**Git ошибка в написании имя**

**0**

[Drauzin](https://www.udemy.com/user/drauzin-anton/) ·

Лекция 8

 · 5 дней назад

Здравствуйте.  
Я создал глобально user.name и ошибся в написании имени.  
Как изменить

**5 ответов**

Следить за ответами

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель· 5 дней назад

**0**

Добрый день.

Если **Windows** - самый простой способ:

Необходимо зайти в:

Панель управления\Учетные записи пользователей и семейная безопасность\Диспетчер учетных данных

в разделе: "**общие учетные данные**" удалить учётку для git. Выглядит она так: git:https://github.com

При следующем взаимодействии в консоле, git попросит ввести логин и пароль в отдельном окне авторизации.

Если MacOS:

в папке пользователя найти скрытый файл и удалить его/поменять:

https://i.udemycdn.com/redactor/raw/2019-11-19_18-35-28-9f3886b1caf50b0a44dc387f4613dd08.png

**Когда нужны кавычки в [ ] при обращении к свойству?**

[Вадим](https://www.udemy.com/user/vadim-832/) ·

Задание 2

 · 2 дня назад

Иван, доброго времени суток!

Уточните, пожалуйста, при обращении к какому-либо свойству , в каких случаях в [ ] скобках мы ставим кавычки, а в каких нет? Спасибо.

Подписаться на ответы

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель · 2 дня назад

Добрый день. Кавычки мы ставим, когда свойство записано не латиницей или имеет сложную структуру (пробелы и тд):

obj["яблоко"]

obj["some text"]

etc...

--------------------------------------------------------------------------------------------------------------------------------------------

**Помогите пожалуйста**

[Руслан](https://www.udemy.com/user/ruslan-441/) ·

Лекция 12

 · 7 дней назад

Чем отличаются два случая, когда казалось бы в обоих есть глобальная переменная, в обоих через функцию ее меняют, но в первом случае глобальная переменная изменяется функцией и выводит в консоли(Вася, Петя, Петя), а во втором случае нет(Вася, Петя, Вася)?

Очень нуждаюсь в помощи, никак не могу победить логику такого поведения?!

Спасибо за любую помощь.

1. let userName = 'Вася';
3. function showMessage() {
4. userName = "Петя";
5. console.log(userName);
6. }
8. console.log(userName); // Вася
10. showMessage(); // Петя
12. console.log(userName); // Петя



17. let userName = "Вася";
19. function showMessage(userName) {
20. userName = "Петя";
21. console.log(userName);
22. }
24. console.log(userName); // Вася
26. showMessage(userName); // Петя
28. console.log(userName); // Вася

Подписаться на ответы

Игорь Федоров

[Игорь](https://www.udemy.com/user/igor-fedorov-5/) · 6 дней назад

В первой ситуации сначала идет выполнение первой консоли где выводит Васю в своей области видимости потом выполняется функция где тебя выводят Петю. После ты вызываешь консоль console.log(userName); где у тебя просто происходит сначала в функции перезапись Васи на Петю так как ты прописал не let userName = "Петя", что по сути и говорит после выполнения заменю Васю на Петю с функции в глобальную переменную.  
  
Во-второй ситуации совсем иначе работает все ты в функции объявил переменную showMessage(userName) что по сути не задевает друг друга и тут что выполняется в функции не имеет ни какого отношения к глобальной переменной потому при вызове консоли у тебя выводит Васю.

------------------------------------------------------------------------------------------------------------------------------------------

**Что значит () после названия функции?**

Это её вызов, призыв к действию. Что бы она, или метод объекта выполнился - нужно всегда ставить**()**

--------------------------------------------------------------------------------------------------------------------------------------

**ООП на практике**

[Даниил](https://www.udemy.com/user/daniil-244/) ·

Лекция 16

 · 5 дней назад

Доброго времени суток, Иван! Хочу вас поблагодарить, т.к. ваш курс дал мне крепкую базу основ JS, которая дала мне возможность попасть на стажировку в веб-контору. Но именно ООП стало камнем преткновения на моем пути становления фронтдэнд разработчиком! Дело в том, что я посмотрел все ваши упоминания о ООП, даже 1.5 часовое видео на youtube от Glo Academy и не нашел описания какого нибудь ярого примера ООП на практике. Одним из моих заданий для прохождения стажировки стал как раз пример реального Объектного Ориентирование, с которым я пока не могу справиться! Попрошу вас посмотреть это задание и возможно дать мне какие нибудь наставления или отметить моменты с которых стоит строить данную структуру! Задание :

В фирме есть директор, который отвечает за набор сотрудников и получение новых проектов. В фирме также есть 3 отдела: веб отдел, мобильный отдел и отдел тестирования в которых могут работать только соответственно веб разработчики, мобильные разработчики и QA специалисты. Каждый день директор может получить от нуля до 4 новых проектов одного из 2 типов (Веб/мобильный), каждый из которых может быть трех уровней сложности.

Полученные проекты директор пытается передать в отделы учитывая их специализацию. В случае если в отделе недостаточно работников, то отдел принимает только проекты на реализацию которых есть ресурсы, а оставшиеся проекты остаются у директора на следующий день.

На следующий день директор нанимает необходимое количество программистов для реализации вчерашних проектов и передает их по отделам. В случае если в отделе, в момент получения проектов от директора есть свободные программисты, то веб программисты получают по 1 проекту на реализацию, а мобильные могут работать на 1 проекте вдвоем или втроем если сложность проекта 2 или 3 соответственно. В зависимости от сложности на реализацию нужно 1,2 или 3 дня для 1 разработчика, после чего проект должен перейти в отдел тестирования.

QA специалист тратит на тестирование проекта любой сложности 1 день. После тестирования проект считается успешно реализованным и удаляется из системы. Программиста увольняют, если он более 3 дней подряд не занимался проектом.В случае если таких несколько, то каждый день из них увольняется только самый неопытный (участвовавший в наименьшем числе проектов).

На вход подается количество дней. На выходе подробная статистика: Количество реализованных проектов, нанятых и уволенных программистов. Начальные условия: в фирме нет проектов и нет программистов.  
  
Организовать структуру код в стиле ООП

Подписаться на ответы

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель · 5 дней назад

Добрый день. Большое спасибо за фидбек)

Насчет задания - попробуйте организовать ваш код примерно в таком виде: (это мое лично субъективное мнение и может не совпадать с другими :) )

- создать несколько классов: фирма, директор, проект, отдел

- в некоторых случаях необходимо создать наследников у классов: отдел разработки, тестирования ...

- у каждого класса будут свои методы и свойства. Например, метод директора - нанимать сотрудников и тд. А свойства - количество сотрудников в отделе на текущий момент, кол-во проектов и тд.

- после этого необходимо прописать логику взаимодействия между классами: какой метод на что влияет, как меняются свойства и тд.

----------------------------------------------------------------------------------------------------------------------------------

**Большая просьба: использовать один пример**

[Ярослав](https://www.udemy.com/user/iaroslav-abramov/) ·

Лекция 13

 · 9 месяцев назад

Очень бы хотелось увидеть иллюстрацию работы callback-функций на примере тех же first() и second()! У меня, например, не получилось!( Я пытался добавить callback в функцию first - пока безуспешно!(

Подписаться на ответы

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель · 9 месяцев назад

Что бы реализовать на примере этих функций - можно написать такой код:

1. function first(callback){
2. console.log(1);
3. callback();
4. }
5. function second(){
6. console.log(2);
7. }
9. first(second);

Заметьте, что я убрал команду setTimeout, тк она создает асинхронность. Это будет чуть позже, но что это значит:  
  
Вот такой код не будет выдавать нужный результат:

1. function first(callback){
2. setTimeout( function(){
3. console.log(1);
4. }, 500 );
5. callback();
6. }

setTimeout - это тоже функция и она действительно запускается раньше, чем callback. Но!  
Она не ждет, пока действия внутри выполнятся, а код идет дальше и запускает функцию callback.  
По факту мы получаем правильный порядок запуска функций, но без учета асинхронных операций внутри.

Отметить как "полезный" (5)

Ярослав Абрамов

[Ярослав](https://www.udemy.com/user/iaroslav-abramov/) · 9 месяцев назад

Так а с setTimeout, который в первом примере, callback ничего не дает что ли?

Я постарался ваш второй пример сделать с первыми двумя функциями - у меня нисчего не вышло - функция, в которой setTimeout, всё равно выполняется дольше.

Это меня несколько огорчило.

UPD: Ещё раз перечитал Ваш ответ, сложилось впечатление, что с setTimeout callback хоть и влияет на что-то, но результат мы видим такой... как и без него!

------------------------------------------------------------------------------------------------------------------------------------------

**обращение к callback**

[Дмитрий](https://www.udemy.com/user/dmitrii-2420/) ·

Лекция 13

 · 3 дня назад

Здравствуйте!

1. Kак при вызове абстрактной function() код понимает, что нужно выполнить именно callback()?

Подписка на ответы

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель · 2 дня назад

Добрый день. Функция-коллбэк передается в качестве аргумента.

1. open(page, cb) {
3. // Много всякого кода
5. cb()
6. }
8. open('index.html', () => console.log('Выполнился callback'))

Вот здесь в вызове функции open вторым аргументом я передал стрелочную функцию и она выполнится только после того, как весь код внутри исполнится.

Отметить как "полезный"

Отметить как "лучший ответ"

Дмитрий

[Дмитрий](https://www.udemy.com/user/dmitrii-2420/) · 2 дня назад

Здравствуйте!

Прошу пояснить следующие предположения:

1) перед () я могу написать любое название? если представить Ваш пример без стрелочной функции и добавить название перед () , то код будет выглядеть так?

open(page, cb) {

// Много всякого кода

cb()

}

open('index.html', randomName() {

console.log('Выполнился callback')

} )

2) я правильно понимаю, что в качестве аргументов open мы записываем абсолютно любые названия(page, cb), а суть им присваивается только в случае вызова этой функции и прописании значений этих аргументов('index.html', ()). Т.е. в наше случае, при вызове open, page - становится строкой 'index.html', а аргумент cb становится функцией ()?

Если мое предположение верно, то код можно переписать следующим образом, при этом его суть не изменится?

city(house, car) {

// Много всякого кода

car()

}

city('map.jpg', garage() {

console.log('Выполнился callback')

} )

Отметить как "лучший ответ"

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель · 2 дня назад

1) если хотите использовать именно название функции - то она должна быть объявлена где-то в коде. Если её нет, то обычно передают анонимную функцию (без имени). Она выполнится и сразу исчезнет.  
  
Важно! Круглые скобки при передаче функции-коллбэка не нужны. Мы передаем лишь суть функции, а вызовется она внутри родительской функции. Мы лишь передаем инструкции о том, что нужно сделать.  
  
2) да, все верно, кроме круглых скобок в аргументе)

[Дмитрий](https://www.udemy.com/user/dmitrii-2420/) · день назад

" Круглые скобки при передаче функции-коллбэка не нужны"

1) Вы про эту строку?

city('map.jpg', garage() {

2) Как это не надо, если в коде урока эти скобки есть( в строке learnJS("JavaScript", function(){   ):

1. function learnJS(lang, callback) {
2. console.log("Я учу "+ lang)
3. callback();
4. }
5. learnJS("JavaScript", function(){
6. console.log("Я прошёл 3й урок!")
7. })

?

Отметить как "лучший ответ"

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель · 6 часов назад

Наверное, немного не так выразился, прошу прощения.

Они не нужны, если мы передаем готовую функцию. Если прямо внутри аргумента - то как обычно. И да, имя может быть каким угодно) А внутри родительской - вызываем по названию аргумента.

Иван, большая просьба при ответе ссылаться на примеры, хотя бы на выше перечисленные, иначе все очень запутанно получается...

Итак, у нас есть три примера -Ваш со стрелочной функцией, без нее и из урока:

1.

1. open(page, cb) {
3. // Много всякого кода
5. cb()
6. }
8. open('index.html', () => console.log('Выполнился callback'))

2.

open(page, cb) {

// Много всякого кода

cb()

}

open('index.html', randomName() {

console.log('Выполнился callback')

} )

3.

1. function learnJS(lang, callback) {
2. console.log("Я учу "+ lang)
3. callback();
4. }
5. learnJS("JavaScript", function(){
6. console.log("Я прошёл 3й урок!")
7. })

Прошу пояснить где не нужны скобки.

В первом примере без них никак, поскольку вызов функции записан как () =>, т.е. практически кроме скобок ничего не записано

Третий пример взят из урока и ошибки там нет(т.е. строка

learnJS("JavaScript", function(){

записана верно)

А второй пример - считай копия третьего, ибо в строке

open('index.html', randomName() {

точно также вызывается функция как и в строке

learnJS("JavaScript", function(){

Так где же лишние скобки и почему?)

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель · день назад

Здесь просто нет такого примера)

Это правильно:

1. function learnJS(lang, callback) {
2. console.log("Я учу "+ lang)
3. callback();
4. }
6. function someAction() {
7. console.log("Я прошёл 3й урок!")
8. }
10. learnJS("JavaScript", someAction)

А это неправильно:

1. function learnJS(lang, callback) {
2. console.log("Я учу "+ lang)
3. callback();
4. }
6. function someAction() {
7. console.log("Я прошёл 3й урок!")
8. }
10. learnJS("JavaScript", someAction())

Разница лишь **в последних трех символах** кода)

Дмитрий:

Теперь начало проясняться...но некоторое недопонимание остается, прошу прощения за дотошность

не могли бы Вы пояснить попунктно мои следующие вопросы и предположения

1) т.е. круглые скобки мы ставим при вызове функции и при этом обязательно должно быть ее описание в { } ?

Вернемся к примеру по которому у Вас было замечание

1. city(house, car) {

2. // Много всякого кода

3. car()

4. }

5. city('map.jpg', garage() {

6. console.log('Выполнился callback')} )

2) Значит в строке 5 скобки в аргументе нужны?

в одном из ответов вы писали:

"Они(скобки) не нужны, если мы передаем готовую функцию. Если прямо внутри аргумента - то как обычно. И да, имя может быть каким угодно) А внутри родительской - вызываем по названию аргумента." по этому у меня следующие предположения:

3) Готовую функцию мы передаем только в строке 1 - называется у нас она car

4) Внутри аргумента скобки после функции ставим - в строке 5

5) Родительской функцией у нас является city - строка 1

6) вызов по названию аргумента происходит в строке 3

7) В случае, если мы использовали бы анонимную функцию, то строки 5-7 выглядели бы следующим образом:

5. city('map.jpg', () {

6. console.log('Выполнился callback') } )

а весь остальной код не изменился бы верно?

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель · 21 час назад

1) Да

2) Да, это скобки объявления функции, а не вызова

3) Это не готовая функция, это аргумент. В неё может прийти и готовая ф-я и самописная, как в строке 5.

4) Нет, не ставим. Скобки только для объявления функции

5) да

6) Да

7) Да, но стрелочная анонимная выглядит красивее)

Не путайте круглые скобки при объявлении функции:

function foo() { console.log(1) }

И при вызове:

(function foo() { console.log(1) }**()**)

Тут функция объявлена и сразу вызвана. (обернута в круглые скобки, чтобы не было багов с запуском)

Дмитрий

[Дмитрий](https://www.udemy.com/user/dmitrii-2420/) · 3 дня назад

Возможно я Вас запутал)

на вопросы

2) Значит в строке 5 скобки в аргументе нужны?

4) Внутри аргумента скобки после функции ставим - в строке 5

Вы ответили

2) Да, это скобки объявления функции, а не вызова

4) Нет, не ставим. Скобки только для объявления функции

обсуждаемый код:

1. 1. city(house, car) {
2. 2. // Много всякого кода
3. 3. car()
4. 4. }
5. 5. city('map.jpg', garage() {
6. 6. console.log('Выполнился callback')} )

I) В обоих вопросах я имел в виду скобки непосредственно после названия garage. Полагаю это скобки объявления функции и они здесь действительно нужны, верно?

II) По ответу "Нет, не ставим. Скобки только для объявления функции" Вы имели в виду, что при передаче готовой функции круглые скобки после названия функции не нужны, верно?

Но в обсуждаемом коде нет такой ситуации

III) Передача готовой функции выглядела бы следующим образом (в строке 6), верно?

1. 1. city(house, car) {
2. 2. // Много всякого кода
3. 3. car()
4. 4. }
5. 5. function garage() { console.log('Выполнился callback')}
6. 6. city('map.jpg', garage)

Отметить как "лучший ответ"

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель · 3 дня назад

Да, все верно)

--------------------------------------------------------------------------------------------------------------------------------------------

**9 вопрос**

**0**

[Арсений](https://www.udemy.com/user/arsenii-skorlupkin/) ·

Задание 5

 · 4 месяца назад

Программа выдает ложь, но мне не очень понятно, как так получилось.

Разве не вот так :

1. == - сравнивание по значению

2. Массивы a и b имеют одинаковое содержимое, то есть значение

3. a == b - правда ?

Или я что-то неправильно понял?

**3 ответа**

Следить за ответами

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель· 4 месяца назадОтвет

**2**

Добрый день. Нет, это работает несколько по другому. В этом случае сравнивается не значение, а именно сущности.

Наталия Назарина

[Наталия](https://www.udemy.com/user/nataliia-314/)· 4 месяца назад

**0**

два массива хоть и имеют одинаковое содержимое, но это два разных массива. как две коробки, в которых лежат одинаковые предметы, но коробки то разные. Я так это понимаю

Дмитрий

[Дмитрий](https://www.udemy.com/user/dmitrii-1239/)· 3 месяца назад

**1**

Дело в памяти. Когда создается массив либо другие объекты то на это выделяется память в компьютере.  
И к примеру когда создаешь массив а = [1, 2, 3] то цифры 1, 2, 3 загружаются в ячейки (биты) под нумерацией к примеру 1, 2, 3.  
Когда создаешь массив b = [1, 2, 3] то содержимое массива загружаются в другие ячейки памяти к примеру 12, 13, 14.

И когда сравнивается массивы то по сути идет сравнение не сколько содержимого а сколько ячеек (битов) и их порядковые номера разные.

То есть a [1(1), 2(2), 3(3)] == b [1(12), 2(13), 3(14)] вернет значение false.

На эту тему есть видео из курса CS50  в ютубе. Очень советую весь курс пересмотреть, в Гарварде сделан.

**alert( "1"[0] )**

**0**

[Наталия](https://www.udemy.com/user/nataliia-314/) ·

Задание 5

 · 6 месяцев назад

Правильно ли я поняла из видео, что строка "1" воспринимается как массив из которого просят вывести тот символ, который стоит на нулевом месте?

**1 ответ**

Следить за ответами

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель· 6 месяцев назад

**0**

Добрый день. Да, все верно

**Вопрос по заданию : Что выведет этот код: alert( "1"[0] )?**

**0**

[Екатерина](https://www.udemy.com/user/ekaterina-1048/) ·

Задание 5

 · 10 месяцев назад

Здравствуйте. Не могли бы Вы более подробно объяснить , почему в итоге выходит 1, а как же массив со значением 0 [0]?  И это тип данных строка выводится на экран в качестве ответа?

**1 ответ**

Следить за ответами

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель· 10 месяцев назад

**4**

Да, выводится тип данных строка. Проверить это можно при помощи:  
alert(typeof("1"[0]) )  
Если по простому, то строка - это набор символов по порядку и мы можем получать каждый символ в соответствии с его номером.

Пример:  
'Test'[1] == 'e'  
'Test'[3] == 't'  
  
Поэтому в строке '1' первым символом будет как раз 1. И именно его мы и выводим на страницу.

[Alex](https://www.udemy.com/user/alex-4057/) ·

Задание 5

 · 7 месяцев назад

Почему [] - null возвращает 0?

Да всё просто null это по сути ложь, и когда ты преобразуешь массив то это пустая строка, пустая строка это false, false - false = 0, надеюсь хоть чучуть понятнее стало

**Я немного запуталась. Почему, когда пишешь console.log([ ] - null + false + true ); на консоль выходит значение "1". А когда**

**0**

[Анастасия](https://www.udemy.com/user/anastasiia-pyrkina/) ·

Задание 5

 · 10 месяцев назад

Я немного запуталась. Почему, когда пишешь console.log([ ] - null + false + true ); на консоль выходит значение "1", а когда  console.log([ ] + false - null + true ) - "NaN"?

**1 ответ**

Следить за ответами

DA

[Demchenko](https://www.udemy.com/user/demchenko-andrii/)· 10 месяцев назад

**3**

**[ ] + false - null + true** - при сложении [] и false результатом будет "false"(строка), **особенность исключительно бинарного оператора "+"**, ну и дальйшея попытка от строки отнять 1 (null + true) приведет к тому самому NaN;

ЗАДАНИЕ 6

**Объявление переменных**

**0**

[Алексей](https://www.udemy.com/user/aleksei-popov-17/) ·

Задание 6

 · 2 дня назад

Добрый день, Иван. Я так понял, что если в дальнейшем не планируется динамическое изменение значения элемента , то можно не объявлять переменную, а писать значение элемента или действие "напрямую". Правильно ли так писать, или всё же лучше через объявление переменных?

1. document.getElementsByClassName('adv')[0].remove();
2. document.body.style.backgroundImage = "url(img/apple\_true.jpg)";

**1 ответ**

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель  
2 дня назад

**0**

Добрый день. Да, можно прописывать так и в этом нет никакой ошибки.

--------------------------------------------------------------------------------------------------------------------------------------------

**Белый prompt**

**0**

[Yukhym](https://www.udemy.com/user/yukhym-rybenchuk/) ·

Задание 6

 · 5 дней назад

Столкнулся с одной естетической проблемкой. В дз prompt вызывается в самом начале на белом фоне, (еще  на не загруженой странице), притом позиция в коде не имеет значения будь он в начале кода или в последнем рядке кода. А хотелось бы, щоб прогрузилась вся измененная страница и только потом наш prompt. Я решил вот так:

1. function promptApple() {
2. promptForApple.innerHTML = prompt('Как вы относитесь к технике Apple?');
3. }
4. setTimeout(promptApple, 1000);

**1 ответ**

Следить за ответами

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Преподаватель  
4 дня назад

**0**

Добрый день. Да, увы, но именно так prompt работает. А решение с таймером - самое правильное)

-------------------------------------------------------------------------------------------------------------------------------------------

**Как убрать из массива лишнее по содержимому?**

**0**

[Сергей](https://www.udemy.com/user/sergei-988/) ·

Задание 6

 · месяц назад

Иван, доброго времени суток! Очень интересный урок и ДЗ получилось! Спасибо!

Кроме простого способа, мне захотелось еще раз повторить массивы и попробовать строку из заголовка перевести в массив, а потом, добавив новое слово вернуть обратно. Однако как оказалось внутри исходного HTML заголовка лежит еще много пробелов и несколько переводов строк. Попробовал использовать splice, но он почему-то в цикле срабатывает один раз + непонятно как идентифицировать символ перевода строки - простое копирование элемента из консоли разработчика не помогло. Возможно,  вы посоветуете алгоритм действий для решения данной задачи! Буду очень признателен!

---

1. // ========= Заголовок ===========
2. // способ 1:
3. // document.getElementById("title").textContent = "Мы продаем только подлинную технику Apple";
5. // способ 2: делаем из строки массив и добавляем новый элемент
6. let title = document.getElementById("title").textContent;
7. let arrTitle = title.split(" ");

10. console.log(arrTitle);
11. // неожиданно обнаружилось множество ненужных элементов в строке
12. // пытаемся их убрать
13. // через delete, splice и forEach массива убрать не получилось, пробуем так:
14. for(key in arrTitle) {
15. console.log("ключ: " + key + " со значением: " + arrTitle[key]);
16. if (arrTitle[key] == " " || arrTitle[key] == "" || arrTitle[key] == "\n") {
17. arrTitle.shift(arrTitle[key]);
18. // что-то удаляет но не все
19. }
20. }
22. // пробуем 3-й раз
23. for(i=0; i < arrTitle.length; i++) {
24. if ( arrTitle[i] == "" || arrTitle[i] == " " ) {
25. arrTitle.splice(i, 1); // почему-то работает всего 1 раз похоже
26. // + не очень понятно как идентифицировать символ перевода строки
27. }
28. }
30. console.log(arrTitle);

https://i.udemycdn.com/redactor/raw/2020-01-19_21-51-07-e85fbf02eca5e54742f5b67e1b969445.jpg

**2 ответа**

Следить за ответами

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — ПреподавательОтвет  
месяц назад

**0**

Добрый день. Если хотите решить эту задачу - то есть способ намного проще.

Не очень часто используемый метод trim() для строки удаляет все пробелы вначале и в конце строки. И задача решается двумя строками:

1. let title = document.getElementById("title").textContent.trim();
2. let arrTitle = title.split(" ");
4. console.log(arrTitle);

Сергей

[Сергей](https://www.udemy.com/user/sergei-988/)  
месяц назад

**0**

О, это очень полезная информация! Иван, спасибо Вам большое!

**задачник/решебник по JS ?**

**0**

[Pavel](https://www.udemy.com/user/pavel-1945/) ·

Lecture 147

 · 4 месяца назад

Здравствуйте!

Не подскажите ресурс/книгу, где было бы много задач (с ответами) по чистому JS ?

Например тема "переменные в JS" и в конце темы огромный перечень задач по теме "переменные". Возможно даже без учебного материала по теме, но с задачами/ответами по определенной теме (задачи простые и среднесложные).

Как пример ресурс learn.javascript (но вот беда, мало задач под каждой темой!) (Ресурсы типа codewars, где всё вперемешку не рассматриваю)

**4 replies**

Follow replies

M

[Mykla](https://www.udemy.com/user/mykla/)  
4 месяца назад

**1**

Вот посмотри [Quizful - тесты онлайн](http://www.quizful.net/test) Тут тесты. Для тренировки вообще норм. Проверить себя да и попрактиковаться.

O

[Oksana](https://www.udemy.com/user/oksana-1057/)  
4 месяца назад

**1**

Морган Н. JavaScript для детей. Самоучитель по программированию. Очень доступно излагается материал с большим количество примеров. Есть задачи. Ответы можно найти в интернете

P

[Pavel](https://www.udemy.com/user/pavel-1945/)  
4 месяца назад

**1**

Спасибо!

Хотелось бы услышать мнение Ивана.

Иван Петриченко

[Иван](https://www.udemy.com/user/yan-kovalenko-2/) — Instructor  
4 месяца назад

**1**

Добрый день. Я согласен с предыдущими вариантами) Обычно во многих учебниках такие есть, но нужно подбирать индивидуально.