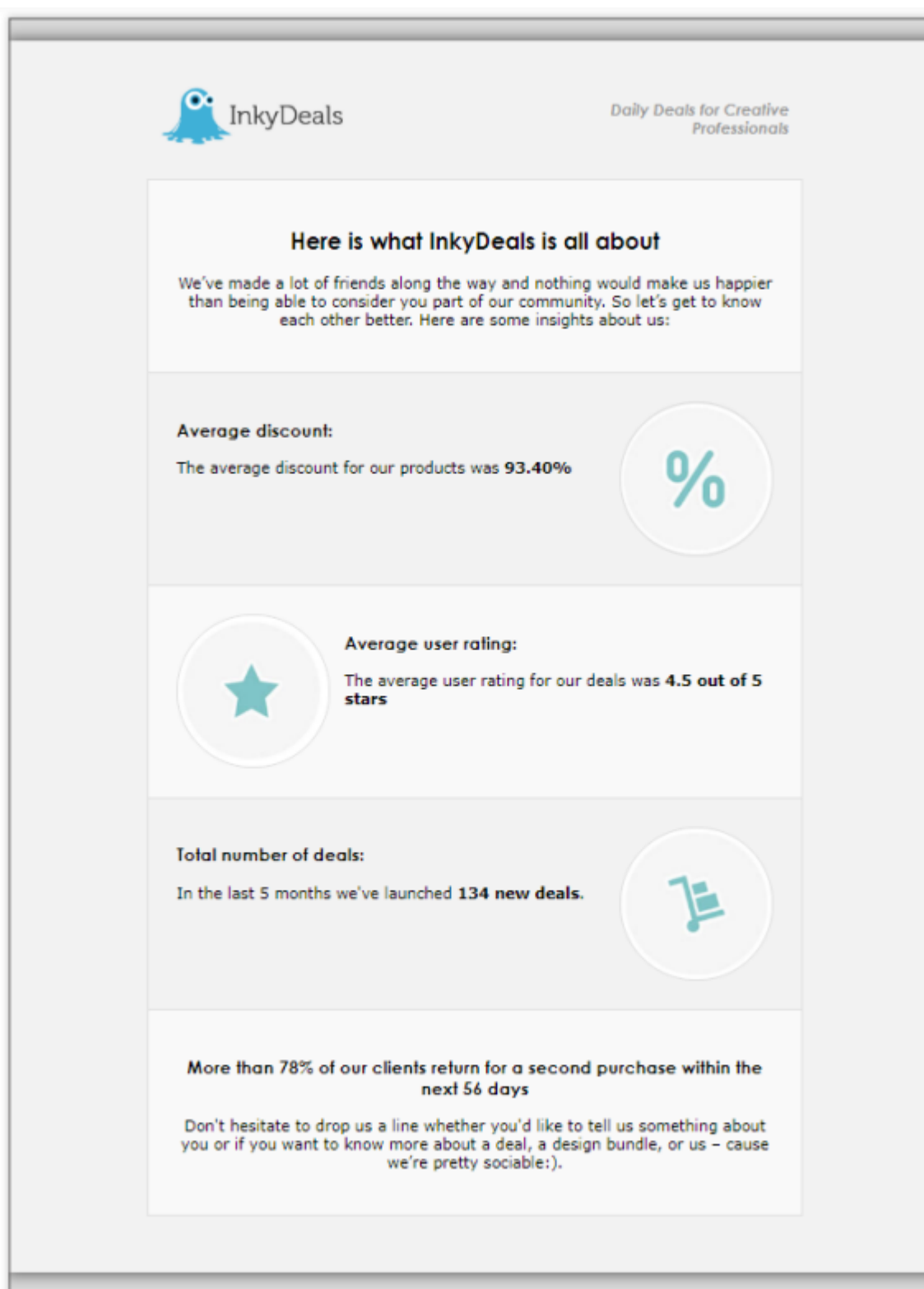




## Верстка статьи с изображениями

Используем изображения для верстки статьи, которая будет похожа на небольшой рекламный блок.



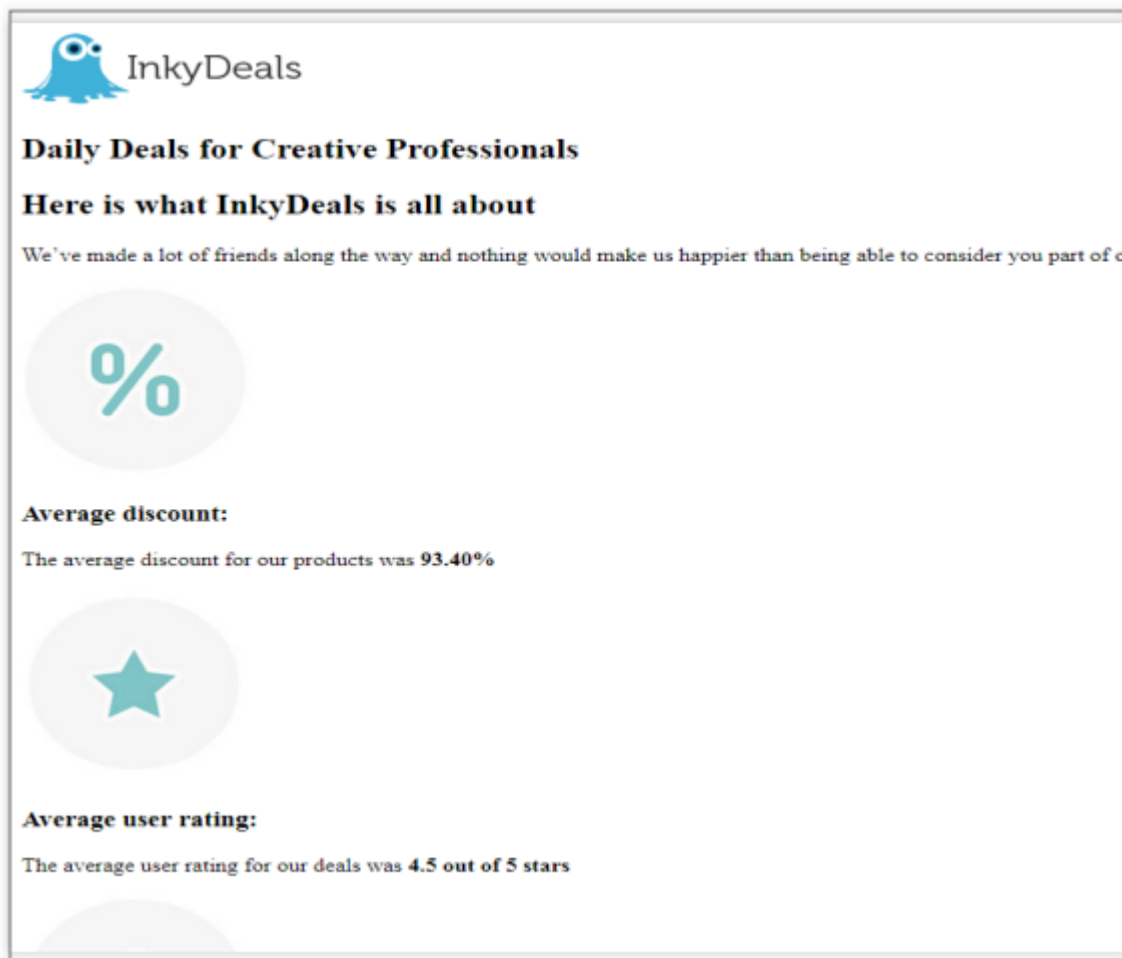
# HTML CSS JS

```
<article>
  <header>
    <div id="logo">
      
    </div>
    <div id="slogan">
      <h1>Daily Deals for Creative Professionals</h1>
    </div>
  </header>

  <div id="main">
    <div class="block">
      <h2>Here is what InkyDeals is all about</h2>
      <p>We've made a lot of friends along ...</p>
    </div>
    <div class="block">
      
      <h3>Average user rating:</h3>
      <p>The average user rating for our deals was
        <strong>4.5 out of 5 stars</strong>
      </p>
    </div>
    <div class="block">
      
      <h3>Total saved by member:</h3>
      <p>InkyDeals members have saved
        <strong>$28,320,804</strong> in 2018 alone.</p>
    </div>

    <div class="block">...</div>
    <div class="block">...</div>
    <div class="block">...</div>
    <div class="block">

      <h3>More than 78% of our ... </h3>
      <p>Don't hesitate to drop us ...</p>
    </div>
  </div> <!-- /#main -->
</article>
```



В разметке есть корневой элемент `<article>`, внутри которого есть тег `<header>` и `<div id="main">`. Внутри основного `div`-а расположены несколько `div`-ов с классом `block`. Первый и последний из `<div class="block">` содержат только заголовок и текст, а все остальные — еще и изображения с классами `alignleft` и `alignright`, которые нужны для размещения изображений слева или справа от текста.

```
* { box-sizing: border-box; }

body {
  font-family: Verdana, Geneva, sans-serif;
  background-color: #d4d4d4;
}

article {
  width: 1000px;
  margin: 40px auto;
  padding: 20px 150px 60px;
  background-color: #f0f0f0;
  box-shadow: 0 0 16px rgba(0, 0, 0, 0.7);
}
```

# HTML CSS JS

Для всех элементов мы задаем **box-sizing: border-box**, чтобы браузер учитывал рамки (**border**) и внутренние отступы (**padding**) в размере ширины (**width**) элемента. Если посмотреть на значение **width: 1000px y article**, то этот элемент должен занять большую часть экрана, однако **article** имеет также довольно приличный **padding: 20px 150px 60px**. Поэтому за счет свойства **boxsizing: border-box** на ширину, занимаемую контентом, придется **1000px — 150px\*2 = 700px**.



Добавим рамку к **div id="main"** и изменим шрифты для заголовков.

```
#main { border: 2px solid #ddd; }  
h1, h2, h3 { font-family: 'Century Gothic', monospace; }
```

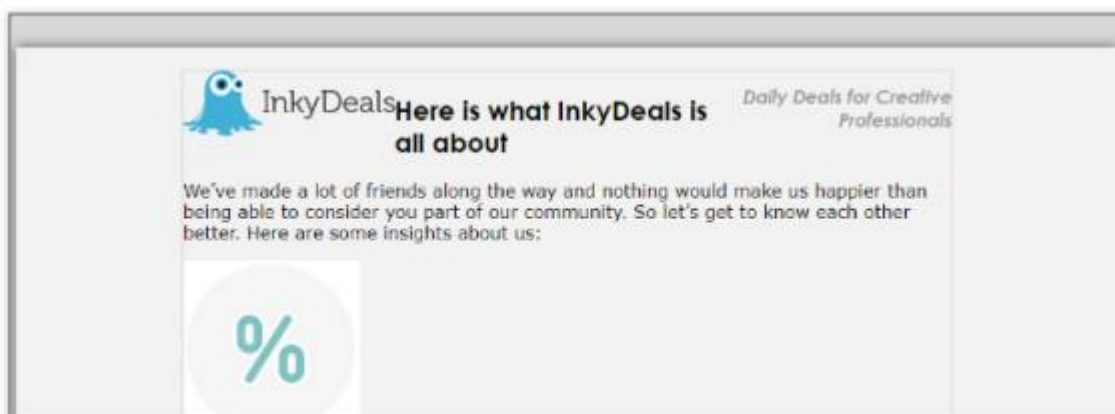
Логотип расположим слева, а слоган — справа — для этого у нас в разметке есть **div**-ы с соответствующими **id**.

```
#logo { float: left; }  
#slogan {  
  float: right;  
  font-style: italic;
```

```
  font-weight: normal;  
  font-size: .7rem;  
  text-align: right;  
  color: #909090;  
  width: 30%;  
}
```

# HTML CSS JS

Получим в результате весьма специфичный вид: рамка, заданная для **div id="main"**, «переехала» выше логотипа и слогана и туда же подтянулся заголовок первого блока.

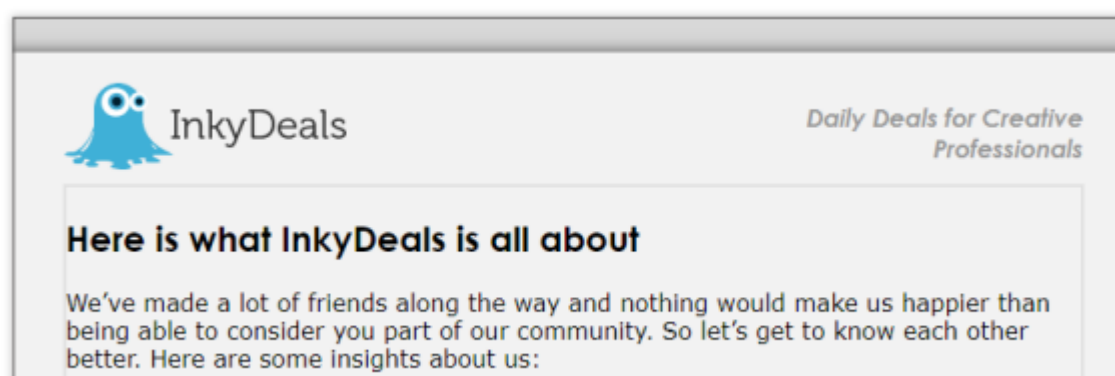


Все это произошло из-за использования свойства **float**: помните о его особенности превращать в **0** высоту родительского элемента (в нашем случае **header**). Поэтому используем для header правила класса **clearfix**.

```
.clearfix::before, .clearfix::after {  
    content: "";  
    display: table;  
}  
.clearfix::after { clear: both; }
```

```
<header class="clearfix">  
    <div id="logo">  
      
    </div>  
    <div id="slogan"> ... </div>  
</header>
```

Добавив атрибут **class="clearfix"** к **header**, мы приведем внешний вид шапки статьи к нужному виду.

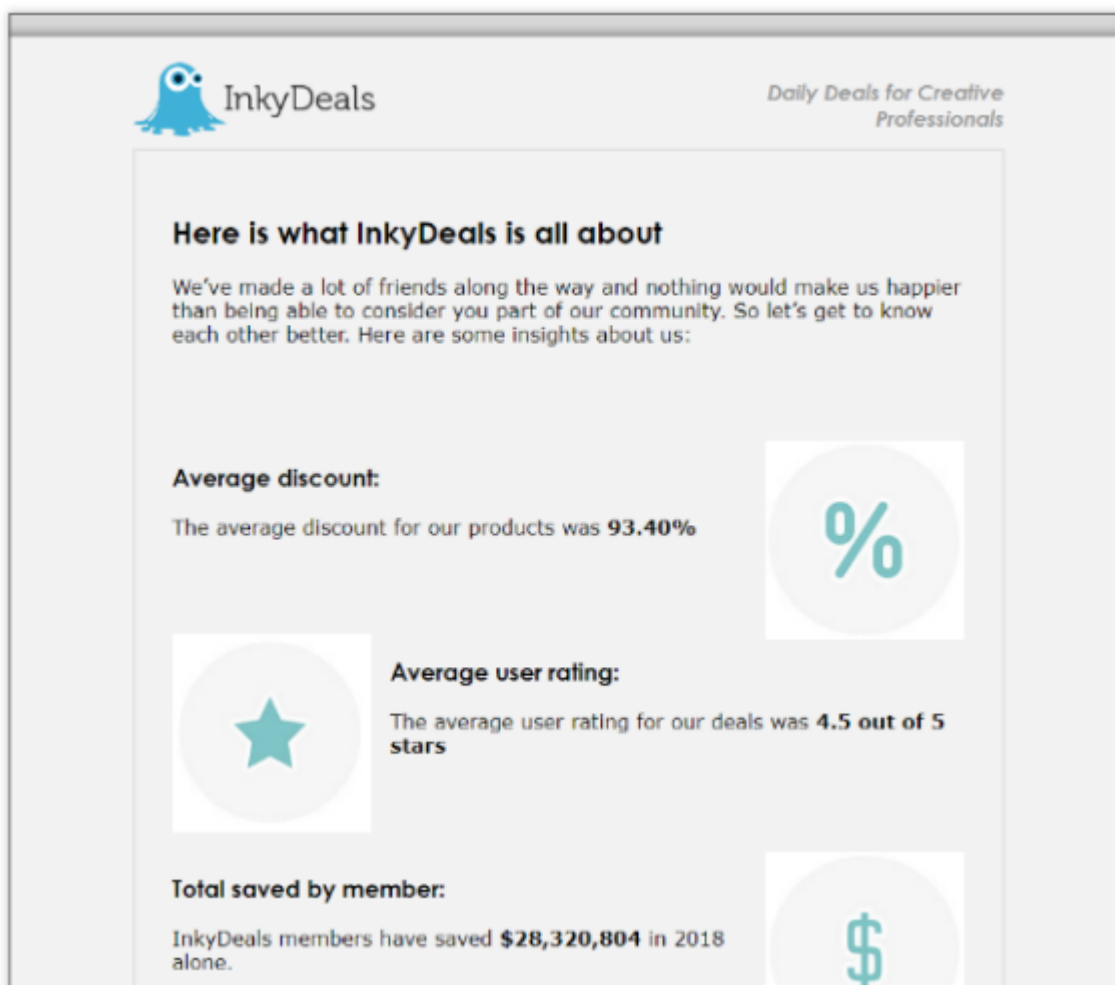


# HTML CSS JS

Изменим теперь внешний вид блоков, добавив стили классов для `div class="block"` и для изображений.

```
.alignleft {  
    float: left;  
    margin-right: 15px;  
}  
  
.alignright {  
    float: right;  
    margin-left: 15px;  
}  
  
.block { padding: 30px; }
```

Внешний вид блоков улучшился, но им не хватает выделения.

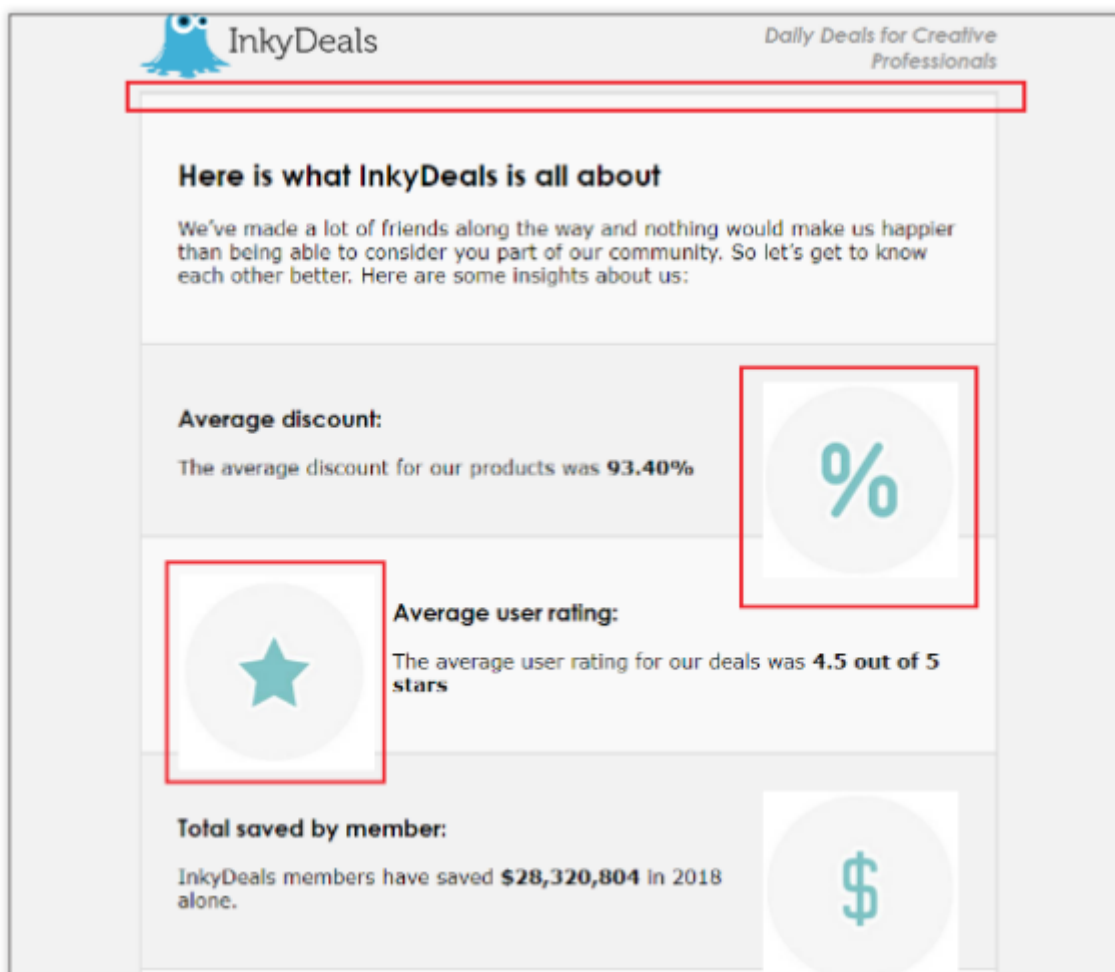


# HTML CSS JS

Добавим для нечетных блоков светлый цвет фона и рамки сверху и снизу.

```
.block:nth-child(odd) {  
    background-color: #f8f8f8;  
    border-bottom: 2px solid #ddd;  
    border-top: 2px solid #ddd;  
}
```

Однако, и здесь не обошлось без сюрпризов: рамка вверху и внизу увеличены и изображения выходят за пределы блока.



Для рамок несколько изменим свойства для `div id="main"`.

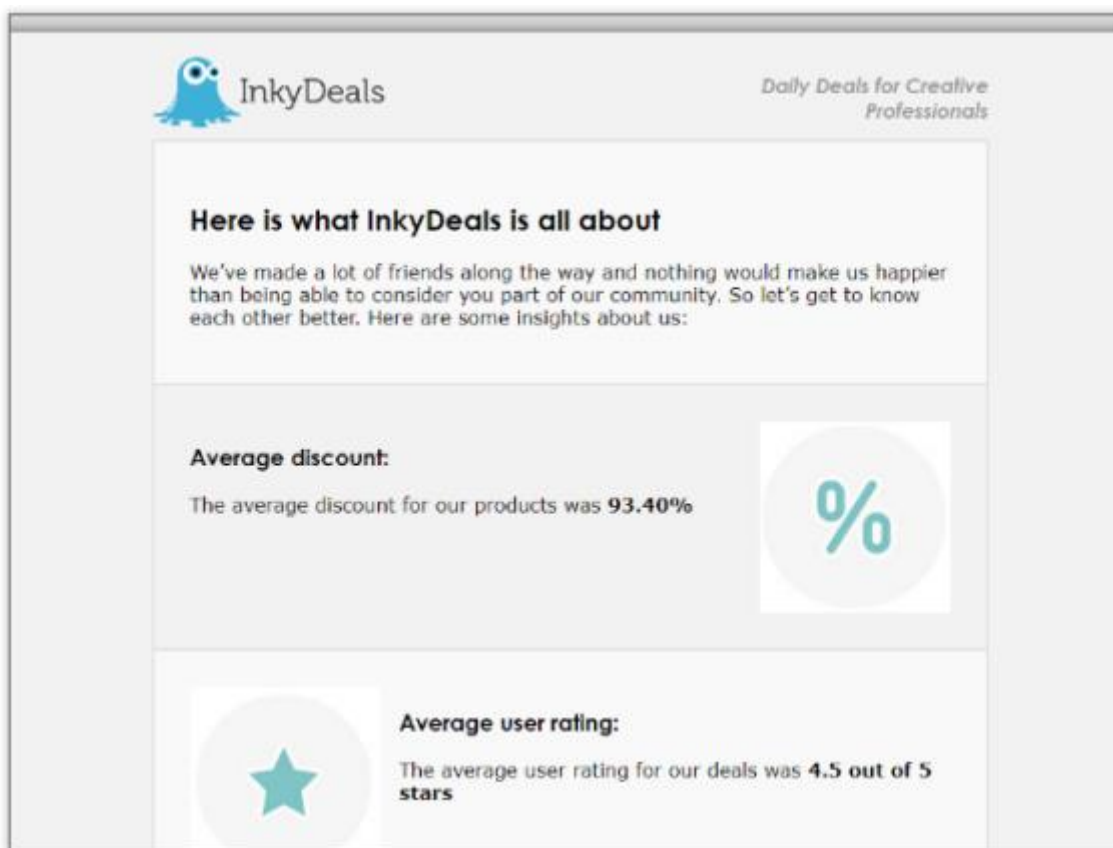
```
#main {  
    border-left: 2px solid #ddd;  
    border-right: 2px solid #ddd;  
}
```

# HTML CSS JS

А для **.block** добавим те же правила, что и для класса **clearfix**.

```
.clearfix::before, .clearfix::after,  
.block::before, .block::after {  
    content: "";  
    display: table;  
}  
.clearfix::after, .block::after { clear: both; }
```

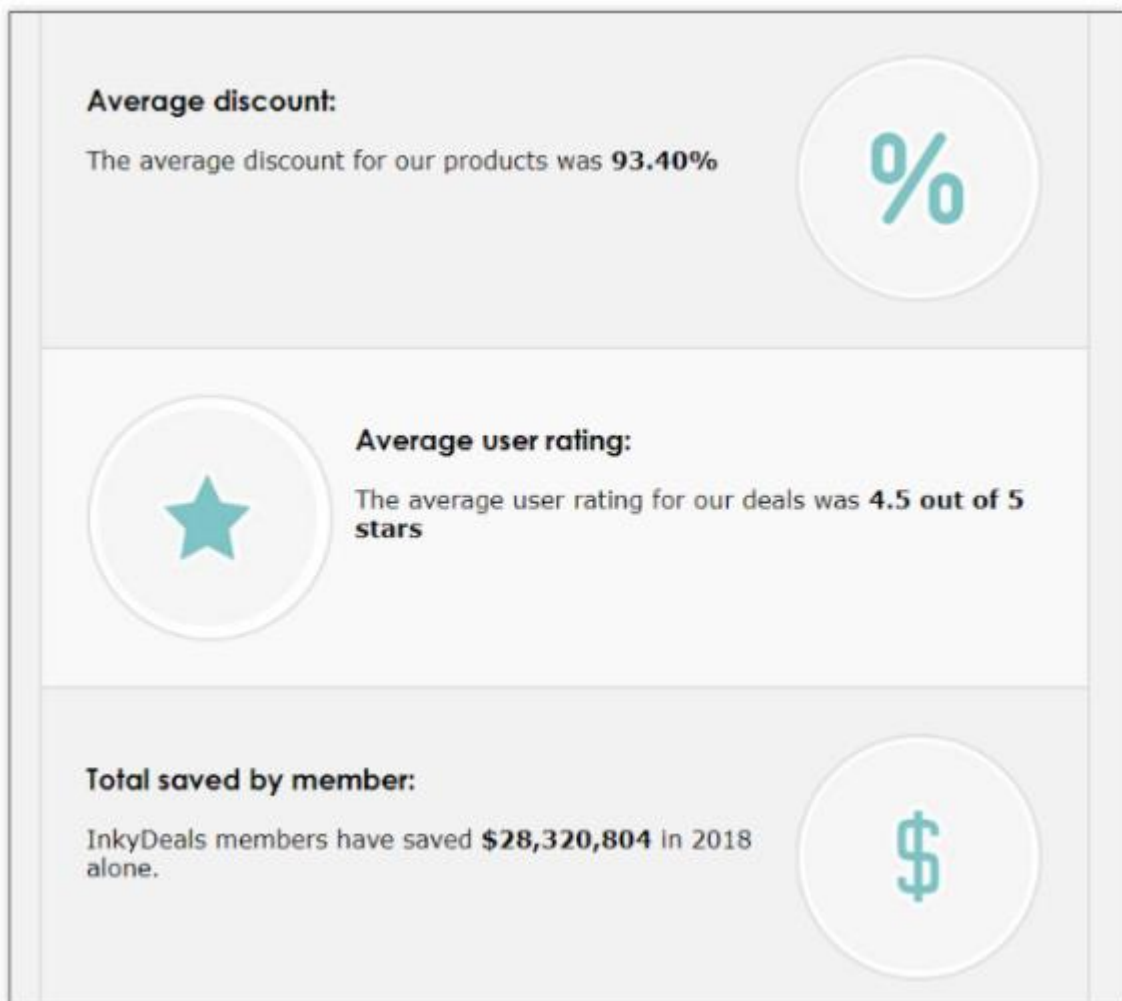
Теперь изображения не выходят за пределы блоков.



Еще одна проблема с изображениями заключается в том, что на фоне блока рамка вокруг изображений, которые сохранены в формате **JPG**, выделяется некрасивым квадратом. Можно удалить фон вокруг изображения и повторно сохранить их в формате **PNG** с прозрачностью. Есть и другой способ — сделать изображения круглыми с помощью **border-radius: 50%**.

```
.block img {  
    border: 2px solid #ddd;  
    border-radius: 50%;  
}
```





Теперь изменим форматирование для первого и последнего **div class="block"**, в котором у нас есть только текст и увеличим отступы в **header**.

```
.block:first-child, .block:last-child {  
    text-align: center;  
}  
  
header {  
    padding: 30px 15px;  
}
```

Теперь статья приняла нужный вид.

