

Лабораторная работа № 6

Тема: Методы объектов.

Цель: Закрепить понимание различий между примитивами и объектами в JavaScript, изучить механизмы упаковки (boxing) и распаковки (unboxing), а также освоить переопределение стандартных методов объектов (toString(), valueOf()) для управления их поведением при преобразованиях..

◆ Вариант 1.

Задание 1: Реализация класса Person.

Создайте класс Person, который будет представлять человека с приватными полями name (имя) и age (возраст):

Реализуйте методы toString() и valueOf(), чтобы при выводе объекта Person можно было получать строку с информацией о человеке и его возраст в числовом формате.

Пример: `const person = new Person('Alice', 30); console.log(person);` // "Alice, 30 years old"

Задание 2: Упаковка и распаковка примитивов.

Создайте примитивный тип (например, число) и оберните его в объект:

Используйте упаковку для создания объекта Number.

Затем выполните распаковку, используя метод valueOf() и покажите результат.

Пример: `let num = new Number(42); console.log(num.valueOf());` // 42

Задание 3: Применение методов toString() и valueOf().

Создайте массив объектов Person и выполните следующие действия: Используйте метод toString() для вывода информации обо всех людях в массиве.

Используйте метод valueOf() для вычисления общей суммы возрастов всех людей в массиве.

Пример: `const people = [new Person('Alice', 30), new Person('Bob', 25)];`

Вариант 2.

Задание 1: Реализация класса Book.

Создайте класс Book, который будет представлять книгу с приватными полями title (название) и author (автор):

Реализуйте методы `toString()` и `valueOf()`, чтобы выводить информацию о книге и количество страниц в числовом формате.
Пример: `const book = new Book('1984', 'George Orwell', 328);`
`console.log(book);` // "1984 by George Orwell"

Задание 2: Упаковка и распаковка строк.

Создайте строку и оберните её в объект:

Используйте упаковку для создания объекта `String`.

Затем выполните распаковку, используя метод `valueOf()` и покажите результат.

Пример: `let str = new String('Hello'); console.log(str.valueOf());` // "Hello"

Задание 3: Применение методов `toString()` и `valueOf()`.

Создайте массив объектов `Book` и выполните следующие действия:

Используйте метод `toString()` для вывода информации о всех книгах в массиве.

Используйте метод `valueOf()` для вычисления общей суммы страниц всех книг в массиве.

Пример: `const books = [new Book('1984', 'George Orwell', 328), new Book('The Great Gatsby', 'F. Scott Fitzgerald', 180)];`

⚡ Отчет должен содержать (см. образец):

- номер и тему лабораторной работы;
- фамилию, номер группы студента и вариант задания;
- скриншоты окна VSC с исходным кодом программ;
- скриншоты с результатами выполнения программ;
- пояснения, если необходимо;
- выводы.

Отчеты в формате **pdf** отправлять на email:

colledge20education23@gmail.com

Шпаргалка по методам объектов в JavaScript.

1. Примитивы и объекты

Примитивы: строки, числа, булевы значения, `null`, `undefined`, `Symbol`, `BigInt`.

Объекты: сложные структуры данных, которые могут содержать примитивы и другие объекты.

```
let num = 42; // примитив
let obj = { value: 42 }; // объект
```

2. Упаковка (Boxing)

Упаковка — это процесс преобразования примитивного типа в объект.

Пример упаковки числа в объект Number

```
let num = 42;
let boxedNum = new Number(num);
console.log(boxedNum); // [Number: 42]
```

Пример упаковки строки в объект String

```
let str = "Hello";
let boxedStr = new String(str);
console.log(boxedStr); // [String: 'Hello']
```

3. Распаковка (Unboxing)

Распаковка — это процесс преобразования объекта обратно в примитивный тип.

Пример распаковки числа

```
let boxedNum = new Number(42);
let unboxedNum = boxedNum.valueOf();
console.log(unboxedNum); // 42
```

Пример распаковки строки

```
let boxedStr = new String("Hello");
let unboxedStr = boxedStr.valueOf();
console.log(unboxedStr); // "Hello"
```

4. Методы toString() и valueOf()

Метод toString()

Возвращает строковое представление объекта.

Пример переопределения toString()

```
class Person {
  constructor(name, age) {
    this.name = name;
    this.age = age;
  }
}
```

```

    toString() {
        return `${this.name}, ${this.age} years old`;
    }
}

let person = new Person('Alice', 30);
console.log(person.toString()); // "Alice, 30 years old"

```

Метод valueOf()

Возвращает примитивное значение объекта.

Пример переопределения valueOf()

```

class Book {
    constructor(title, pages) {
        this.title = title;
        this.pages = pages;
    }

    valueOf() {
        return this.pages;
    }
}

let book = new Book('1984', 328);
console.log(book.valueOf()); // 328

```

5. Применение методов в массивах

Использование toString() и valueOf() в массивах объектов

```

class Person {
    constructor(name, age) {
        this.name = name;
        this.age = age;
    }

    toString() {
        return `${this.name}, ${this.age}`;
    }

    valueOf() {
        return this.age;
    }
}

```

```
let people = [new Person('Alice', 30), new Person('Bob', 25)];

// Вывод информации о всех людях
people.forEach(person => {
    console.log(person.toString());
});

// Сумма возрастов
let totalAge = people.reduce((sum, person) => sum +
    person.valueOf(), 0);
console.log(`Общий возраст: ${totalAge}`); // 55
```