 3-ій елемент масиву має тип string
6. Протестувати твердження, що 8-го елементу масиву немає.

const userList = ['Nick', 'Kate', 'quest123', 'admin', 'new\_user\_2'];

**3.** Створити файл calcCircle.js в якому міститься 2 функції для розрахунку довжини кола getCircleLength(radius) і площі кола getCircleArea(radius). Виконати експорт функцій.

Створити файл calcCircle.test.js в якому потрібно протестувати задані функції.

1. Протестувати функцію getCircleLength() з аргументом 22, перевірити твержєення, що довжина кола дорівнює 138.2 при розрахунку з точністю до 1-го знаку після коми.
2. Протестувати функцію getCircleArea() з аргументом 9, перевірити твержєення, що приблизна площа кола дорівнює 254.47.
3. Протествувати обидві функції без переданих аргументів.

function getCircleLength(radius) {

return 2 \* Math.PI \* radius;

}

function getCircleArea(radius) {

return Math.PI \* radius \*\* 2;

}

**4.** Створити файл food.js в якому міститься функція filterFoodPrice(food, min, max), яка фільтрує масив продуктів по ціні. Причому food – це масив об’єктів з типом і ціною продукту, min - це встановлена мінімальна ціна продукту, max - це встановлена максимальна ціна продукту Виконати експорт функції.

function filterFoodPrice(food, min, max) {

return food.filter(element => element.price >= min && element.price <= max);

}

const food = [

{ kind: 'potato', price: 10 },

{ kind: 'bred', price: 16 },

{ kind: 'pepper', price: 27 },

{ kind: 'banana', price: 32 },

{ kind: 'lemon', price: 50 }

];

Створити файл food.test.js в якому потрібно протестувати функцію для заданого масиву food з аргументами min = 12, max = 40.

1. Протестувати чи очікувана довжина відсортованого масиву 3, відповідає реальній.
2. Протестувати чи відсортований масив містить об’єкт { kind: 'pepper', price: 27 }.
3. Протестувати чи відсортований масив містить 2, 3 і 4 елементи вихідного масиву food.
4. Протестувати чи властивість price 1-го елемента відсортованого масиву є більшою значення min.
5. Протестувати чи властивість price 3-го елемента відсортованого масиву є меншою значення max.
6. Протестувати твердження про те, що відсортований масив не містить елемента { kind: 'lemon', price: 50 }.
7. Запропонуйте і протестуйте щонайменше 2-3 своїх твердження для заданої функції і масиву food.