

Convert yyyy-MM-dd to MM/dd/yyyy in javascript

Asked 6 years, 9 months ago Active 3 months ago Viewed 38k times



6



3



This might be a simple solution but I am stuck, basically I need convert an incoming yyyy-MM-dd to MM/dd/yyyy also, if incoming date is nil, then output should also be nil.

Incoming date could be of following format

2015-01-25 or nil

Output date should be

01/25/2015 or nil

I was trying one from the following link [Convert Date yyyy/mm/dd to MM dd yyyy](#) but couldn't make it work.

Thanks for any help.

Forgot to mention, the incoming date which comes as nil is of the following format in an xml file

```
<Through_Date__c xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/>
```

So if I get the above format the output should be just be nil

javascript

Share Improve this question

Follow

edited May 23 '17 at 12:26



Community Bot

1 1

asked Jan 31 '15 at 0:24



Abhi

173 1 4 13

7 Answers

Active

Oldest

Votes



14



The date `toString` function has some support for formatting. See [this](#). And you also want to handle the undefined case which I took from [here](#). So, for your case you can just do this:

```
function format(inputDate) {
    var date = new Date(inputDate);
    if (!isNaN(date.getTime())) {
        // Months use 0 index.
        return date.getMonth() + 1 + '/' + date.getDate() + '/' + date.getFullYear();
    }
}
```

EDIT: Addressing the comment

If the padding is important you just need to add that in:

```
var d = date.getDate().toString();
(d[1]?d:"0"+d[0])
```

I've made an update to the [fiddle](#)

Share Improve this answer Follow

edited Nov 6 '18 at 0:01

answered Jan 31 '15 at 0:38



Pattle

6,605

7

29

53



hassassin

4,904

1

26

37

return date.getMonth() + 1 should be return (date.getMonth() + 1) – [user3172663](#) Jan 20 '17 at 5:46

Actually that is not needed, since it evaluates left to right it will do x.getMonth() + 1 first, then add the + '/' which will stringify. Although, I do agree that would be more readable – [hassassin](#) Jan 20 '17 at 18:44

Try using RegEx:

10

[Show code snippet](#)

Using String#split and Array#join, push & shift:



[Show code snippet](#)

Share Improve this answer Follow

edited Jun 13 '18 at 14:05

answered Jan 31 '15 at 1:47

[user2575725](#)

try nicer variable names. don't use abbreviations like ptrn, just write "pattern". It's soo much easier to read! – [Kinesias](#) Jun 13 '18 at 11:42

in the 2nd example, line #2 can be this (without extra escape character): var array = (input || '').toString().split(/-/g); – [Darren Gates](#) May 24 at 3:56

if you wanna go ghetto style and use easily understandable code, and you dont care about using a date object, try this!

3

```
function changeDateFormat(inputDate){ // expects Y-m-d
  var splitDate = inputDate.split('-');
  if(splitDate.count == 0){
    return null;
  }

  var year = splitDate[0];
  var month = splitDate[1];
  var day = splitDate[2];

  return month + '\\' + day + '\\' + year;
```

```

    }

    var inputDate = '2015-01-25';
    var newDate = changeDateFormat(inputDate);

    console.log(newDate); // 01/25/2015
  
```

Share Improve this answer Follow

answered Jan 31 '15 at 0:44



hamobi

7,486 4 33 57

If your date has not yet been parsed from a string, you can simply rearrange its components:

0

```

var s = '2015-01-25';
if (s) {
  s = s.replace(/(\d{4})-(\d{1,2})-(\d{1,2})/, function(match,y,m,d) {
    return m + '/' + d + '/' + y;
  });
}
  
```

Share Improve this answer Follow

answered Jan 31 '15 at 0:42



Ben Grimm

4,176 2 13 24

Thanks guys, I was able to do grab some ideas from all your posts and came up with this code which seems to working fine in my case

0

```

if((typeof inStr == 'undefined') || (inStr == null) ||
(inStr.length <= 0)) {
  return '';
}
var year = inStr.substring(0, 4);
var month = inStr.substring(5, 7);
var day = inStr.substring(8, 10);
return month + '/' + day + '/' + year;
  
```

Share Improve this answer Follow

answered Jan 31 '15 at 18:46



Abhi

173 1 4 13

You can also try the method below using vanilla JS. I have converted the date to a string & parsed it to get the format you're looking for:

0

```

function tranformDate(strDate) {
  let result = '';

  if (date) {
    let parts = date.split('-');
    result = `${parts[1]}/${parts[2]}/${parts[0]}`;
  }
  return result;
}
  
```

```

}

let date = new Date().toISOString().split('T')[0];
console.log('raw date: ' + date);
console.log('formatted date: ' + tranformDate(date));

```

Run code snippet

Hide results

[Full page](#)

raw date: 2021-11-10	10:07:53.388	▲
formatted date: 11/10/2021	10:07:53.388	▼

Share Improve this answer Follow

edited Jan 17 '19 at 12:59

answered Jan 17 '19 at 11:25



Hamzeen Hameem

1,887 1 24 23



you can deal your javascript dates in various formats.

0

For dd/MM/yyyy you can use



```
var date = new Date().toLocaleDateString()
```



or

```
var date = new Date('2021-07-28').toLocaleDateString()
```

output: '28/07/2021'

For MM/dd/yyyy

```
var date = new Date().toLocaleDateString("en-US", { year: "numeric", month: "2-digit",
day: "2-digit" })
```

or

```
var date = new Date('2021-07-28').toLocaleDateString("en-US", { year: "numeric",
month: "2-digit", day: "2-digit" })
```

output: '07/28/2021'

Alternatively you can handle custom date formats using following date functions

```
let date = new Date()
let dateString = [
    date.getMonth() + 1,
    date.getDate(),
    date.getFullYear(),
].join('/')
}
```

output: 07/28/2021

[Share](#) [Improve this answer](#) [Follow](#)

answered Jul 28 at 6:57



[sarath chandra](#)

61 4