

# Домашнее задание. Урок 5. Курс Javascript 1

1. Посмотрите как сделана игра бродилка на странице, а не в консоли <https://youtu.be/5m0vo007dlQ> Исходный код находится в папке в примерах к уроку «игра (в дз)».
2. Посмотрите как можно сделать валидацию формы <https://youtu.be/3i-2PwgrTkg> Исходный код находится в папке в примерах к уроку «валидация формы (в дз)».

### 3. а. Сделайте модальное (появляющееся и исчезающее) окно

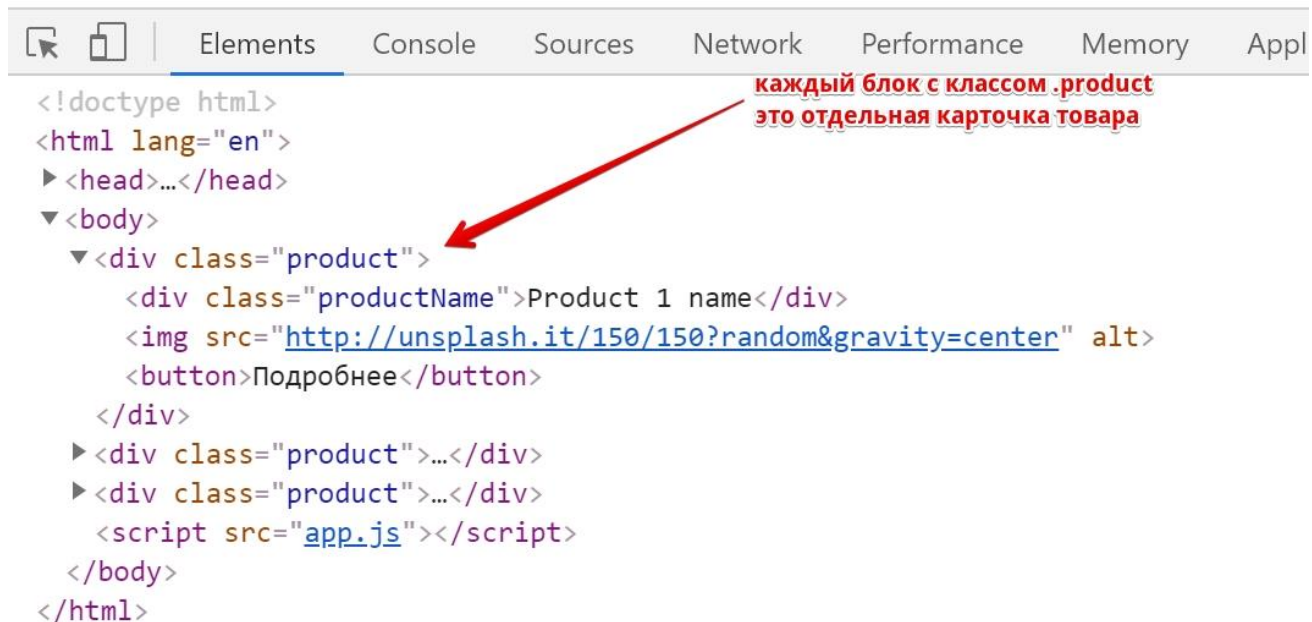
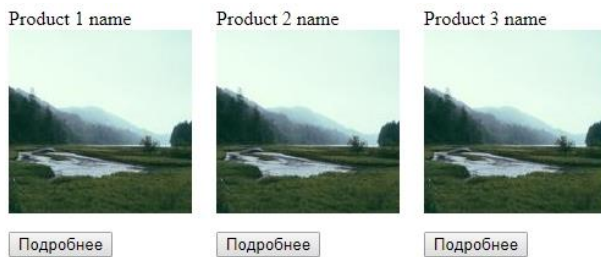
Решение можно посмотреть здесь: <https://youtu.be/cZZaBXlvPiE>

б. (не обязательное задание) Сделайте анимацию при появлении и исчезании. Анимации можно взять здесь:

1. <https://github.com/miniMAC/magic> (демо <https://www.minimamente.com/project/magic/> )
2. <https://github.com/daneden/animate.css> (демо <https://daneden.github.io/animate.css/> )
3. <http://animista.net/>

Решение можно посмотреть здесь: <https://youtu.be/oseHRK99hvk>

### 4. Сделайте несколько карточек товара, примерно такого вида:



При нажатии на кнопку «Подробнее» вместо картинки должен появляться текст с описанием товара, примерно так:

Product 1 name  
Lorem ipsum dolor sit  
amet, consectetur  
adipiscing elit. Aenean  
commodo ligula eget  
dolor. Aenean massa.  
Cum sociis natoque  
penatibus et magnis dis  
parturient montes,  
nascetur ridiculus mus.

Product 2 name



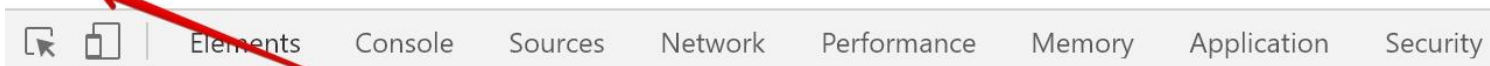
Подробнее

Product 3 name



Подробнее

Отмена



```
<!doctype html>
<html lang="en">
  <head>...</head>
  <body>
    <div class="product">
      <div class="productName">Product 1 name</div>
      <div class="desc">
        "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget
        dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes,
        nascetur ridiculus mus."
      </div>
      
      <button>Отмена</button>
    </div>
    <div class="product">...</div>
    <div class="product">...</div>
    <script src="app.js"></script>
  </body>
</html>
```

поменяли текст на кнопке

добавили описание

а картинку скрыли

### Подсказки:

1. Имейте в виду, что назначить обработку события можно по отдельности каждому элементу, а не всем сразу, т.е., например, для этих кнопок:

```
const buttons = document.querySelectorAll('button');
```

мы не можем написать:

```
buttons.addEventListener('click', function() {});
```

т.к. это коллекция, а не одна конкретная кнопка, воспользуйтесь например методом `forEach`

<https://mzl.la/1AOMMWX>

2. С помощью `querySelector` можно искать не только в документе, но и внутри какого-то элемента, например: <https://codepen.io/IgorKubikov/pen/arWKyz?editors=1010>

3. У объекта события `MouseEvent`, которое происходит при клике есть свойство `target`, которое указывает на каком элементе произошел клик, а у него в свою очередь есть свойство `parentNode`, которое содержит родительский элемент. Подробнее: <https://codepen.io/IgorKubikov/pen/arWKpo?editors=1010>

Решение можно посмотреть здесь: <https://youtu.be/N15nBxwBAAY>

**5.** Создать функцию, генерирующую шахматную доску. При этом можно использовать любые html-теги по своему желанию. Доска должна быть разлинована соответствующим образом, т.е. чередовать черные и белые ячейки. Строки должны нумероваться числами от 1 до 8, столбцы – латинскими буквами A, B, C, D, E, F, G, H.

Решение можно посмотреть здесь: <https://youtu.be/8s7ne43R5oM>

**6.** (усложненное задание, делайте по желанию) Заполнить созданную таблицу фигурами, фигуры должны выглядеть как картинки, либо можно использовать символы (существуют символы шахматных фигур) причем все фигуры должны стоять на своих местах и быть соответственно черными и белыми.

Решение можно посмотреть здесь: <https://youtu.be/5Zzz7K0MWtg>