**Задание №1. Встроенные типы объектов**

Создайте тип объектов Прямоугольник со свойствами координат верхнего левого (x1, y1) и нижнего правого (x2, y2) углов.

Переопределите методы toString() и valueOf(). Метод toString() должен возвращать строку с описанием текущего состояния. Метод valueOf() должен возвращать текущий периметр.

**Задание №2. Встроенные типы объектов**

Дана строка, которая содержит имена пользователей, разделенные запятой – "Login1,LOgin2,login3,loGin4". Разбейте эту строку на массив строк (чтобы отдельно были логины), и приведите их все в нижний регистр.

**Задание №3. Встроенные типы объектов**

Напишите программу, которая запрашивает у пользователя строки (prompt()). Ввод строк должен быть завершен, когда пользователь вводит строку end. Далее программа проверяет, есть ли в строке цифры, если да, добавляет в начало строки текст "Numbers: ", если нет – "No numbers: " и выводит результат в виде следующей таблицы:

+-----------------+---------------------+

| Исходная строка | Обработанная строка |

+-----------------+---------------------+

| строка\_1 | строка\_1\_обр |

+-----------------+---------------------+

| строка\_2 | строка\_2\_обр |

+-----------------+---------------------+

**Задание №4. Встроенные типы объектов**

Напишите функцию addTwoDays(*date*), которая принимает объект типа Date и добавляет к дате данного объекта два дня.

**Задание №5. Встроенные типы объектов**

1. Создайте массив с элементами "Джаз", "Блюз";

2. Добавьте в конец элемент "Рок-н-Ролл";

3. Замените предпоследний элемент с конца на "Классика". Код замены должен быть универсальный, т.е. работать для массивов любой длины;

4. Удалите первый элемент массива и выведите его на экран;

5. Добавьте в начало элементы "Рэп" и "Регги".

Массив в результате каждого шага:

1. ["Джаз", "Блюз"]

2. ["Джаз", "Блюз", "Рок-н-Ролл"]

3. ["Джаз", "Классика", "Рок-н-Ролл"]

4. ["Классика", "Рок-н-Ролл"]

5. ["Рэп", "Регги", "Классика", "Рок-н-Ролл"]

**Задание №6. Встроенные типы объектов**

Расширьте тип объектов Array методом sortDesc(), который сортирует массив по убыванию.

// Ваш код

var arr = [5, 1, 4, 2, 3];

arr.sortDesc();

console.log(arr); // [5, 4, 3, 2, 1]

**Задание №7. Встроенные типы объектов**

Напишите функцию deleteEvenNumbers(*array*), которая принимает массив *array*, удаляет из массива *array* четные числа и возвращает его.

function deleteEvenNumbers(array) {

// Ваш код

}

var arr = [1, 2, 3, 4, 5];

var arr = deleteEvenNumbers(arr);

console.log(arr); // [1, 3, 5]

**Задание №8. Встроенные типы объектов**

Есть следующий код:

var arr = ["Tom", "Sam", "Bob"];

var arrLength = [];

for (var i = 0; i < arr.length; i++) {

arrLength[i] = "<li>" + arr[i] + "</li>";

}

console.log(arrLength); // ["<li>Tom</li>", "<li>Sam</li>", "<li>Bob</li>"]

Перепишите данный код, заменив цикл for на метод map().

**Задание №9. Встроенные типы объектов**

Напишите функцию matrixToArray(*matrix*), которая принимает двумерный массив (матрицу) *matrix*, разворачивает двумерный массив в одномерный и возвращает его. Данное задание необходимо решить при помощи **аккумулирующих** методов.

function matrixToArray(matrix) {

// Ваш код

}

var arr = [[1, 2], [3, 4, 5], [6, 7, 8], [9]];

var arr = matrixToArray(arr);

console.log(arr); // [1, 2, 3, 4, 5, 6, 7, 8, 9]