Федеральное государственное автономное

образовательное учреждение

Высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

|  |
| --- |
| Институт космических и информационных технологий |
| институт |
| Информатика |
| кафедра |

**ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ**

|  |
| --- |
| Работа с объектами JavaScript |
| тема |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Преподаватель | |  |  |  | Погребников А. К. |
|  | |  | подпись, дата |  | инициалы, фамилия |
| Студент | КИ21-16/1б, 032155832 |  |  |  | Н. А. Терентьев |
|  | номер группы, зачётной книжки |  | подпись, дата |  | инициалы, фамилия |

Красноярск 2023

**СОДЕРЖАНИЕ**

1 Цели 3

2 Задачи 3

3 Описание варианта задания 3

4 Ход выполнения 3

4.1 Несколько способов создания объектов 3

4.2 Конструктор для собственного объекта 5

4.3 Расширение встроенного типа функцией 7

5 Выводы 7

Список использованных источников 8

**1 Цели**

Изучить основы работы с объектами JavaScript. Получить навыки работы с объектами JavaScript.

**2 Задачи**

Для выполнения практической работы необходимо выполнить  
следующие задачи:

* создать несколько объектов;
* создать конструктор для собственного объекта;
* расширить функцией встроенный тип.

**3 Описание варианта задания**

Необходимо выполнить следующие действия.

1. Продемонстрировать несколько способов создания объектов, несколько способов доступа к их свойствам (на чтение и запись). Найти минимальный элемент в последовательности.
2. Создать конструктор для собственного объекта, добавить в него несколько свойств и методов, продемонстрировать работу с этими объектами.
3. Расширить встроенный тип (например, Array или Date) функцией, оперирующей данными расширяемого объекта (например, вывод среднего арифметического для массива, или подсчет количества секунд со дня вашего рождения для даты).

**4 Ход выполнения**

**4.1 Несколько способов создания объектов**

В JavaScript есть несколько способов создания объектов. На листинге 1 представлен весь необходимый код для выполнения этого пункта.

Листинг 1 – Создание объектов

// Несколько способов создания объектов

let student = {

age: 19,

name: "Никита",

surname: "Терентьев",

group: "КИ21-16/1б",

printInfo: function() {

console.log(`Студент: ${this.surname} ${this.name}, возраст: ${this.age} лет, группа: ${this.group}`);

}

}

student.printInfo()

function Student(age, name, surname, group) {

this.age = age;

this.name = name;

this.surname = surname;

this.group = group;

}

Student.prototype.printInfo = function() {

console.log(`Студент: ${this.surname} ${this.name}, возраст: ${this.age} лет, группа: ${this.group}`);

}

let newStudent = new Student(20, "Иван", "Иванов", "КИ21-17/1б");

newStudent.printInfo()

student.age = 20;

let newStudentSurname = newStudent.surname;

console.log(`Доступ на чтение: ${newStudentSurname}, доступ на запись: ${student.age} (было 19)`);

Работа кода представлена на рисунке 1.

****

Рисунок 1 – Доступ к свойствам и методам

**4.2 Конструктор для собственного объекта**

Создадим конструктор для собственного объекта и добавим в него несколько свойств и методов. Весь код представлен на листинге 2.

Листинг 2 – Собственный объект

// Создание конструктора для собственного объекта

function Circle(x, y, radius) {

this.\_centerX = x;

this.\_centerY = y;

this.\_radius = radius;

this.circumference = function() {

return 2 \* Math.PI \* radius;

}

this.areaOfCircle = function() {

return Math.PI \* Math.pow(radius, 2);

}

}

Object.defineProperty(Circle.prototype, 'centerX', {

get: function() {

return this.\_centerX;

},

set: function (newX) {

this.\_centerX = newX;

}

})

Object.defineProperty(Circle.prototype, 'centerY', {

get: function() {

return this.\_centerY;

},

set: function (newY) {

this.\_centerY = newY;

}

})

Object.defineProperty(Circle.prototype, 'radius', {

get: function() {

return this.\_radius;

},

set: function (newRadius) {

this.\_radius = newRadius;

}

})

Object.defineProperty(Circle.prototype, 'diameter', {

get: function() {

return this.\_radius \* 2;

}

})

let circle = new Circle(3, 4, 5);

console.log(circle.circumference());

console.log(circle.areaOfCircle());

console.log(circle.centerX);

console.log(circle.centerY);

console.log(circle.radius);

console.log(circle.diameter);

circle.radius = 7;

console.log(circle.radius);

console.log(circle.diameter);

Весь вывод представлен на рисунке 2.

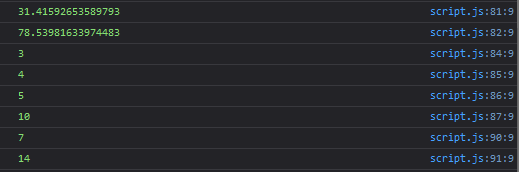
****

Рисунок 2 – Вывод

**4.3 Расширение встроенного типа функцией**

Напишем функцию для встроенного типа Array, которая будет вычислять среднее арифметическое для массива. Код представлен на листинге 3.

Листинг 3 – Произведение элементов

// Расширение встроенного типа

Array.prototype.average = function() {

let sum = 0;

for (let index = 0; index < this.length; index++) {

sum += this[index];

}

return sum / this.length;

}

let testArr = [10, 3, 1, 6, 2, 5, 4, 7, 8, 9]

let testAverage = testArr.average();

console.log(`Среднее арифметическое для массива: ${testAverage}`);

Работа функции представлена на рисунке 3.

****

Рисунок 3 – Работа функции

**5** **Выводы**

В ходе работы были изучены основы работы с объектами JavaScript.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1 СТО 4.2-07-2014 Система менеджмента качества. Общие требования  
к построению, изложении и оформлению документов учебной деятельности.  
Дата введения – 30.12.2-13.