

# Шаблоны писем в Webimitation

Будем считать это одним из главных мануалов в отправке писем через вебмейлинг. Правильно сделанные макеты, с корректным HTML обеспечивают стабильную отправку писем без ошибок в сервисе, а так же, корректное прохождение внутреннего спам фильтра у почтовиков.

В мануале используем очень много иллюстрация и примечаний к ним. Рекомендуем изучать все последовательно и вникать в суть. Если вам что-то не понятно, обратитесь к автору.

Изначально определим термины, которые будет использоваться в мануале:

**Шаблон** - это структура HTML кода, используемая как заготовка для одного или нескольких макетов.

**Макет** - само письмо, состоящее из шаблона HTML кода и текста. Непосредственно макет отправляется в рассылку.

**Валидный HTML** - в вебмейлинге подразумеваем под HTML тегами, используемыми почтовыми сервисами в своем стандартном редакторе. Непосредственно корректный код, оформленный согласно стандартного обобщенного языка разметки по стандарту [ISO 8879](#).

**Сигнатура** - набор идентичных символов в шаблоне, макете, структуре HTML кода, которые повторяются и дублируются.

## Общее понимание создания шаблонов для вебмейлинга

HTML код в письмах отличается от HTML, который мы привыкли видеть на своих сайтах. Он крайне ограничен и минимизирован. Имеет определенный набор используемых тегов. Используется табличная верстка шаблонов.

В отличие, от отправки писем через SMTP протокол, когда сам Администратор решает что ему отправлять и какой HTML указать в BODY, мы ограничены набором тегов, используемых редактором самого почтового сервиса. Да, так как мы создаем письмо через POST/GET запросы, то по сути тоже можем внести любой формат и символы HTML кода, контент и он по сути будет принят. Но малая вероятность что он будет отправлен при массовых рассылках.

Простой пользователь, при написании личного письма, не может через редактор указывать различные теги и атрибуты HTML. Поэтому необходимо использовать валидный HTML.

В определенных случаях, когда пользователь пишет письмо к примеру в Microsoft Word, потом его просто копирует и вставляет в редактор, то редактор Mailru согласно своим свойствам пытается преобразовать данный текст с примерным дизайном. При отправке такого письма, в свойствах, мы можем увидеть "сложный" HTML с различными тегами, стилями и стилевыми свойствами.

Но это как раз хорошая новость. Значит MailRu это все поддерживает и мы можем использовать данные теги, стили в написании своих шаблонов.

Я же рекомендую, для правильного понимания, начинать от простого к сложному. Сделать просто валидный HTML со стандартными тегами.

Но как же понять, какой именно валидный HTML и какие теги использовать можно?

Для нашего примера будем рассматривать почтовый сервис MailRu. Отправьте со своего личного ящика письмо, самому себе с простым текстом:

Привет друг  
Как твои дела?  
Что у тебя сегодня по плану?

А теперь, посмотрите в самом коде, какие именно указаны HTML теги. Если не копировать текст, а вот прямо написать его, без какого либо форматирования, то можно увидеть, что весь текст MailRu помещает в тег `div`, при этом дает ему определенный свой идентификатор. Строки не завернуты нево что, а просто разбиты между собой тегом `<br>`, при этом старого формата. Новый формат такой `<br />`

Стоит заметить, что нет никакой лишней разметки, по типу: `<head>`, `<body>`, `<title>` и тд. Исключительно текст и тег `<br>`

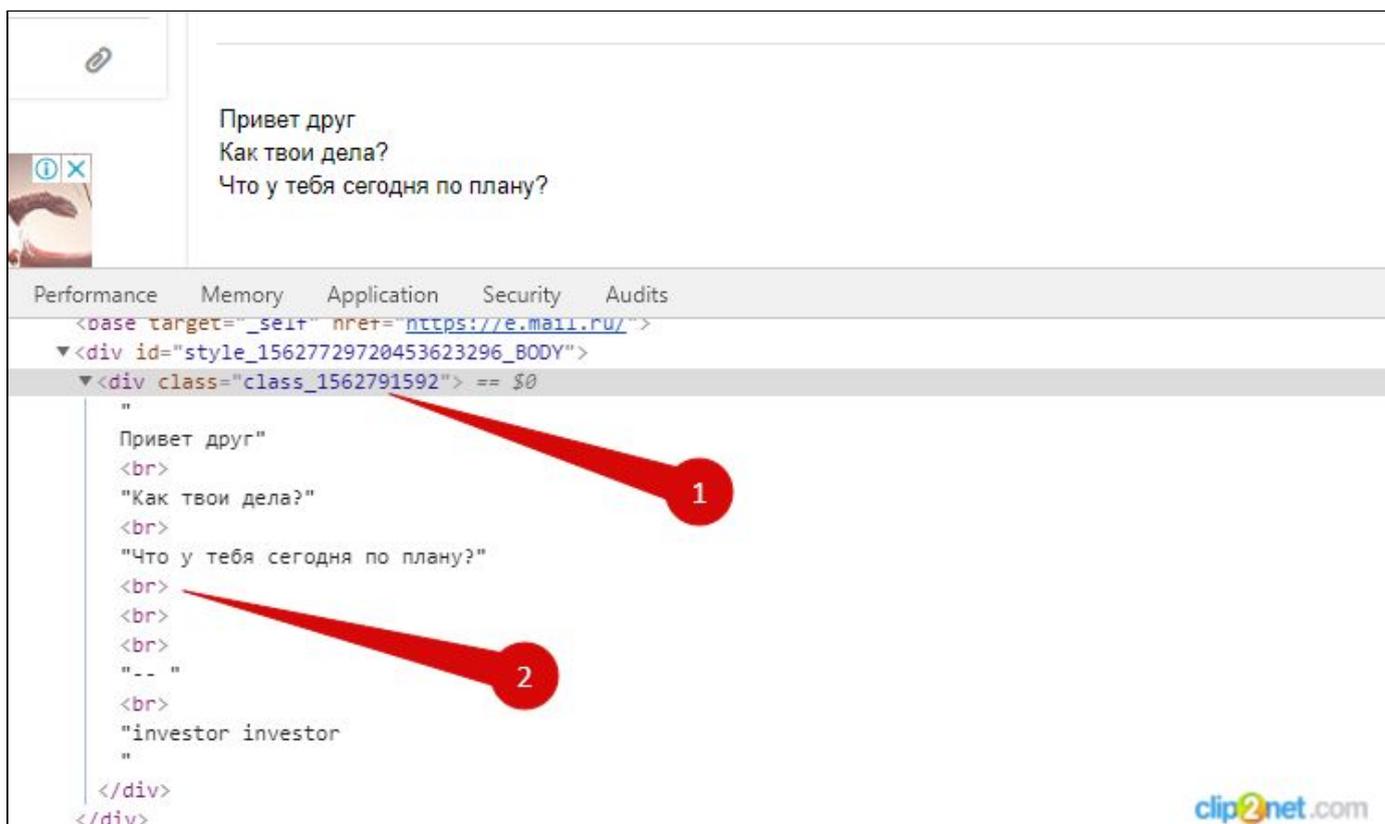


Рис #1

Если этот такой же текст, выделить жирным и сделать наклон, мы увидим немного другую картину. Опять таки, весь текст MailRu помещает в тег `<div>` и дает ему свой идентификатор. Далее, наши слова уже обернуты в дополнительные теги `<em>`, а потом в `<strong>`. Таким образом, мы

понимаем, что MailRu, для того чтоб сделать изменение в шрифте может использовать ступенчатую структуру. То есть, сначала заворачивает в один тег, а поверх в другой. Значит, в своих шаблонах, мы можем использовать такой метод.

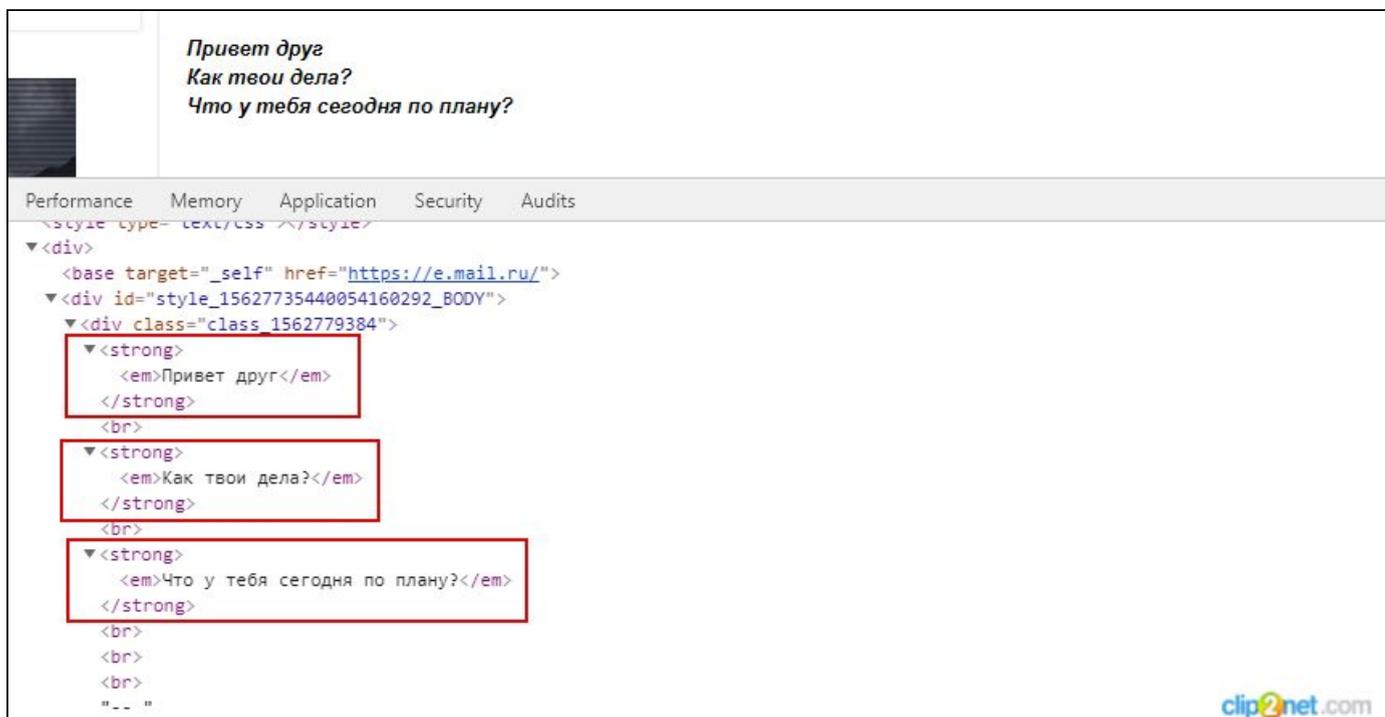


Рис #2

Можно еще больше усложнить форматирование текста, добавив размер, цвет, шрифт. В этом случае, мы наблюдаем появление тега `<span>`, в который добавлен атрибут `style`. Уже в нем используются стилевые свойства. Стоит посмотреть, какие именно стилевые свойства использует MailRu и по возможности придерживаться данного формата.

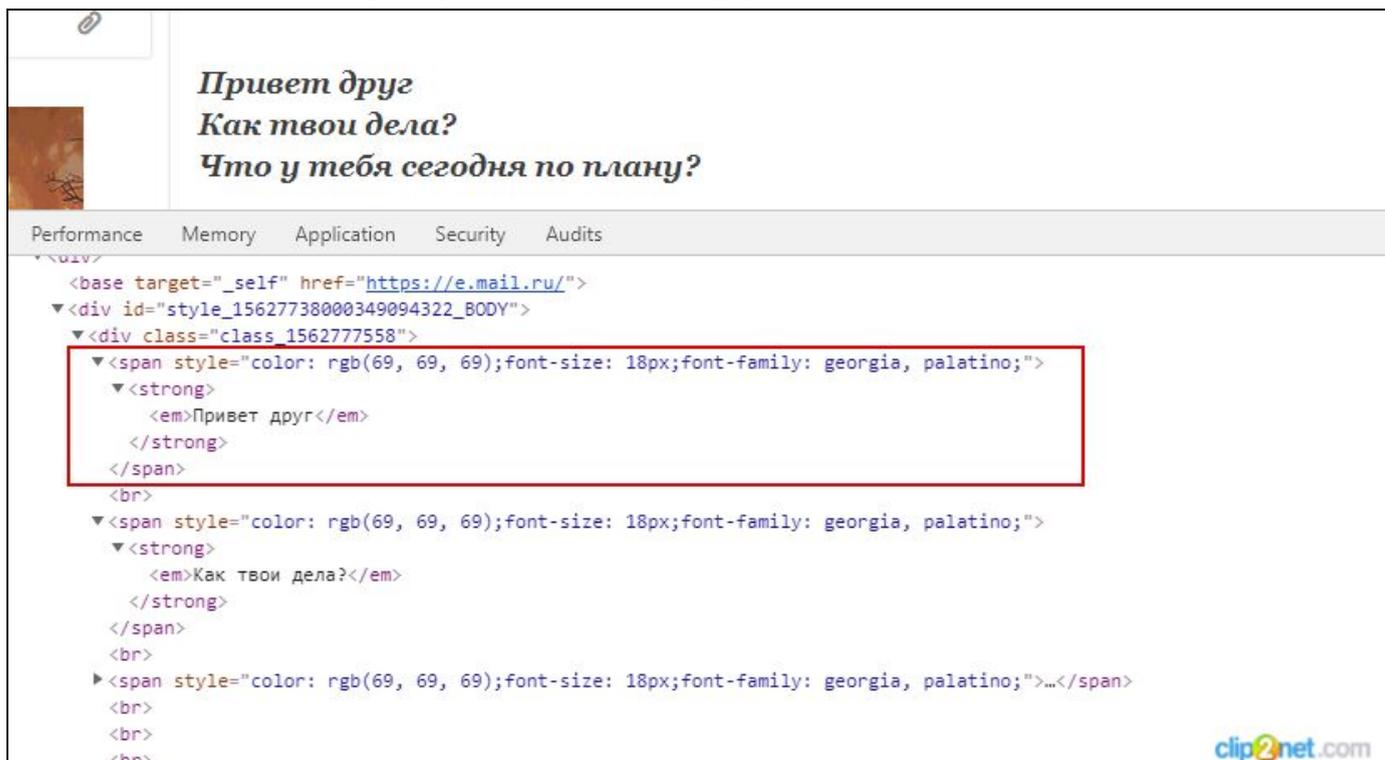


Рис #3

В конечном итоге, мы для себя выясним, что при создании шаблона, для формирования строк текста, мы можем использовать базовые теги и вот пример:

```
<p>тестовый текст # 1</p>
<p>тестовый текст # 2</p>
<p>тестовый текст # 3</p>
```

```
<div>тестовый текст # 1</div>
<div>тестовый текст # 2</div>
<div>тестовый текст # 3</div>
```

```
<span>тестовый текст # 1</span><br>
<span>тестовый текст # 2</span><br>
<span>тестовый текст # 3</span>
```

Можно не каждую строку заворачивать в теги, а весь текст, а строки разделять тегом `<br>`

```
<p>тестовый текст # 1
<br>
тестовый текст # 2
<br>
тестовый текст # 3</p>
```

```
<div>тестовый текст # 1
<br>
тестовый текст # 2
<br>
тестовый текст # 3</div>
```

```
<span>тестовый текст # 1
<br>
тестовый текст # 2
<br>
тестовый текст # 3</span>
```

И самое главное, что все это можно перемешивать и структурировать:

```
<p><span>тестовый текст # 1</span></p>
<br>
<div><span>тестовый текст # 1</span></div>
```

Также, хорошим решением будет создать данные структуры и поместить в один шаблон, рандомизируя между собой. Это поможет минимизировать создание сигнатуры. Получается у вас как бы один шаблон, но в нем несколько структур для макетов, которые рандомизируются от письма к письму.

```
{random}
<p>тестовый текст # 1</p>
<p>тестовый текст # 2</p>
<p>тестовый текст # 3</p>
|
<div>тестовый текст # 1</div>
<div>тестовый текст # 2</div>
<div>тестовый текст # 3</div>
|
<span>тестовый текст # 1</span><br>
<span>тестовый текст # 2</span><br>
<span>тестовый текст # 3</span>
{/random}
```



Рис #4

Используйте шорткоды для создания структур. Это позволит шаблон сделать минималистическим и понятным, при этом очень рандомизированным.

Отличным решением будет поместить в шорткоды теги стилизации `<em>`, `<strong>`, `<i>` и тд. А для того, чтоб открывающийся тег всегда совпадал с закрывающимся, используйте атрибут `id` из макросов.

Ниже представлена строка текста, завернутая в шорткоды.

`{shortcode code='tags_open' id='tags01'}` тестовый текст `{shortcode code='tags_close' id='tags01'}`

Сами шорткоды необходимо создать в личном кабинете.

437	Текстовые теги Open	<code>{shortcode code='tags_open'}</code>	9	  
438	Текстовые теги Close	<code>{shortcode code='tags_close'}</code>	9	  



Рис #5

Краткое описание ⓘ

Текстовые теги Open

---

Код ⓘ

tags\_open

Только латинские буквы нижнего регистра, цифры и знак "\_". Без скобок. Код не должен повторяться. Пример: hello\_1

---

Подставляемые данные ⓘ

```
</>
<b>
<basefont>
<big>
<blockquote>
<cite>
<font>
<em>
<strong>
```

Значение на которое будет заменен шорткод в письме. Не больше 500 вариантов. Каждый с новой строки.

clip2net.com

Рис #6

На рисунке 6 показаны подставляемые данные в шорткод. Это открывающиеся теги.

Краткое описание ⓘ	Текстовые теги Close
Код ⓘ	tags_close <small>Только латинские буквы нижнего регистра, цифры и знак "_". Без скобок. Код не должен повторяться. Пример: hello_1</small>
Подставляемые данные ⓘ	<pre>&lt;/i&gt; &lt;/b&gt; &lt;/basefont&gt; &lt;/big&gt; &lt;/blockquote&gt; &lt;/cite&gt; &lt;/font&gt; &lt;/em&gt; &lt;/strong&gt;</pre> <small>Значение на которое будет заменен шорткод в письме. Не больше 500 вариантов. Каждый с новой строки.</small>

[clip2net.com](http://clip2net.com)

Рис #7

На рисунке 7 показаны подставляемые данные в шорткод. Это закрывающиеся теги и они в таком же порядке, как и шорткоде с рисунка 6.

## Табличная верстка

Более сложный, но не менее интересный вариант - табличная верстка. Именно тут можно воплотить все мечты рандомизации).

Для того, чтоб шаблон получился с валидным HTML, предлагаю начинать его делать с базовой структуры и по чуть чуть усложнять.

Далее, буду делать макет и объяснять что именно сделал и какой результат в итоге.



Рис #8

Это основа. Таблица на всю ширину экрана, с вложенной строкой и ячейкой. Атрибуты **cellpadding="0" cellspacing="0" border="0"** означают что не будет никаких границ по умолчанию. В теге **<td>** есть атрибут **align="left"** - означает, что весь последующий контент будет с левой стороны. Левое позиционирование. Под контентом тут подразумевается больше не сам текст, а последующая структура, все следующие таблицы будут прибиты к левому краю.

Далее, мы вставляем тег **<div>** и задаем ему ширину в 600px. Означает, что максимальная ширина нашего письма будет 600px.

Внутри данного тега, делаем таблицу с вложенной строкой и ячейкой. Уже внутри их будет контент. Хочу заметить, что у данной таблицы ширина 100%, но она не более чем 600px, так как упирается в блок **<div>**.

Тег **<td>** имеет атрибуты, которые уже непосредственно влияют на сам контент:

**align="left"** - текст позиционируется слева

**font-size:16px** - размер шрифта

**line-height:150%** - межстрочный интервал от размера шрифта.

Внутри всего этого идет сам контент. Смотрите рисунок #9

```
<table width="100%" cellpadding="0" cellspacing="0" border="0" align="center">
  <tbody>
    <tr>
      <td align="left" valign="top">
        <div style="max-width:600px;display: inline-block;">
          <table align="left" border="0" cellspacing="0" cellpadding="0" style="width:100%">
            <tbody>
              <tr>
                <td valign="top" align="left" style="font-size:16px;line-height:150%">
                  тут текст
                </td>
              </tr>
            </tbody>
          </table>
        </div>
      </td>
    </tr>
  </tbody>
</table>
```



Рис #9

Если отправить письмо на почту, то мы увидим исключительно текст.

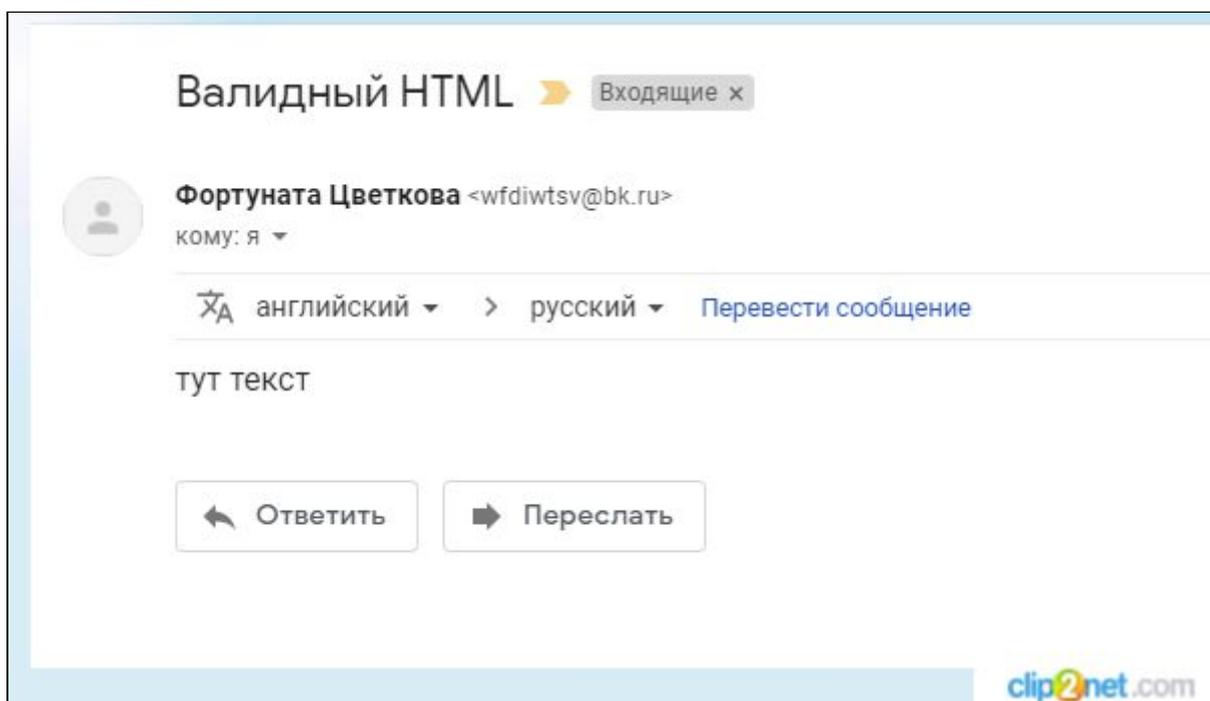


Рис #10

А если посмотреть в источник письма, в сам код, то там можно увидеть нашу табличную верстку.

Валидный HTML Входящие x

**Фортуната Цветкова** <wfdiwtsv@bk.ru>  
кому: я

английский > русский [Перевести сообщение](#)

тут текст

[← Ответить](#) [➡ Переслать](#)

```
<div id=":ltg" class="a3s aXjCH ">
  <div class="adM">
    </div>
  <div>
    <table width="100%" cellpadding="0" cellspacing="0" border="0" align="center">
      <tbody>
        <tr>
          <td align="left" valign="top">
            <div style="max-width:600px">
              <table align="left" border="0" cellspacing="0" cellpadding="0" style="width:100%">
                <tbody>
                  <tr>
                    <td valign="top" align="left" style="font-size:16px;line-height:150%">
                      тут текст
                    </td>
                  </tr>
                </tbody>
              </table>
            </div>
          </td>
        </tr>
      </tbody>
    </table>
  </div>
  <div class="yj6qo"></div>
  <div class="adL">
    </div>
  </div>
</div>
<div id=":id" class="ii et" style="display:none">.</div> == $0
```

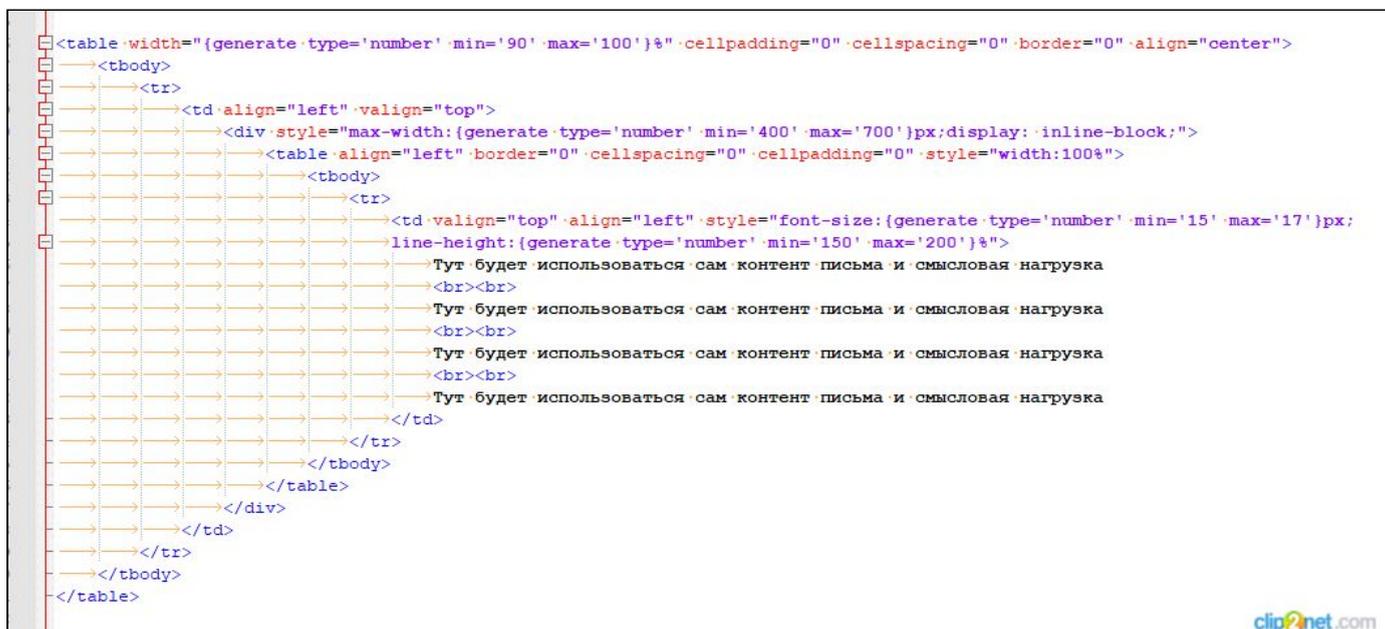
clip2net.com

Рис №11

Ну что же, продолжаем...

Теперь, когда мы создали шаблон с валидным HTML в базовой версии, мы его будем рандомить. Все что можно, но так, чтоб визуально все письма оставались примерно одинаковыми по дизайну.

1. В первом теге `<table>` в атрибуте **width** сделаем рандом от 90 до 100%. Добавляем макрос `{generate type='number' min='90' max='100'}`
2. В теге `<div>` мы сделаем рандом ширины, от 400px до 700px, используя макрос `{generate type='number' min='400' max='700'}`
3. В теге `<td>` изменим размер шрифта от 15px до 17px используя макрос `{generate type='number' min='15' max='17'}`, а так же, изменим межстрочный интервал от 150% до 200% используя макрос `{generate type='number' min='150' max='200'}`



```
<table width="{generate type='number' min='90' max='100'}%" cellpadding="0" cellspacing="0" border="0" align="center">
  <tbody>
    <tr>
      <td align="left" valign="top">
        <div style="max-width:{generate type='number' min='400' max='700'}px;display: inline-block;">
          <table align="left" border="0" cellspacing="0" cellpadding="0" style="width:100%">
            <tbody>
              <tr>
                <td valign="top" align="left" style="font-size:{generate type='number' min='15' max='17'}px;
                  line-height:{generate type='number' min='150' max='200'}%">
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                <br><br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                <br><br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                <br><br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                </td>
              </tr>
            </tbody>
          </table>
        </div>
      </td>
    </tr>
  </tbody>
</table>
```

Рис #12

Мы можем увидеть, что теперь наша структура имеет хороший рандом, а письмо при этом выглядит визуально одинаково и лаконично, как будто написано пользователем.

Предлагаю добавить еще немного интересных атрибутов для рандомизации

1. В тег `<div>` добавляем атрибут `padding` и значения от 1px до 15px делаем макросом `{generate type='number' min='1' max='15'}`
2. В тег `<div>` добавляем атрибут `margin` и значения от 1px до 15px делаем макросом `{generate type='number' min='1' max='15'}`

```
<table width="{generate.type='number'.min='90'.max='100'}%" cellpadding="0" cellspacing="0" border="0" align="center">
  <tbody>
    <tr>
      <td align="left" valign="top">
        <div style="max-width:{generate.type='number'.min='400'.max='700'}px;display:inline-block;
          padding:{generate.type='number'.min='1'.max='15'}px;margin:{generate.type='number'.min='1'.max='15'}px">
          <table align="left" border="0" cellspacing="0" cellpadding="0" style="width:100%">
            <tbody>
              <tr>
                <td valign="top" align="left" style="font-size:{generate.type='number'.min='15'.max='17'}px;
                  line-height:{generate.type='number'.min='150'.max='200'}%">
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                  <br><br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                  <br><br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                  <br><br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                </td>
              </tr>
            </tbody>
          </table>
        </div>
      </td>
    </tr>
  </tbody>
</table>
```

clip2net.com

Рис #13

В таком виде у нас получается простое и красивое письмо, которое имеет валидный HTML и отображается нормально на всех устройствах и браузерах.

Такой шаблон можно использовать для рассылки текста с прикрепленным файлом PDF.

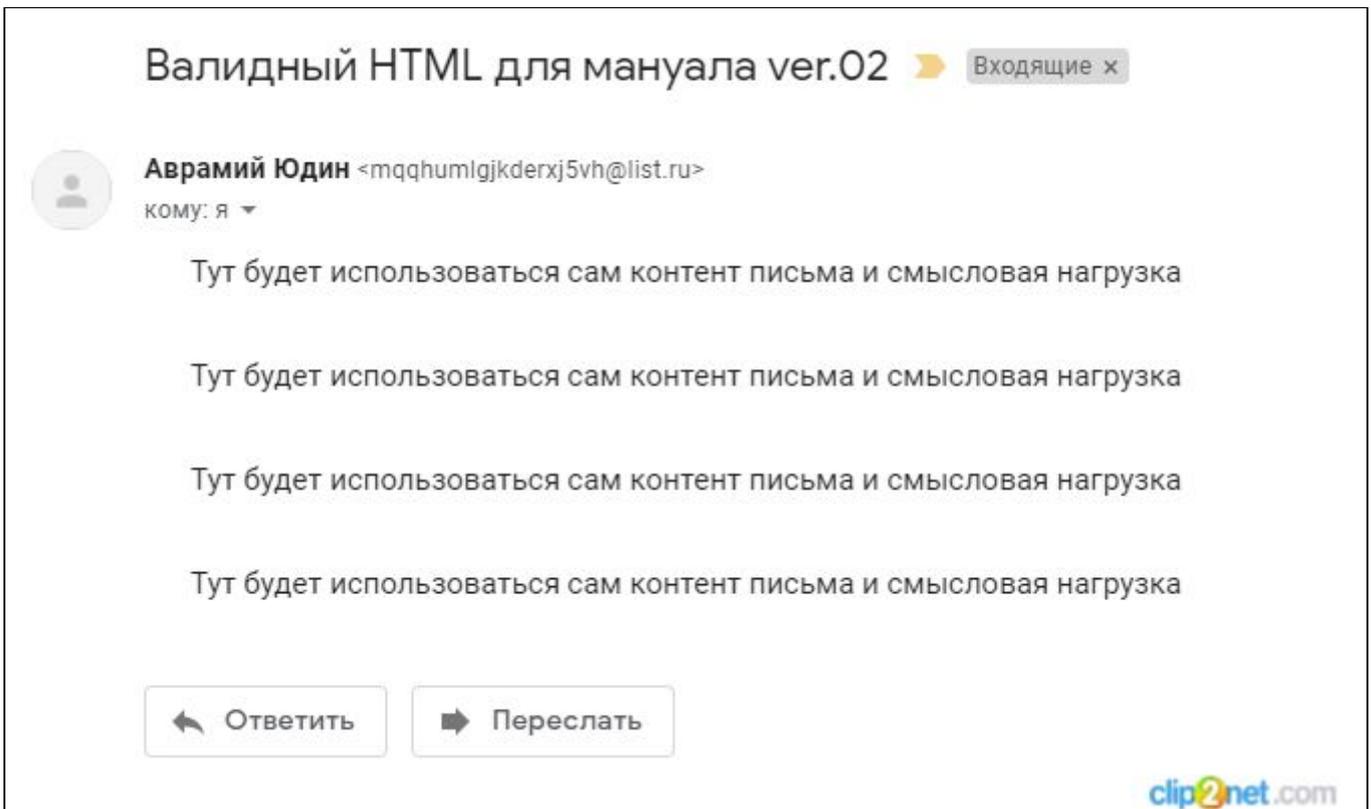


Рис #14

При этом, если посмотреть в код, то можем увидеть, что значения в атрибутах имеют случайный порядок от письма к письму, в рамках наших заданных значений.

```
▼<div id=":lxd" class="ii gt">
  ▼<div id=":lva" class="a3s axjCH ">
    <div class="adM">
    </div>
    ▼<div>
      ▼<table width="97%" cellpadding="0" cellspacing="0" border="0" align="center">
        ▼<tbody>
          ▼<tr>
            ▼<td align="left" valign="top">
              ▼<div style="max-width:546px">
                ▼<table align="left" border="0" cellspacing="0" cellpadding="0" style="width:100%">
                  ▼<tbody>
                    ▼<tr>
                      ▶<td valign="top" align="left" style="font-size:15px;line-height:175%">_</td>
                    </tr>
                  </tbody>
                </table>
              </div>
            </td>
          </tr>
        </tbody>
      </table>
    </div>
    <div class="yj6qo"></div>
    <div class="adL">
    </div>
  </div>
  </div>
  ▶<div id=":lwp" class="ii gt" style="display:none">_</div>
  <div class="hi"></div>
```



Рис #15

## Табличная верстка корпоративного стиля

Хорошим решением будет сделать корпоративный макет, с более красочным HTML, но при этом конечно же валидным.

Называю его именно так (корпоративный стиль), потому что многие компании используют за основу такие макеты.

И так, изменим наш шаблон немного..

В тег `<table>` добавим атрибут **bgcolor**. Это будет общий фон самого письма и сделаем его в различных тонах серого цвета. Саму стилизацию мы сделаем через шорткод и назовем его:

```
{shortcode code='bg_grey_letters'}
```

Обратите внимание на содержимое шорткода и в каком формате там цвета.

Краткое описание ⓘ Серый фон для макета Table

Код ⓘ `bg_grey_letters`

Только латинские буквы нижнего регистра, цифры и знак "\_". Без скобок. Код не должен повторяться. Пример: hello\_1

Подставляемые данные ⓘ

```
#cccc  
#d1d1d1  
#d6d6d6  
#eaeaea  
#e3e3e3  
#e5e5e5  
#e8e8e8  
#eeeeee  
#f3f3f3  
#f9f9f9  
#ebebeb  
#efefef  
#f4f4f4  
#f8f8f8  
#fcfcfc  
#d3d3d3  
#d7d7d7
```

Значение на которое будет заменен шорткод в письме. Не больше 500 вариантов. Каждый с новой строки.



clip2net.com

Рис #16

Продолжаем вносить ряд изменений

1. В теге `<td>` изменим содержимое атрибута **align** на **center**.
2. В теге `<div>` содержимое в атрибутах **padding** и **margin** изменим от 15px до 25px
3. В теге `<div>` добавим атрибут **border** в 1px и сплошной линией **solid**. Цвет же для этого атрибута, мы сделаем используя наш макрос `{shortcode code='bg_grey_letters'}`.
4. В теге `<div>` добавим скругление углов для нашего **border**, используя атрибут **border-radius**, а значения для сделаем макросом `{generate type='number' min='2' max='10'}`
5. А для того чтоб выделить теперь поле, где будет сам текст, добавим атрибут **background-color:#fffff;**

Смотрим что у нас получилось в шаблоне

```
<table width="{generate.type='number' min='90' max='100'}%" cellpadding="0" cellspacing="0" border="0" align="center" bgcolor="{shortcode code='bg_grey_letters'}">
  <tbody>
    <tr>
      <td align="center" valign="top">
        <div style="max-width:{generate.type='number' min='400' max='700'}px;display: inline-block;
          padding:{generate.type='number' min='15' max='25'}px;margin:{generate.type='number' min='15' max='25'}px;
          border: 1px solid {shortcode code='bg_grey_letters'};border-radius:{generate.type='number' min='2' max='10'}px;
          background-color:#ffffff;">
          <table align="left" border="0" cellspacing="0" cellpadding="0" style="width:100%">
            <tbody>
              <tr>
                <td valign="top" align="left" style="font-size:{generate.type='number' min='15' max='17'}px;
                  line-height:{generate.type='number' min='150' max='200'}%">
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                <br><br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                <br><br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                <br><br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                <br><br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                </td>
              </tr>
            </tbody>
          </table>
        </div>
      </td>
    </tr>
  </tbody>
</table>
```

Рис #17

В итоге мы получаем макет, где используется случайный серый цвет, разная ширина всех таблиц. Блок, где написан текст имеет рамку серого цвета и скругления, при этом сам фон белого цвета. Присутствуют внешние и внутренние отступы.

Текст имеет различный размер и межинтервальные отступы.

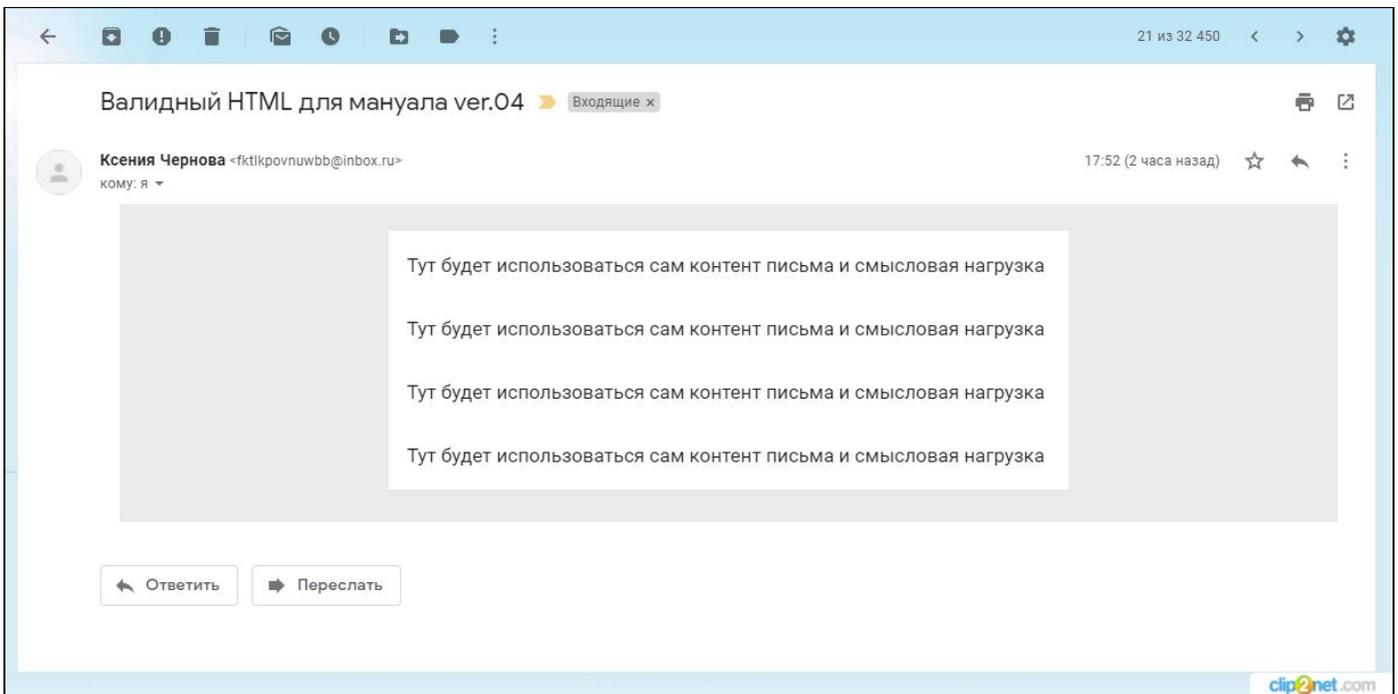


Рис #18

Для более правильного понимания, прилагаются дополнительные скриншоты для сравнения готовых макетов. Визуально можно увидеть отличия, но глобально выдержан одинаковый стиль.

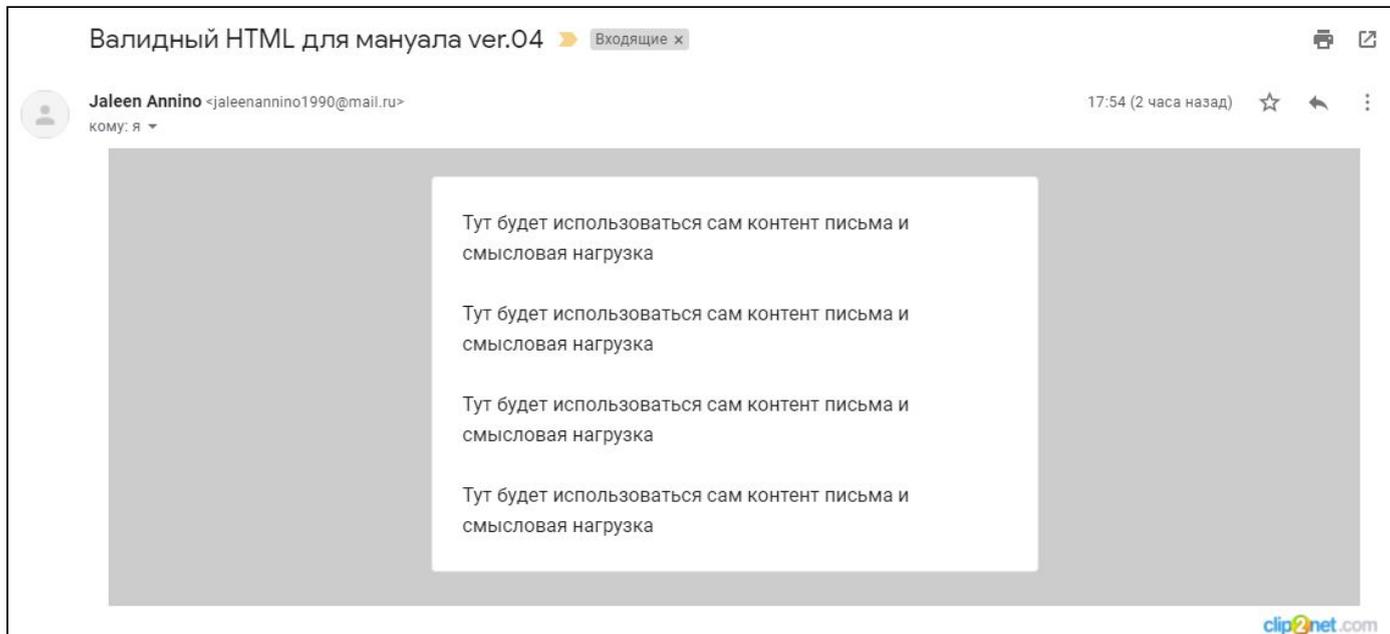


Рис #19

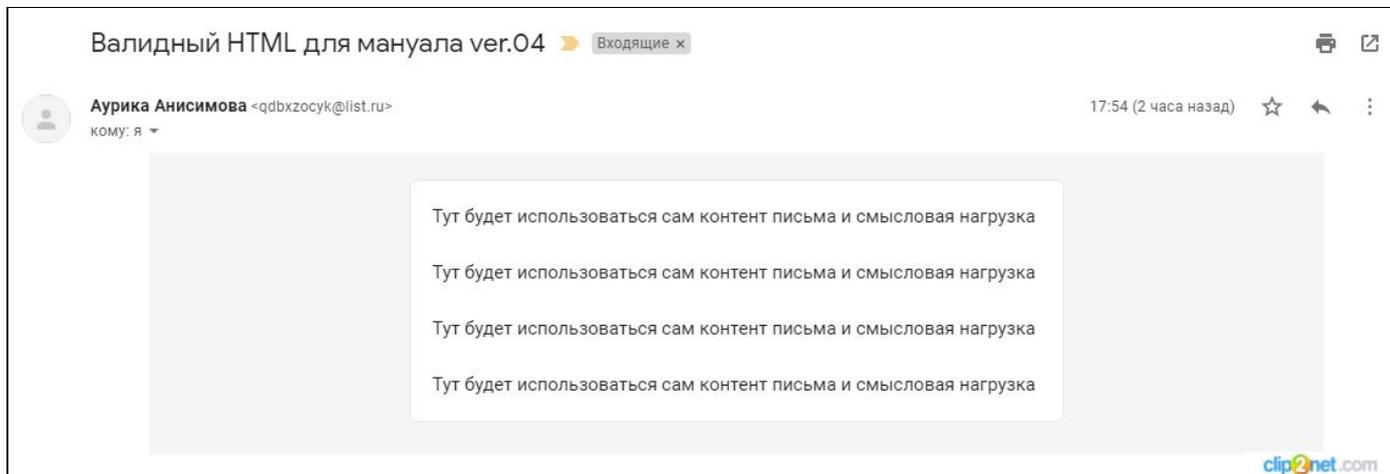


Рис #20

Если мы хотим использовать в письме ссылку, то предлагаю ее сделать красивой кнопкой. Для этого, в данный валидный HTML добавим тег **A** и сделаем ему красивый стиль используя ряд атрибутов и стилей.

1. Атрибут **display: block;**
2. Для того чтоб кнопка и текст были по центру добавим атрибуты **text-align: center;** и **margin: 0 auto;**
3. Добавим ширину кнопки с помощью атрибута **width** и содержимым с помощью макроса для него от 100px до 200px
4. Необходимо сделать отступы, чтоб кнопка была не маленькая, с помощью атрибута **padding** и макроса **{generate type='number' min='10' max='20'}**
5. Сделаем вокруг кнопки обводку, используя атрибут **border** и уже ранее используемый макрос с серыми цветами **{shortcode code='bg\_grey\_letters'}**
6. Добавим скругление углов для нашего border, используя атрибут **border-radius**, а значения для него сделаем макросом **{generate type='number' min='1' max='5'}**
7. Шрифт текста на кнопке сделаем белого цвета, используя атрибут **color** и содержимое **#ffffff;**
8. Саму кнопку сделаем разных цветов. Для этого необходимо будет создать макрос, с содержимым разных кодов разных цветов и конечно же добавить сам атрибут **background-color**

Смотрим пример кода:

```
<table width="{generate type='number' min='90' max='100'}%" cellpadding="0" cellspacing="0" border="0" align="center" bgcolor="{shortcode code='bg_grey_letters'}">
  <tbody>
    <tr>
      <td align="center" valign="top">
        <div style="max-width:{generate type='number' min='400' max='700'}px; display: inline-block;
          padding:{generate type='number' min='15' max='25'}px; margin:{generate type='number' min='15' max='25'}px;
          border: 1px solid {shortcode code='bg_grey_letters'}; border-radius:{generate type='number' min='2' max='10'}px;
          background-color:#ffffff;">
          <table align="left" border="0" cellspacing="0" cellpadding="0" style="width:100%">
            <tbody>
              <tr>
                <td valign="top" align="left" style="font-size:{generate type='number' min='15' max='17'}px;
                  line-height:{generate type='number' min='150' max='200'}%">
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                  <br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                  <br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                  <br>
                  Тут будет использоваться сам контент письма и смысловая нагрузка
                  <br>
                  <a href="" style="display: block; text-align: center; margin: 0 auto;
                    width:{generate type='number' min='100' max='200'}px;
                    padding:{generate type='number' min='10' max='20'}px;
                    border: 1px solid {shortcode code='bg_grey_letters'};
                    border-radius: {generate type='number' min='1' max='5'}px; color: #fff;
                    background-color: #f00;">Шорткод с анкором</a>
                </td>
              </tr>
            </tbody>
          </table>
        </div>
      </td>
    </tr>
  </tbody>
</table>
```

Рис #21

Смотрим полученный результат ...

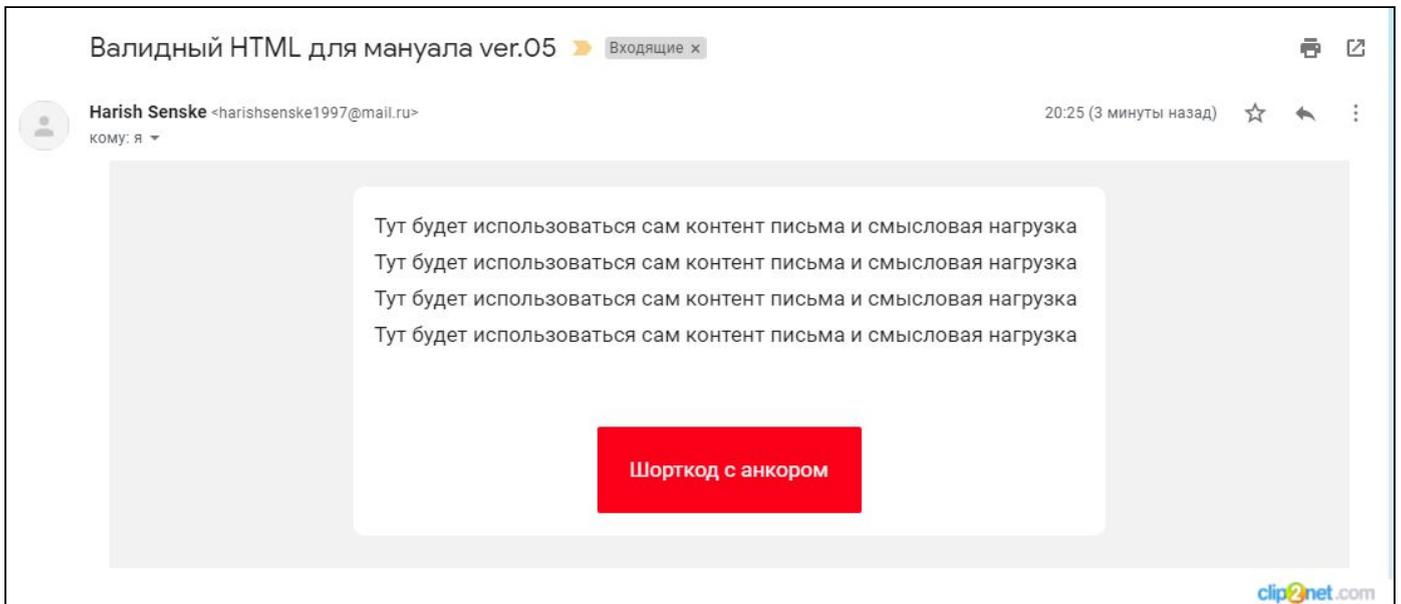


Рис #22

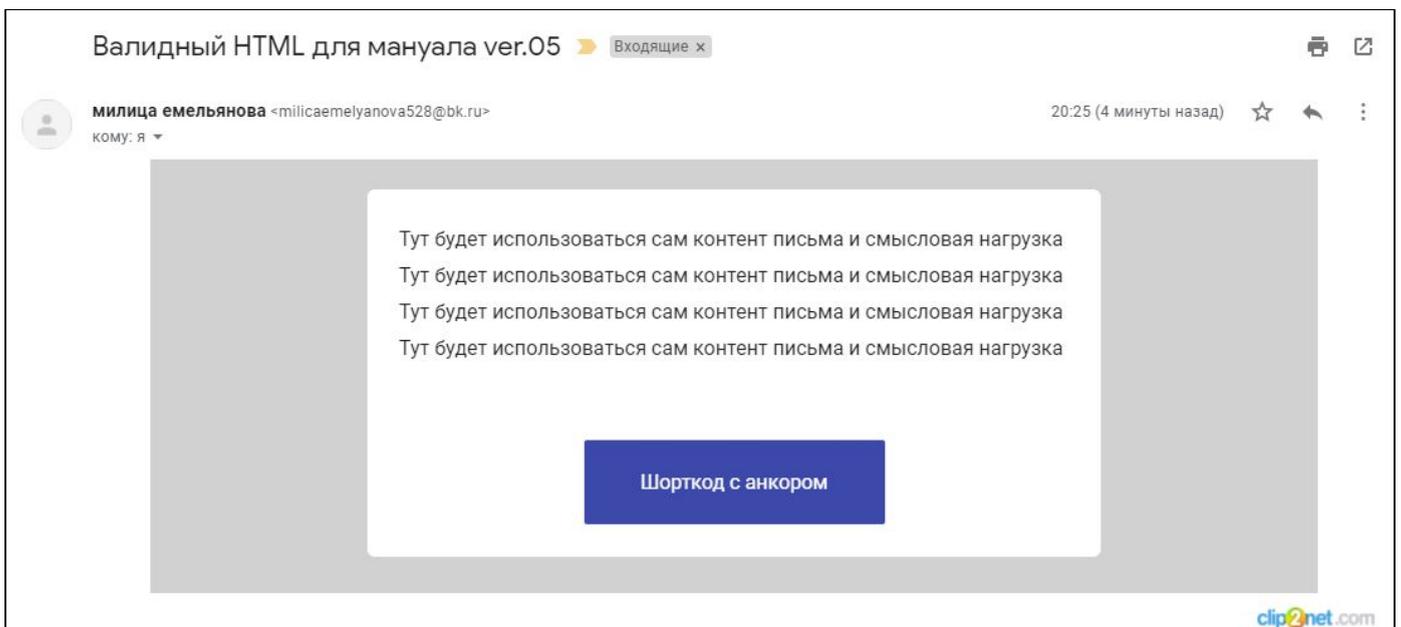


Рис #23

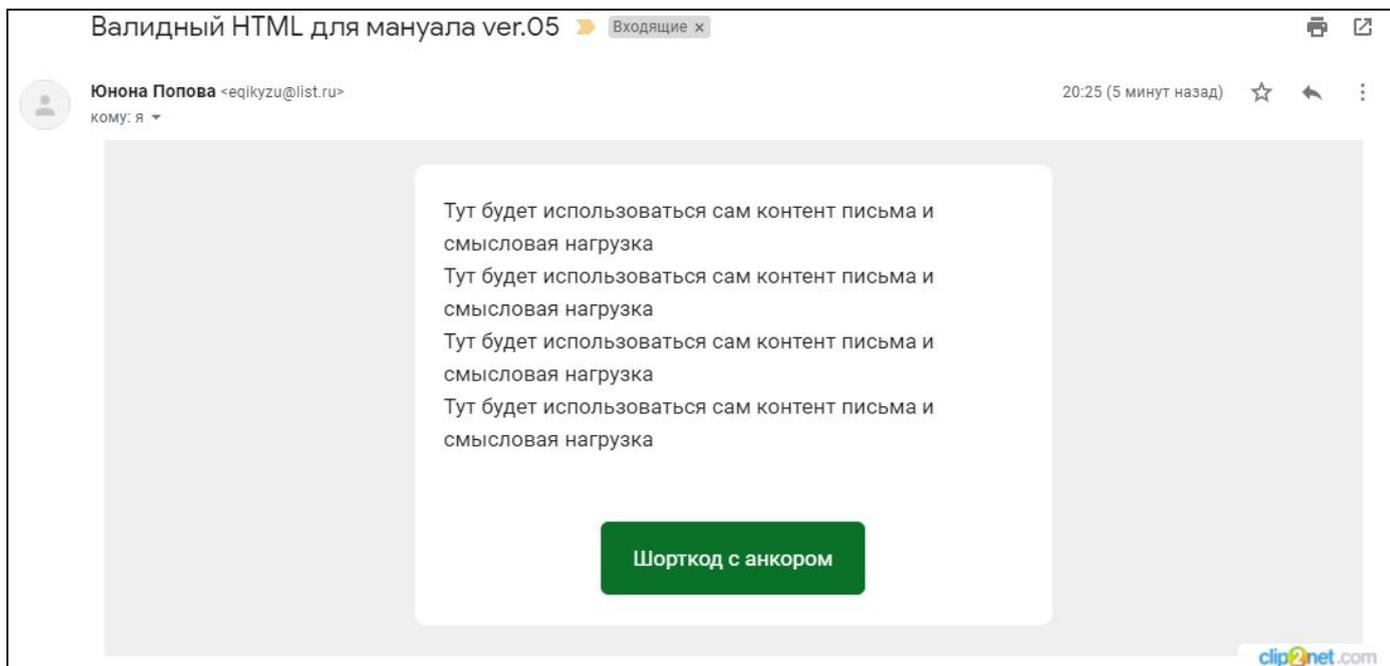


Рис #24

Вот сейчас наглядно видно, что стилистика сохранилась, а HTML круто рандомится.

P.S. Через данный мануал нельзя передать саму структуру кода и готовый HTML данных макетов, но если, вы клиент сервиса Webimitation, то всегда сможете запросить у нас и получите исходные файлы.

Используйте данный шаблон в своих рассылках, усовершенствуйте его и улучшайте. И обязательно оставьте обратную связь по полученным данным администратору сервиса, ведь мы же старались;)