

JS Advanced: Exam 23 July 2017

Problems for exam preparation for the [“JavaScript Advanced” course @ SoftUni](#). Submit your solutions in the SoftUni judge system at <https://judge.softuni.bg/Contests/699/>.

Problem 4. Contacts Builder (Object Interacting with DOM)

Write a JS **class** that generates a **contact** info box. You will receive a person's first name, last name, phone and email. Compose the markup for the contact box, attach all the needed events and when a **render** function is called, **append** the newly created element to the document.

A contact info box is **composed** of first name, last name, phone, email (all strings) and a property which indicates if the contact is online or not. Clicking a button on the box **toggles** the visibility of the person's contact information (phone and email). *See the examples for more details.*

The **constructor** of your class needs to take **four** string arguments - first name, last name, phone, email. Additionally, the class should also contain the following functionality:

- Property **online** – Boolean value, initially set to **false**
- Function **render(id)** – takes one string argument, **generates the HTML element** and **appends** it to the element with ID equal to the argument

When the value of the **online** property is changed, the corresponding HTML should be updated – if it's set to **true**, add the class **"online"** to the div with class **"title"** (containing the name). If it's **false**, remove the class **"online"**.

A contact info box should have the following HTML structure:

Contact
<pre><article> <div class="title">{firstName lastName}<button>&#8505;</button></div> <div class="info"> &#9993; {phone} &#9993; {email} </div> </article></pre>

When the box is initially created, the div with class **"info"** must be **hidden**. Clicking the button **toggles its visibility**.

You can use the following HTML skeleton to test your functionality:

```
contacts.html
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Contacts Builder</title>
  <style>
    article { background: #EEE; padding: 1px; width: 250px; margin: 1px; }
    .title { background: #DDD; padding: 5px; font-weight: bold; }
    .title button { float: right }
    .info { margin: 5px; }
    .info span { display: block; }
    .online { background: #9E9; }
  </style>
```

```

<script src="https://code.jquery.com/jquery-3.1.1.min.js"></script>
</head>
<body>
<div id="main"></div>
</body>
</html>

```

Examples

Your solution can be **tested** using the following code:

Sample JavaScript

```

let contacts = [
  new Contact("Ivan", "Ivanov", "0888 123 456", "i.ivanov@gmail.com"),
  new Contact("Maria", "Petrova", "0899 987 654", "mar4eto@abv.bg"),
  new Contact("Jordan", "Kirov", "0988 456 789", "jordk@gmail.com")
];
contacts.forEach(c => c.render('main'));

// After 1 second, change the online status to true
setTimeout(() => contacts[1].online = true, 2000);

```

Ivan Ivanov
i

Elements

<!DOCTYPE html>
<html>
 <#shadow-root (open)>
 <head>...</head>
 <body>
 <div id="main">
 <article>
 <div class="title"> == \$0
 "Ivan Ivanov"
 <button>i</button>
 </div>
 <div class="info" style="display: none;">...</div>
 </article>
 </div>
 </body>
 </html>

Ivan Ivanov
i

0888 123 456
i.ivanov@gmail.com

Elements

<!DOCTYPE html>
<html>
 <#shadow-root (open)>
 <head>...</head>
 <body>
 <div id="main">
 <article>
 <div class="title"> == \$0
 "Ivan Ivanov"
 <button>i</button>
 </div>
 <div class="info" style="display: block;">...</div>
 </article>
 </div>
 </body>
 </html>

Ivan Ivanov
i

0888 123 456
i.ivanov@gmail.com

Elements

<html>
 <#shadow-root (open)>
 <head>...</head>
 <body>
 <div id="main">
 <article>
 <div class="title online"> == \$0
 "Ivan Ivanov"
 <button>i</button>
 </div>
 <div class="info" style="display: block;">...</div>
 </article>
 </div>
 </body>
 </html>