

Поиск по каталогу

В HostCMS в редакции "Халява" функция поиска не работает. Здесь приведен способ реализации этой функции. В данном случае поиск работает как фильтрация по товарам.

Страница поиска по каталогу аналогична странице каталога /shop/. Поиск работает с тем же XSL шаблоном, что и каталог.

ТДС интернет магазина возвращает все товары, если это каталог, или товары, отфильтрованные по условию поиска, если это страница поиска.

ТДС интернет магазина, помимо данных для вывода товаров, возвращает переменную `isSearch` со значениями 0, если это каталог, и 1, если это страница поиска

В XSL шаблон необходимо внести условие, которое при помощи переменной `isSearch` будет определять, отображать элементы для каталога или для страницы поиска. В частности, вывести название страницы "КАТАЛОГ" или строку поиска и название страницы "ПОИСК". Такие элементы необходимо обернуть в конструкции:

```
<!-- Если каталог то выводим название каталога -->
<xsl:if test="isSearch = 0">
  <h1>
    <xsl:value-of disable-output-escaping="yes" select="name"/>
  </h1>
</xsl:if>

<!-- Если поиск то выводим форму поиска-->
<xsl:if test="isSearch = 1">
  <h1>
    <xsl:text>Поиск</xsl:text>
  </h1>
  <!-- Форма поиска -->
  <form id="search_mini_form" method="get" action="/shop/">
    <!------->
  </form>
</xsl:if>
```

также, в xsl шаблоне необходимо обернуть в условия функции, реализующие пагинацию.

```
<!-- Ставим ссылку на страницу-->
<li>
  <xsl:choose>
    <xsl:when test="isSearch=0">
      <!-- Выводим ссылки на видимые страницы для каталога-->
      <a
href="{ $group_link } { $number_link } { $tag_path } { $shop_producer_path } { $filter } { $
on_page }" class="page_link">
        <xsl:value-of select="$i + 1"/>
      </a>
    </xsl:when>
```

```

<xsl:otherwise>
  <!-- Выводим ссылки на видимые страницы для поиска-->
  <a
href="{ $group_link } { $number_link } { $tag_path } { $shop_producer_path } { $filter } { $
on_page } ?search={ searchText }" class="page_link">
  <xsl:value-of select="$i + 1"/>
  </a>
</xsl:otherwise>
</xsl:choose>
</li>

```

Переходим в ТДС интернет магазина

Условие для поиска ТДС получает методом GET, например `example.ru/shop/?search=111`

В Коде страницы перед выводом контроллера показа добавляем:

1. Переменную `isSearch`, определяющую отображение страницы `/shop/` (каталог или поиск), по умолчанию = 0 (каталог)
2. Условие, переданы ли данные методом GET?
3. Если данные переданы, фильтруем элементы контроллера показа по заданному условию, переменной `isSearch` ставим значение 1 (поиск).

Пример кода:

```

/**
 * поиск по имени и артикулу
 *
 * @author Malinovskii Aleksandr, KAD Systems (©) 2015
 * @date 23-11-2015
 * Начало >>
 */
//переменная состояния, 0-каталог, 1-поиск
$isSearch = ;

if ($words = Core_Array::getGet('search')) {
  $Shop_Controller_Show->shopItems()
  ->queryBuilder()
  ->open()
  ->where('shop_items.name', 'LIKE', '%'.$words.'%')
  ->setOr()
  ->where('shop_items.marking', 'LIKE', '%'.$words.'%')
  ->close();

  $isSearch = 1;
}

$isSearch =
Core::factory('Core_Xml_Entity')->name('isSearch')->value($isSearch);
$Shop_Controller_Show->addEntity($isSearch);
/**

```

* << Конец

* @author Malinovskii Aleksandr, KAD Systems (©) 2015

* @date 23-11-2015

*/