

## Динамический контент в рассылках. Ленты новостей

В платформе Sensay для email-рассылок можно настроить автоматическое формирование выпусков с использованием динамического контента, в частности, лент новостей.

Лента новостей - это набор от одного до бесконечности каналов новостей, сообщения из которых собираются и могут быть использованы для формирования индивидуальных писем или общих рассылок.

### Доступные источники новостей для лент новостей

**RSS и Atom каналы** - задаются обычной ссылкой (*http,https,ftp,ftps*)

**Поиск по блогам Google** - задаётся ссылкой вида *search://gblogs/поисковый-запрос*

**Поиск по блогам Яндекс** - задаётся ссылкой вида *search://yblogs/поисковый-запрос*

**Поиск по Twitter** - задаётся ссылкой вида *search://tsearch/поисковый-запрос*

**LiveJournal пользователя** - Лента записей пользователя задаётся ссылкой вида *lj://user/ник-пользователя*

**LiveJournal сообщества** - Лента записей сообщества задаётся ссылкой вида *lj://user/ник-сообщества*

**Twitter** - Лента записей пользователя задаётся ссылкой вида *tw://id-пользователя*

**Facebook пользователя или сообщества** - Лента записей пользователя или сообщества задаётся ссылкой вида *fb://id-пользователя-или-сообщества*

**Facebook друзей пользователя** - Лента записей друзей пользователя задаётся ссылкой вида *fb://id-пользователя/friends*

**ВКонтакте пользователя или сообщества** - Лента записей пользователя или сообщества задаётся ссылкой вида *vk://id-пользователя-или-сообщества*

**ВКонтакте друзей пользователя** - Лента записей друзей пользователя задаётся ссылкой вида *vk://id-пользователя/friends*

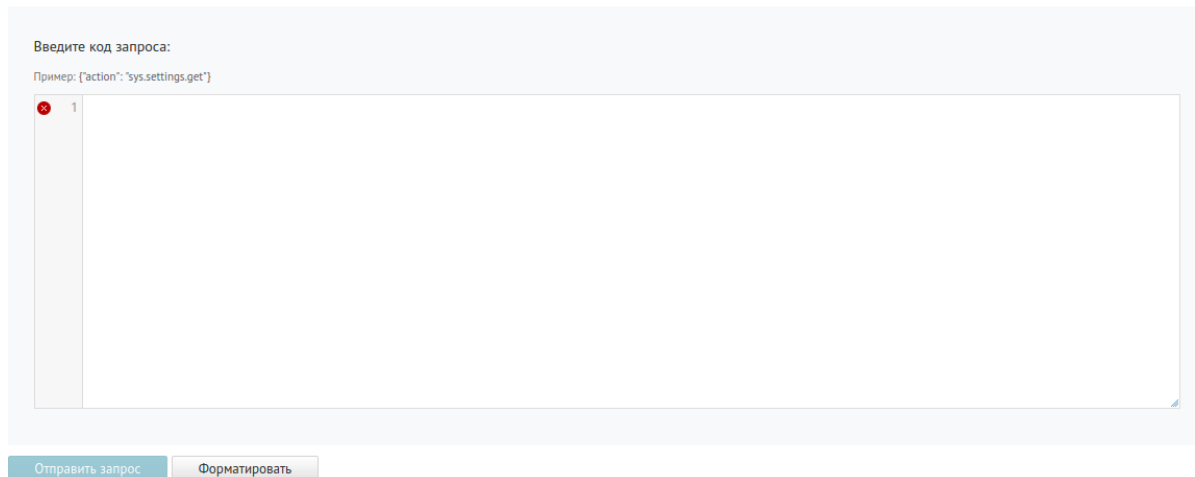
**Я.Ру пользователя или сообщества** - Лента записей пользователя или сообщества задаётся ссылкой вида *ya://ник-пользователя-или-сообщества*

**Я.Ру друзей пользователя** - Лента записей друзей пользователя задаётся ссылкой вида *ya://ник-пользователя/friends*

**МойМир пользователя или сообщества** - Лента записей пользователя или сообщества задаётся ссылкой вида *mt://id-пользователя-или-сообщества*

Все действия с Лентами новостей выполняются через АПИ-консоль  
([https://sendsay.ru/account/#system\\_api\\_test](https://sendsay.ru/account/#system_api_test))

API-Консоль



## Создание ленты новостей

Для создания ленты новостей необходимо в АПИ-консоли выполнить следующий запрос:

```
{
  "action" : "lenta.set"
, "obj" : {
    "name" : "название ленты"
    , "member" : "адрес получателя" -- только при создании. изменить нельзя
-- для лент привязанных к получателю
    , "decor" : "идентификатор информационного письма или черновика выпуска"
    -- расписание выпусков
    -- если параметр полностью отсутствует, то с 8 по 18 с понедельника по
пятницу
    , "schedule" : {
      -- часы по которым высылать письма. если не указано, то с
8 по 18
      "hour" : [ час, час, ... ]
      -- номера дней недели по которым высылать письма, если не
указано, то с Пн по Пт
      -- от 1 (Пн) до 7 (Вс)
```

```
        , "weekday" : [ номер, номер, ... ]
      }
    }
  }
-- необязательные
, "id" : номер ленты
, "return_fresh_obj": "" -- вернуть объект в формате lenta.get
}
```

Введите код запроса:

Пример: {"action": "sys.settings.get"}

```
1 {
2   "action" : "lenta.set"
3
4   , "obj" : {
5     "name" : "LOST" }}
6
7
```

Отправить запрос

Форматировать

**ВАЖНО!** Для создания индивидуальной для каждого подписчика ленты необходимо указать остальные параметры в запросе, включая id черновика, высылаемого письма!

## Добавление источника новостей в ленту

**Важно!** Одна лента новостей может содержать сколько угодно источников новостей.

Для добавления источника в ленту необходимо в АПИ консоли выполнить следующий запрос:

```
{
  "action" : "lenta.source.add"
  , "lenta.id" : номер ленты
  , "url" : "адрес источника добавляемого в ленту"
}
```

Введите код запроса:

Пример: ['action': 'sys.settings.get']

```
1 {
2
3   "action" : "lenta.source.add"
4
5   , "lenta.id" : 1
6
7   , "url" : "http://www.lostfilm.tv/rssdd.xml"
8
9 }
```

Остальные команды для лент новостей можно посмотреть тут:

<https://sendsay.ru/api/api.html#Ленты-Новостей>

### **Важная информация!**

**Один источник новостей может входить в несколько лент** - фактически он опрашивается один раз в 1-2 часа и все найденные свежие новости собираются в базу и становятся независимо доступны во всех лента и использование определенной новости в одной из них никак не повлияет на ее состояние в другой ленте.

**Одна и та же лента может использоваться в выпуске несколько раз.** Например первый раз для формирования "меню" из списка заголовков новостей, а второй для вывода полной новости.

**В выпуске может быть использовано несколько лент.**

**Лента хранит отметку времени последнего использования в рассылке** и основываясь на нём система включает в письма только те новости, которые были получены после этой даты.

**После формирования рассылки (или индивидуального письма)** отметка времени у всех использованных в рассылке лент устанавливается в текущее время. Таким образом новости одной ленты уже использованные в рассылке повторно использованы не будут.

**Старые новости регулярно удаляются из базы** как только проходит 15 дней с момента получения новости.

## Использование лент новостей в Индивидуальных письмах

Индивидуальные письма подписчикам формируются из новостей накопившихся в их личных лентах которые к подписчикам привязаны владельцем при их создании (см. [Создание ленты новостей](#)). Письмо формируется по указанному в ленте расписанию с оформлением по указанному шаблону выпуска с использованием новостей накопившихся с прошлой рассылки. **Если свежих новостей нет, то письмо не высылается.**

*Это позволяет организовать индивидуальное оповещение клиентов персонализированными подборками информации. Например, организовать на сайте подписку на результаты поиска по каталогу товаров - пользователи будут автоматически получать обновления в интересующих их разделах.*

## Использование лент новостей в рассылках

Для использования в рассылке новостей одной или нескольких лент надо воспользоваться функциям шаблонизатора `news_get()` или `lenta_get()`.

```
[% news_get(id ленты) %]  
[% news_get(URL) %]
```

```
[% lenta_get(id ленты) %]  
[% lenta_get(URL) %]
```

Функция `news_get()` не требует специальной обработки и результатом её работы становится содержимо полей "content" всех свежих новостей. Это упрощает работу с лентами если готового источника у вас ещё нет и вы можете легко формировать его как нравится. Или когда имеющийся источник выдаёт в "content" новости то, что надо для выпуска.

Функция `lenta_get()` только присваивает переменной `lenta` специальную структуру данных описывающую все свежие на данный момент новости и их источники. Для их отображение требуется использование в теле письма оператора `[% FOREACH %]` для обработки каждой новости. Это сложнее, но вы можете оформить вывод новостей самым гибким образом.

Обе функции могут работать как с заранее созданной лентой новостей (указывается её номер), так и напрямую с каналом новостей (указывается его `url`).

**При указании `url` механизм исключения уже использованных новостей не применим** и в выпуск всегда попадают все новости канала имеющиеся в нём на момент выпуска.

**Если свежих новостей нет** ни в одной используемой в шаблоне ленте и новостей нет вообще ни в одном используемом в шаблоне источнике указанным напрямую, **то выпуск рассылки не высылается.**

По умолчанию новости отсортированы в порядке убывания и даты публикации. Задать свой порядок и группировку по источникам можно с помощью шаблонизатора - рассмотрите

пример "[Новости по источникам](#)"

## Проверка содержимого письма с лентой новостей (аналогично тестовым копиям)

Вызов **lenta.send** отправляет тестовое письмо с указанным количеством последних новостей ленты на указанный адрес (а не на адрес для которого создана лента). В отличие от реального выпуска, отосланные новости не отмечаются как использованные и потом будут в итоге использованы при реальной отсылке. В реальном выпуске новости отмечаются как использованные и если лента не обновилась, письмо пустое будет приходить.

Для отправки тестовой копии письма с содержимым ленты новостей в АПИ консоли необходимо задать следующий вызов:

```
{ "action" : "lenta.send"
, "id" : номер ленты
, "email" : адрес для отсылки тестового письма
, "n" : количество последних новостей для теста
      -- если не указано, то используются все ещё не отосланные новости (их
      может не быть)
, "decor" : "черновик" -- если лента привязана к подписчику, то можно не
      указывать - будет взят черновик указанный в ленте
      -- если лента не привязана к подписчику, то черновик
      указать необходимо
}
```

Введите код запроса:

Пример: {"action": "sys.settings.get"}

```
1 { "action" : "lenta.send"
2
3 , "id" : 1
4
5 , "email" : "ask@sendsay.ru"
6
7
8 , "decor" : "9876"
9 }
```

## Структура переменной `lenta` (для понимания команд шаблонизатора в макетах)

Структура данных ленты в индивидуальных письмах и после вызовов `lenta()` и `news()` содержит информации о ленте новостей, всех источниках новостей новости которых попали в ленту и содержимое всех свежих новостей.

### Лента новостей

```
"lenta" : {
  "id" : "id ленты в базе"

  , "name" : "название ленты"

  -- описания всех источников ленты, чьи новости попали в письмо
  , "sources" : {
    "id-источника" : { описание источника }
    , "id-источника" : { описание источника }
    .....
  }

  -- id источников упорядоченные так же как упорядочены новости
  , "sources_order" : [ "id-источника", "id-источника" ..... ]

  -- описание новостей в заказаном порядке (по умолчанию по убыванию даты
  публикации новости)
  , "news" : [
    { описание новости }
    , { описание новости }
    , { описание новости }
  ]
}
```

### Описание одного источника

```
{
  "id" : "id источника в базе"

  -- индексы в lenta.news[] новостей именно этого источника в порядке
  совпадающем с порядком lenta.news[]
  , "news" : [ индекс, индекс, индекс .... ]

  -- параметры источника как получены из описания RSS/Atom-канала при его
  последнем анализе
  -- названия параметров совпадают с названиями тегов XML

  , "title" : "название источника"

  , "descr" : "описание источника"

  , "link" : "ссылка на обычную версию источника"

  , "logo" : "ссылка на логотип источника"
}
```

```
, "author" : "информация об авторе"
, "copyright" : "информация об авторских правах"
}
```

### Описание одной новости

```
{
  "ID" : "id новости в базе"

  , "source" : "id в базе источника из которого получена новость. ключ в
  lenta.sources{}"

  , "n" : "номер по порядку в lenta.news[]"

  -- параметры новости как получены из описания RSS/Atom-канала при его
  последнем анализе
  -- названия параметров в основном совпадают с названиями тегов XML

  , "title" : "название источника",

  , "content" : "содержимое новости" -- тег <description> или <yandex:full-text>

  , "link" : "ссылка на обычную версию новости" -- тег <link>, если пусто и
  isPermaLink="true" и guid похож на урл, то guid

  , "dt" : "YYYY-MM-DD hh:mm:ss" -- дата публикации новости (тег <pubDate> или
  <dc:creator>)

  , "id" : "идентификатор новости" -- тег <guid>

  , "author" : "информация об авторе новости"

  , "category" : { -- категории новости
    "домен" : {
      "категория" : "метка"
      , .....
    }
    , .....
  }

  , "enclosure" : [ -- приложения к новости на основании тегов <enclosure> и
  <media:thumbnail>
    -- обычно так указывают картинки и звуковые файлы
    {
      "url" : "ссылка"
      , "type" : "mime-type"
      , "length" : "длина"
    }
    , .....
  ]

  -- дополнительные параметры полученные из dt

  , "ymd" : "YYYY-MM-DD"

  , "hm" : "hh:mm"
```



```
}
```

**Примеры команд шаблонизатора для вставки в ваши письма  
(используются как в индивидуальных шаблонах, так и в шаблонах для  
массовых отправлений)**

[Не забывайте в начале шаблона инициализировать ленту командами,  
указанными выше!](#)

## Все новости ленты по порядку

```
<h1>Свежие новости ленты "[% lenta.name %]"</h1>

[% FOREACH news IN lenta.news %]

<p>
<a name="news_[% news.n %]"></a>
[% IF news.title == "" %]
[% IF news.link != "" %]
<a href="[% news.link %]"><b>[% news.link %]</b></a><br />
[% END %]
[% ELSE %]
[% IF news.link == "" %]
<b>[% news.title %]</b><br />
[% ELSE %]
<a href="[% news.link %]"><b>[% news.title %]</b></a><br />
[% END %]
[% END %]
<small>[% news.dt %]
[% IF news.author %] &ndash; [% news.author %][% END %]
[% SET srcid = news.source %]a
[% SET src = lenta.sources.$srcid %]
&ndash; [% src.title %]
</small><br />
[% IF news.content %]
[% news.content | raw %]
[% END %]
</p>
<div style="clear:both"></div>

[% END %]
```

## Все новости ленты с оглавлением и группировкой по источникам

```
<h1>Свежие новости ленты "[% lenta.name %]"</h1>

[% IF lenta.sources_order.size() > 1 %]
<ul>
[% FOREACH srcid IN lenta.sources_order %]
  [% SET src = lenta.sources.$srcid %]
```



```
<li> [% src.news.size() %] - <a href="#src_[% srcid %]">[% src.title %]</a>
[% END %]
</ul>
[% END %]
```

```
[% FOREACH srcid IN lenta.sources_order %]
```

```
<a name="src_[% srcid %]"></a>
[% SET src = lenta.sources.$srcid %]
[% IF src.link == "" %]
<h2>[% src.title %] ([% src.news.size() %])</h2>
[% ELSE %]
<h2><a href="[% src.link %]"><b>[% src.title %]</b></a> ([% src.news.size()
%])</h2>
[% END %]
```

```
[% FOREACH newsid IN src.news %]
[% SET news = lenta.news[$newsid] %]
```

```
<p>
<a name="news_[% news.n %]"></a>
[% IF news.title == "" %]
[% IF news.link != "" %]
<a href="[% news.link %]"><b>[% news.link %]</b></a><br />
[% END %]
[% ELSE %]
[% IF news.link == "" %]
<b>[% news.title %]</b><br />
[% ELSE %]
<a href="[% news.link %]"><b>[% news.title %]</b></a><br />
[% END %]
[% END %]
<small>[% news.dt %]
[% IF news.author %] &ndash; [% news.author %][% END %]
</small><br />
[% IF news.content %]
[% news.content | raw %]
[% END %]
</p>
<div style="clear:both"></div>
```

```
[% END %]
[% END %]
```