

Задания

1. Создать функцию, принимающую в качестве параметра строку. Функция должна переводить первую букву строки в верхний регистр и возвращать измененную строку.

Пример работы:

```
alert(func("hello world")); // "Hello world"
```

2. Создать функцию, принимающую в качестве параметров 2 числа, задающих границы диапазона лет, внутри которого осуществляется поиск високосных годов. Функция должна вернуть **строку**, состоящую из високосных годов, попадающих в заданный диапазон.

Примечание: проверка делимости года на 4 без остатка – недостаточная.

<https://learn.microsoft.com/ru-ru/office/troubleshoot/excel/determine-a-leapyear>

Пример работы:

```
alert(func(2000, 2023)) // "2000 2004 2008 2012 2016 2020"
```

3. Создать функцию, принимающую в качестве параметров букву и строку и возвращающая количество этих букв, входящих в эту строку. Буква может быть записана в обоих регистрах.

Пример работы:

```
alert(func("a", "Amsterdam is the capital of the Netherlands")) // 5
```

4. Создать функцию, которая проверяет, является ли аргумент степенью числа **2**. Функция должна возвращать **true**, если является, и **false** – в противном случае. Например, число **16** является степенью числа **2**, т.к. $2^4 = 16$, функция должна вернуть **true**.

5. Создать функцию, которая возвращает сумму ряда до **n**-го члена (задан его в качестве параметра функции). Параметр может быть только натуральным числом.

Ряд: $1 + 1/4 + 1/7 + 1/10 + 1/13 + 1/16 + \dots$

Ответ округлить до **2** знаков после запятой.

Пример работы:

```
alert(func(1)); // 1.00
```

```
alert(func(2)); // 1 + 1/4 = 1.25
```

```
alert(func(5)) // 1 + 1/4 + 1/7 + 1/10 + 1/13 = 1.57
```

6. Создать функцию для проверки числа на то, является ли оно совершенным цифровым инвариантом. Если является, функция должна вернуть **true**, иначе **false**.

Совершенный цифровой инвариант – натуральное число, которое равно сумме своих цифр, возведённых в степень, равную количеству его цифр.

7. Создать функцию, принимающую массив имен и возвращающую строку по следующим правилам:

```
alert(func([])); // "no one likes this"
```

```
alert(func(["Adam"])); // "Adam likes this"
```

```
alert(func(["Bob", "Cindy"])); // "Bob and Cindy like this"
```

```
alert(func(["Adam", "Bob", "Cindy"])); // "Adam, Bob and Cindy like this"
```

```
alert(func(["Adam", "Bob", "Cindy", "Donald"])); // "Adam, Bob and 2 others like this"
```

Примечание: если в массиве 4 и более имен, часть строки **"and 2 others"** должна меняться в соответствии с количеством имен в массиве.

8. Создать функцию, принимающую в качестве параметра строку. Функция должна возвращать массив, состоящий из 2 «соседних» элементов строки. Если длина строки не является четным числом, функция должна добавлять к строке **"_"**.

Пример работы:

```
alert(func('qwe')) // ['qw', 'e_']
```

```
alert(func('qwer')) // ['qw', 'er']
```