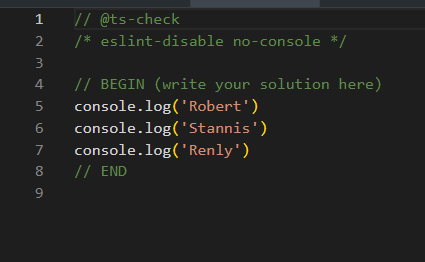
задание 1

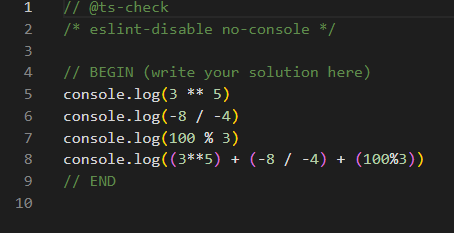
Выведите на экран друг за другом три имени: Robert, Stannis, Renly.



Задание 2

Напишите программу, которая считает и выводит на экран последовательно (по одному значению в каждой строке) значения следующих математических выражений:

* 3 в степени 5
* -8 разделить на -4
* остаток от деления 100 на 3
* сумму трёх предыдущих выражений

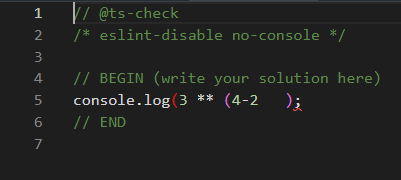


Задание 3

Проанализируйте этот код и скопируйте его в редактор:

console.log(3 \*\* (4 - 2));

Внесите в код такие изменения, которые приведут к синтаксической ошибке.

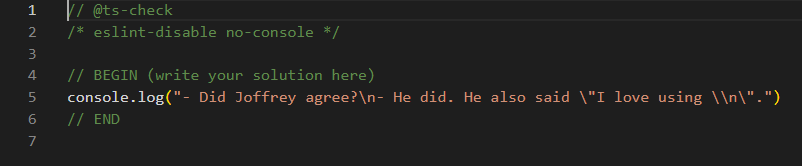


Задание 4

Напишите программу, которая выводит на экран:

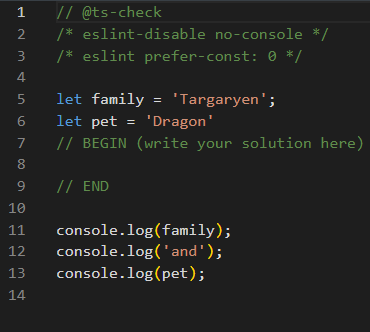
|  | - Did Joffrey agree? |
| --- | --- |
|  | - He did. He also said "I love using \n". |

При этом программа использует только один console.log(), но результат на экране должен выглядеть в точности, как показано выше.



Задание 5

Найдите в программе необъявленную переменную и объявите её, присвоив ей значение Dragon.



Задание 6

Напишите программу, которая берет исходное количество евро, записанное в константу eurosCount, переводит евро в доллары и выводит на экран. Затем полученное значение переводит в рубли и выводит на новой строчке.

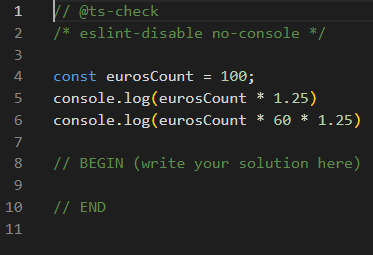
Пример вывода для 100 евро:

|  | 125 |
| --- | --- |
|  | 7500 |

Считаем, что:

1 евро = 1.25 долларов

1 доллар = 60 рублей



Задание 7

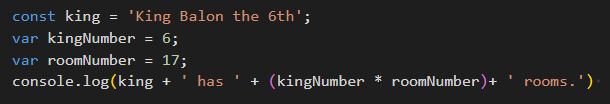
Вы столкнулись с таким кодом, который выводит на экран общее количество комнат во владении нынешнего короля:

|  | const king = 'King Balon the 6th'; |
| --- | --- |
|  | console.log(king + ' has ' + (6 \* 17) + ' rooms.'); |

Как видите, это магические числа: непонятно, что такое 6 и что такое 17. Можно догадаться, если знать историю королевской семьи: каждый новый король получает в наследство все замки от предков и строит новый замок — точную копию родительского. Эта странная династия просто плодит одинаковые замки…

Избавьтесь от магических чисел, создав новые переменные, а затем выведите текст на экран:

King Balon the 6th has 102 rooms.



Задание 8

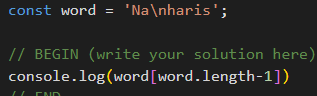
Выведите на экран строку Do you want to eat, <name>?, где вместо <name> должна использоваться константа stark. Вывод должен получиться таким:

Do you want to eat, Arya?



Задание 9

Выведите на экран последний символ строки, находящейся в константе word.



Задание 10

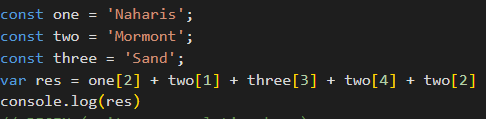
Выведите на экран число -0.304.



Задание 11

Вам даны три переменные с фамилиями разных людей. Составьте и выведите на экран слово из символов в таком порядке:

1. Третий символ из первой строки
2. Второй символ из второй строки
3. Четвертый символ из третьей строки
4. Пятый символ из второй строки
5. Третий символ из второй строки



Задание 12

Выведите на экран абсолютное значение числа, находящегося в константе fruitsCount, используя функцию [Math.abs()](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Math/abs). Эта функция возвращает абсолютное значение переданного числа. Абсолютное значения числа — это само это число без знака перед ним.



Задание 13

Теперь ваша очередь посмотреть на сигнатуру функции в документации и разобраться, как её использовать. Можете читать документацию на русском языке, но программист должен уметь читать документацию на английском. Используйте словари или переводчики при необходимости. Лучше сразу привыкать и подтягивать навыки чтения на английском, иначе будут сложности в будущем.

В Math есть функция ceil(). Изучите её [документацию](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Math/ceil).

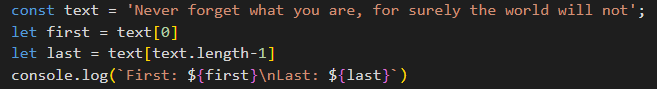
Напишите программу, которая использует функцию Math.ceil() с константой number и выводит результат на экран.



Задание 14

Выведите на экран первую и последнюю буквы предложения, записанного в константу text, в следующем формате:

|  | First: N |
| --- | --- |
|  | Last: t |



Задание 15

Посчитайте программно (а не в голове) минимальное число среди 3, 10, 22, -3, 0 — и выведите его на экран. Воспользуйтесь функцией Math.min(), которая работает аналогично Math.max().



Задание 16

Функция Math.random() возвращает случайное число от 0 до 1 с большим количеством знаков после запятой. Но в реальных задачах бывает нужно получать случайные целые числа, например, в диапазоне от 0 до 10 включительно. Реализуйте код, который выводит на экран именно такое число. Для этой задачи вам понадобятся функции [Math.random()](https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Math/random) и [Math.floor()](https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Math/floor)



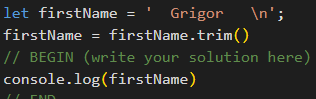
Задание 17

Выведите на экран тип значения константы motto.



Задание 18

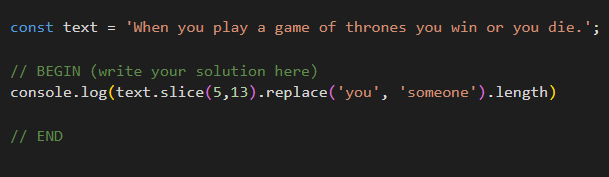
Обновите переменную firstName записав в неё то же самое значение, но обработанное методом .trim(). Распечатайте то, что получилось, на экран.



Задание 19

С помощью метода .slice() получите часть предложения, записанного в константу text, c 6 по 13 символы включительно. В полученной подстроке с помощью метода .replace() замените слово you на someone и выведите на экран длину итоговой подстроки. Выполните эти методы подряд в цепочке без создания промежуточных переменных.

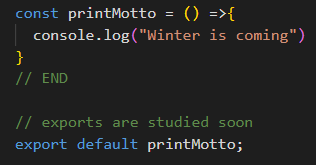
* [slice()](https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/String/slice)
* [replace()](https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/String/replace)



Задание 20

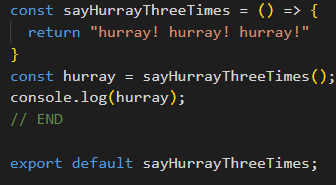
Реализуйте функцию с именем printMotto(), которая выведет на экран фразу Winter is coming.

printMotto(); // => "Winter is coming"



Задание 21

Реализуйте функцию sayHurrayThreeTimes(), которая возвращает строку 'hurray! hurray! hurray!'.

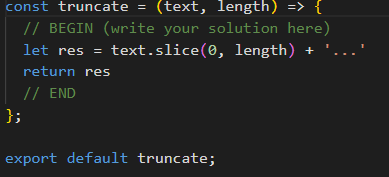


Задание 22

Реализуйте функцию truncate(), которая обрезает переданную строку до указанного количества символов, добавляет в конце многоточие и возвращает получившуюся строку. Подобная логика часто используется на сайтах, чтобы отобразить длинный текст в сокращенном виде.

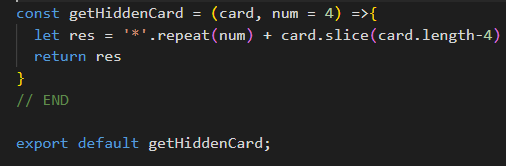
Функция принимает два параметра:

1. Строка, которую нужно обрезать
2. Число символов, которые нужно оставить



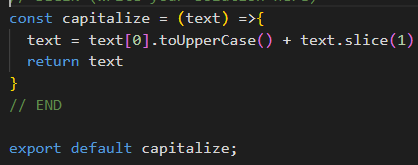
Задание 23

Реализуйте функцию getHiddenCard(), который принимает на вход номер кредитки (состоящий из 16 цифр) в виде строки и возвращает его скрытую версию, которая может использоваться на сайте для отображения. Если исходная карта имела номер 2034399002125581, то скрытая версия выглядит так \*\*\*\*5581. Другими словами, функция заменяет первые 12 символов, на звездочки. Количество звездочек регулируется вторым необязательным параметром. Значение по умолчанию — 4.



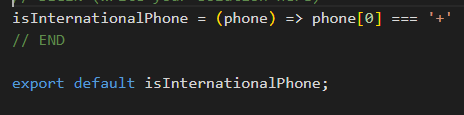
Задание 24

Реализуйте функцию capitalize(), которая принимает непустую строку и приводит первую букву строки к верхнему регистру:



Задание 25

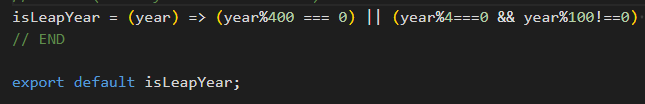
Реализуйте функцию isInternationalPhone(), которая проверяет формат указанного телефона. Если телефон начинается с +, значит это международный формат.



Задание 26

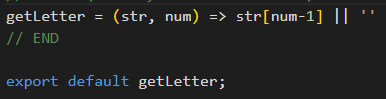
Реализуйте функцию isLeapYear(), которая определяет, является ли год високосным. Если год високосный, функция должна вернуть true, если нет – false. Год считается високосным, если он кратен (то есть делится без остатка) 400 или он одновременно кратен 4 и не кратен 100. Как видите, в определении уже заложена вся необходимая логика, осталось только переложить её на код:

|  | isLeapYear(2018); // false |
| --- | --- |
|  | isLeapYear(2017); // false |
|  | isLeapYear(2016); // true |



Задание 27

Реализуйте функцию getLetter(), которая извлекает из переданной строки указанный символ (по порядковому номеру, а не индексу) и возвращает его наружу. Если такого символа нет, то функция возвращает пустую строку.

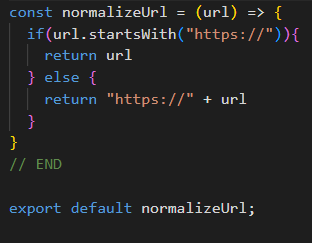


Задание 28

Реализуйте функцию normalizeUrl(), которая выполняет так называемую нормализацию данных. Она принимает адрес сайта и возвращает его с https:// в начале.

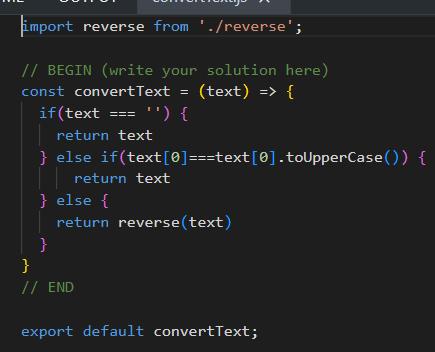
Функция принимает адреса в виде АДРЕС или https://АДРЕС, но всегда возвращает адрес в виде https://АДРЕС

Можно использовать метод [startsWith()](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String/startsWith), чтобы проверить, начинается ли строка с префикса https://. А потом на основе этого добавлять или не добавлять https://.



Задание 29

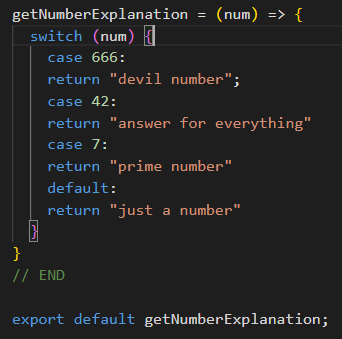
Реализуйте функцию convertText(), которая принимает на вход строку и, если первая буква не заглавная, возвращает перевернутый вариант исходной строки. Если первая буква заглавная, то строка возвращается без изменений. Если на вход передана пустая строка, функция должна вернуть пустую строку.



Задание 30

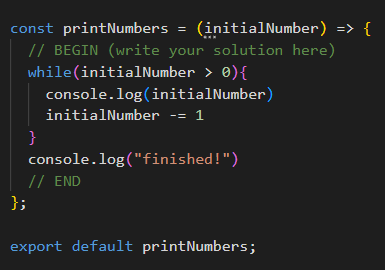
Реализуйте функцию getNumberExplanation(), которая принимает на вход число и возвращает объяснение этого числа. Если для числа нет объяснения, то возвращается just a number. Объяснения есть только для следующих чисел:

|  | \* 666 - devil number |
| --- | --- |
|  | \* 42 - answer for everything |
|  | \* 7 - prime number |



Задание 31

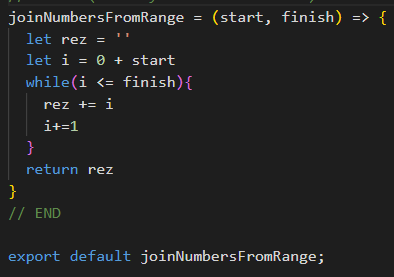
Модифицируйте функцию printNumbers() так, чтобы она выводила числа в обратном порядке. Для этого нужно идти от верхней границы к нижней. То есть счётчик должен быть инициализирован максимальным значением, а в теле цикла его нужно уменьшать до нижней границы.



Задание 32

Реализуйте функцию joinNumbersFromRange(), которая объединяет все числа из диапазона в строку и возвращает её:

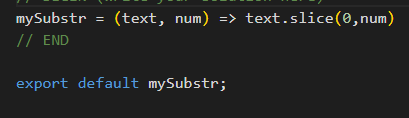
|  | joinNumbersFromRange(1, 1); // '1' |
| --- | --- |
|  | joinNumbersFromRange(2, 3); // '23' |
|  | joinNumbersFromRange(5, 10); // '5678910' |



Задание 33

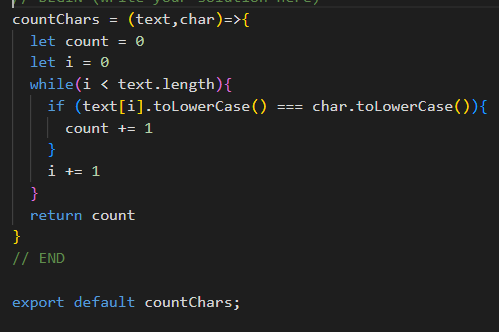
Реализуйте функцию mySubstr(), которая извлекает из строки подстроку указанной длины. Она принимает на вход два аргумента (строку и длину) и возвращает подстроку, начиная с первого символа:

|  | const text = 'If I look back I am lost'; |
| --- | --- |
|  | console.log(mySubstr(text, 1)); # => 'I' |
|  | console.log(mySubstr(text, 7)); # => 'If I lo' |



Задание 34

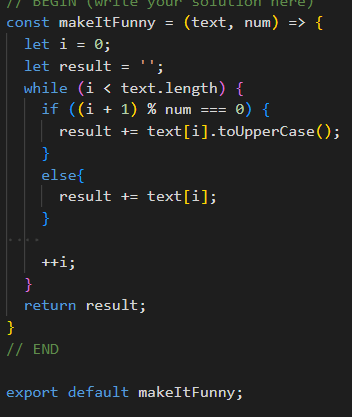
Функция countChars() из теории считает, сколько раз входит буква в предложение и при этом учитывает регистр букв. То есть A и a с её точки зрения разные символы. Реализуйте вариант этой же функции, так чтобы регистр букв был не важен



Задание 35

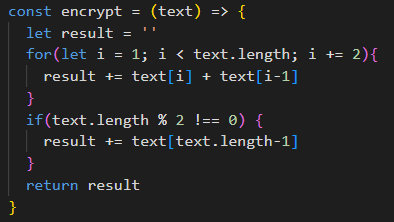
Напишите функцию makeItFunny(), которая принимает на вход строку и возвращает её копию, у которой каждый n-ный элемент переведен в верхний регистр. n – задается на входе в функцию.

Для определения каждого n-ного элемента понадобится остаток от деления %. Подумайте, как его можно использовать.



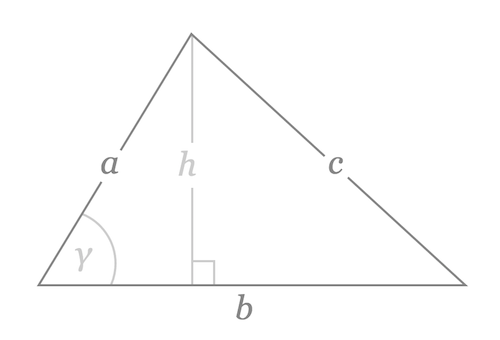
Задание 36

Сэмвелл обнаружил, что его сообщения перехватываются и читаются в замке «Близнецы», поэтому его атаки перестали быть внезапными. Немного подумав, он разработал программу, которая будет шифровать передаваемые сообщения по следующему алгоритму: программа получает на вход строку и меняет местами в ней каждые 2 подряд идущих символа. Если длина строки нечётная, то последний символ остаётся на своём месте.



Задание 37

В файле myMathModule.js:

Создайте функцию getTriangleArea(), которая принимает два аргумента h и b и вычисляет площадь треугольника по формуле A = 1/2 \* h \* b, где h — высота, а b — основание треугольника.  


В файле solution.js:

Импортируйте функцию getTriangleArea() из модуля myMathModule.

Создайте функцию, которая принимает аргумент n и возвращает площадь треугольника высотой n и основанием n2/2. Используйте функцию square() (принимает число и возвращает его квадрат).

Экспортируйте созданную функцию по умолчанию.

