

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

Факультет інформаційних технологій  
Кафедра інженерії програмного забезпечення

Практичне заняття 4.

Робота з структурованими даними (об'єктами)

Мета роботи: закріпити знання про об'єкти користувача мови JavaScript;  
придбати практичні навички використання об'єктів, як структур даних.

Виконав: Луппа Олексій Андрійович

Група: ПД-21

Київ – 2018

Task 1:

```
function getCount(objects){  
    let cnt = 0;  
    for(let i = 0; i < objects.length; i++){  
        if(objects[i].x == objects[i].y)  
            cnt++;  
    }  
    return cnt;  
}
```

### Кількість об'єктів

Дописати у вікні нижче функцію, яка приймає на вхід масив об'єктів. Кожен об'єкт має два поля: *x* та *y*. Функція повинна

*Наприклад:*

Вхідні дані: [{x: 1, y: 1}, {x: 2, y: 3}, {x: 3, y: 3}, {x: 3, y: 4}, {x: 4, y: 5}]

Результат: 2

```
1 function getCount(objects){  
2     let count = 0;  
3     for(let i = 0; i < objects.length; i++){  
4         if(objects[i].x == objects[i].y)  
5             count++;  
6     }  
7     return count;  
8 }  
9
```

chrome 14

OK

OK

OK

Task 2:

```
function totalCost(goods, discounts) {  
    let sum = 0;  
    for(let i = 0; i < goods.length; i++){  
        let value = goods[i].value, amount = goods[i].amount;  
        let discount = 0;  
        for(let j = 0; j < discounts.length; j++){  
            if(goods[i].name == discounts[j].name){  
                discount = Number(discounts[j].discount.slice(0,2)) / 100;  
            }  
        }  
        sum += (value - value * discount) * amount;  
    }  
    return sum;  
}
```

```
1 function totalCost(goods, discounts) {  
2     let sum = 0;  
3     for(let i = 0; i < goods.length; i++){  
4         let value = goods[i].value, amount = goods[i].amount;  
5         let discount = 0;  
6         for(let j = 0; j < discounts.length; j++){  
7             if(goods[i].name == discounts[j].name){  
8                 discount = Number(discounts[j].discount.slice(0,2)) / 100;  
9             }  
10        }  
11        sum += (value - value * discount) * amount;  
12    }  
13    return sum;  
14 }
```

twilight 14

OK

OK

OK

OK

Task 3:

```
function validSequenceSimple(braces){  
  let l = 0, r = 0;  
  for(let i = 0; i < braces.length; i++){  
    if(braces.charAt(i) == '(')  
      l++;  
    if(braces.charAt(i) == ')')  
      r++;  
    if(l < r)  
      return false;  
  }  
  if(r == l)  
    return true;  
  else  
    return false;  
}
```

**Результат роботи:** true, якщо вхідний рядок містить правильну дужкову послідовність; або false, якщо послід закриваюча дужка слідує після відкриваючої.

```
1 function validSequenceSimple(braces){
2   let l = 0, r = 0;
3   for(let i = 0; i < braces.length; i++){
4     if(braces.charAt(i) == '(')
5       l++;
6     if(braces.charAt(i) == ')')
7       r++;
8     if(l < r)
9       return false;
10  }
11  if(r == l)
12    return true;
13  else
14    return false;
15 }
16
```

chrome ▾

14

Перевірити

OK

OK

OK

OK

OK

OK

OK

OK

OK

OK

OK

Task 4:

```
function validSequence(braces){
  let tracer = []
  for(let i=0;i < braces.length; i++){
    if ( braces[i] === "(" || braces[i] === "{" || braces[i] === "[" || braces[i] === "<")
      tracer.push(braces[i])
    else {
      if(tracer.length === 0)
```

```

    return false

    let lastValue = tracer[tracer.length-1]

    if( (braces[i] === '[' && lastValue === '[') || (braces[i] === ']'
        && lastValue === ']') || (braces[i] === '{' && lastValue === '{')
        || (braces[i] === '}' && lastValue === '}') || (braces[i] === '('
        && lastValue === '(') || (braces[i] === ')' && lastValue === ')')
        || (braces[i] === '>' && lastValue === '<'))

        tracer.pop()

    else

        break;

}

}

return tracer.length === 0

}

```

**Результат роботи:** true, якщо вхідний рядок містить правильну дужкову послідовність; або false, якщо послідовність є неправильною. Дужкова послідо-  
парі закриваюча дужка слідує після відкриваючої.

```

i 2   let tracer = []
i 3   for(let i=0; i < braces.length; i++){
i 4     if ( braces[i] === "(" || braces[i] === "{" || braces[i] === "[" || braces[i] === "<")
i 5       tracer.push(braces[i])
i 6     else {
i 7       if(tracer.length === 0)
i 8         return false
i 9       let lastValue = tracer[tracer.length-1]
i 10      if( (braces[i] === '[' && lastValue === '[') || (braces[i] === ']'
i 11              && lastValue === ']') || (braces[i] === '{' && lastValue === '{')
i 12              || (braces[i] === '}' && lastValue === '}') || (braces[i] === '('
i 13              && lastValue === '(') || (braces[i] === ')' && lastValue === ')')
i 14              || (braces[i] === '>' && lastValue === '<'))
i 15        tracer.pop()
i 16      else
i 17        break;
i 18    }
    }
    return tracer.length === 0

```

chrome 14

OK

OK

OK

OK

OK

OK

OK

OK

OK

OK

OK

OK

1. Об'єкт - це сукупність пов'язаних даних і/або функціональності (які зазвичай складаються з декількох змінних і функцій, які називаються властивостями та методами, якщо вони знаходяться всередині об'єктів).  
Давайте розберемо приклад, щоб показати, як вони виглядають.
2. 

```
var person = {  
  name: ['Bob', 'Smith'],  
  age: 32,  
  gender: 'male'  
}
```
3. Асоціативний масив — абстрактний тип даних, що дозволяє зберігати дані у вигляді набору пар ключ — значення та доступом до значень за їх ключем.
4. Властивості (параметрів) об'єкта — значення, які повністю описує об'єкт. Наприклад, об'єкт «учень» має такі властивості (подано невичерпний перелік): вік, зріст, маса тіла, успішність з певного предмету тощо.
5. 

```
var objectName = {  
  member1Name: member1Value,  
  member2Name: member2Value,  
  member3Name: member3Value  
}
```
6. `objectName.member1Name`
7. Посилання на об'єкт — це певний ключ, призначений для опосередкованого доступу до об'єкта. Може бути адреса об'єкта в пам'яті, чи будь-який неповторний ключ (рядок або число) в асоціативному масиві або що.
8. «Копіювання за значенням» - це створення такого самого об'єкта у новій комірці пам'яті.
9. «Копіювання за посиланням» - це створення посилання на вже існуючий об'єкт.
10. Вони копіюються за значенням.
11. `this` - це ключове слово, яке посилається на поточний об'єкт, всередині якого написаний код.