

Customization of search and advance search fields.

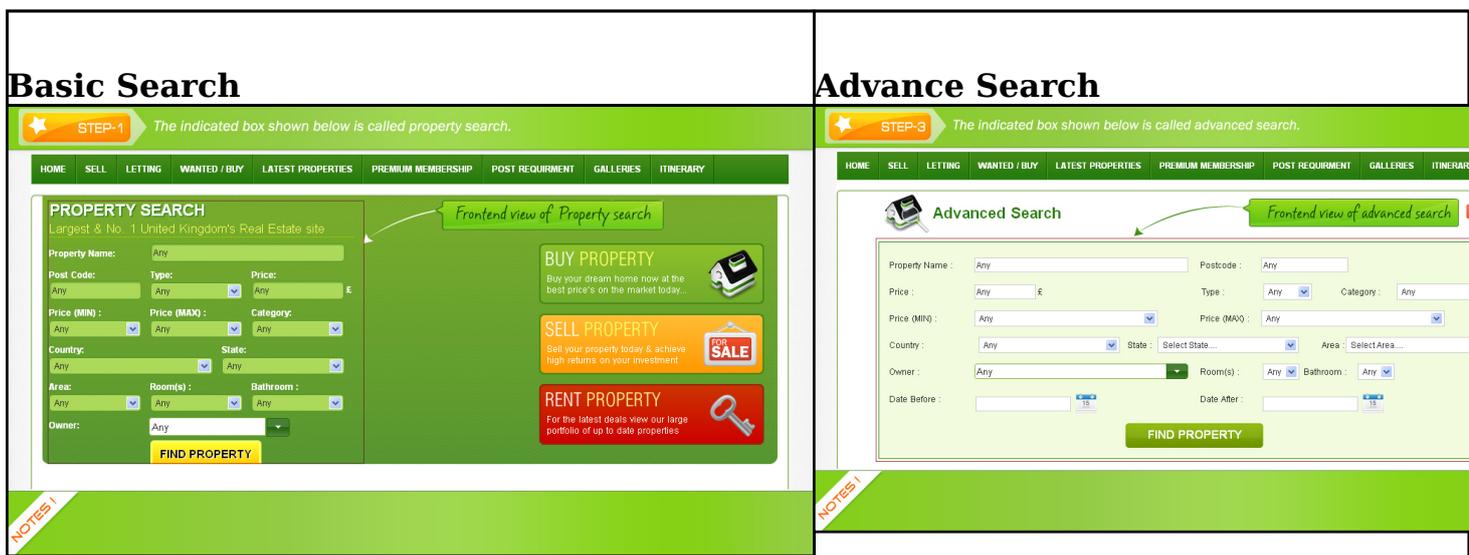
This tutorial will help you understanding, how to customize (add, edit and remove) search fields according to you requirements. We have used property modules as example but this process can be applied to any other modules such as Flight, hotel, vacation rentals, tours or autos etc. There are two step of customizing the search form. There are:

Step 01 - Customizing the fields of a search form.

Step 02 - Mapping search fields with application's module.

Step 01 - Customizing the Search Form:

By default there are two types of search forms. One is general / basic search form and other is advance search form with more fields followed by extended search criteria. Please find the sample image shown below:

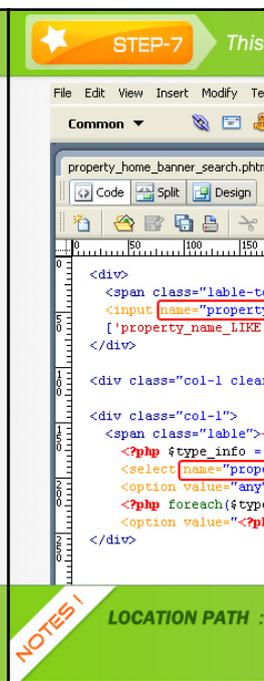
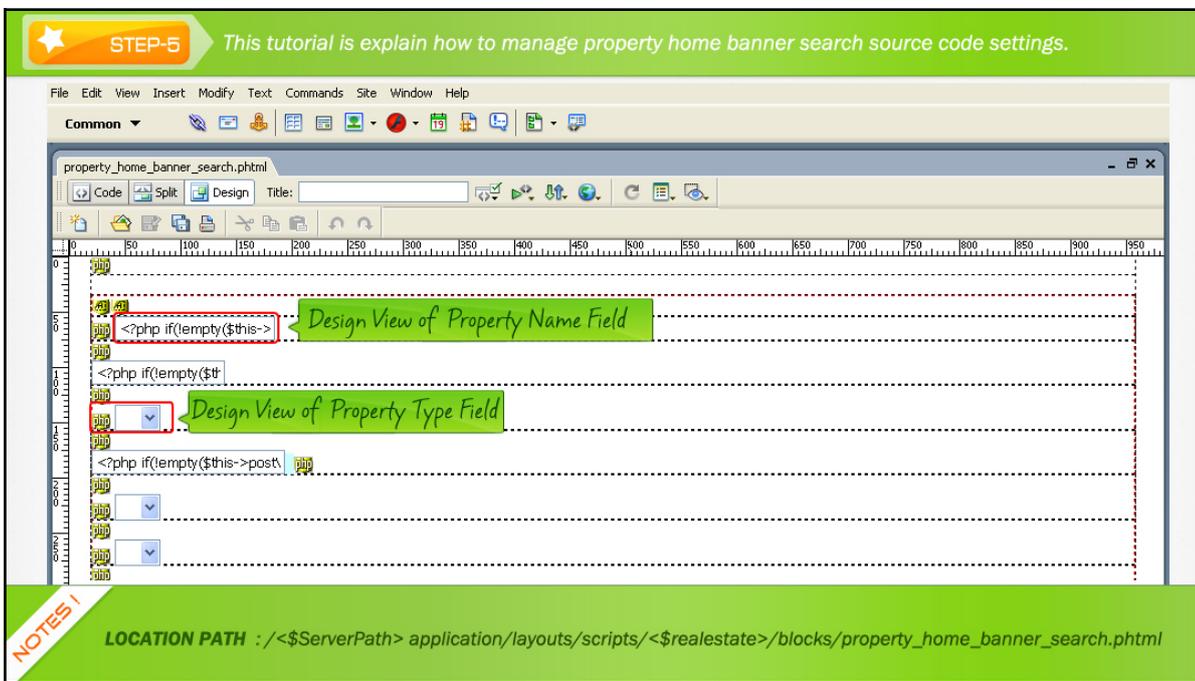


** Click on the above images for greater representation. **

As you know, all of our designs are build by combination of various blocks and all blocks files are located at /application/layouts/scripts/\$TemplatePath/blocks/. Therefore the search form is also a phtml based block file residing under block directory. For example our property search blocks are located at

/application/layouts/scripts/\$TemplatePath/blocks/property_home_banner_search.phtml by default. You can open the .phtml file with any of your preferred editor (e.g Dreamweaver , FrontPage etc) and select a field to modify. The following image illustrates in greater view.

Editing with html editor	Go to Source f



** Click on the above images for greater representation. **

As shown in source code view, all field's name are followed by SQL suffix. This is one of the MOST IMPORTANT aspects that indicates what type of sql query to run in order fetch desired data. We have summarized the SQL suffixes for your clear understanding as follows:

•

Text Search Query:

- The Sign “_LIKE” must be appended as suffix (The ending part of a field name) to run LIKE SQL query. This keyword (_LIKE) is handy when you search a text date which matched with input field's data.

•

Both Number and Text Query:

- The Sign “_=” must be appended as suffix (The ending part of a field name) to run equals SQL query. This keyword (_=) is handy when you want to fetch text or number which is equals to given input field's data.

•

Number Query:

- The Sign “_<” must be appended as suffix (The ending part of a field name) to run smaller then SQL query. This keyword (_<) is handy when you want to fetch all numbers which is lower / smaller than given number.
- The Sign “_>” must be appended as suffix (The ending part of a field name) to run

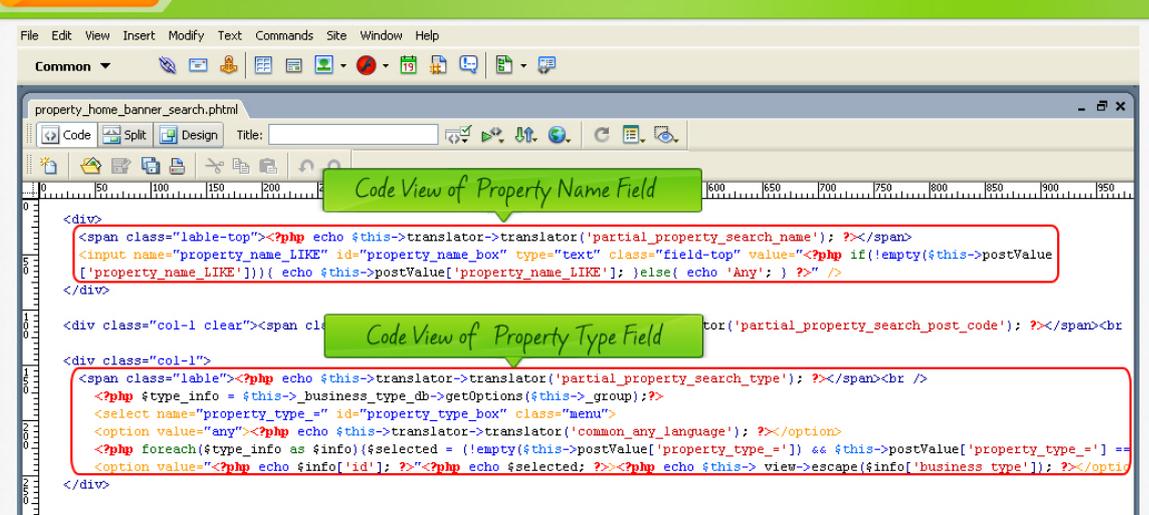
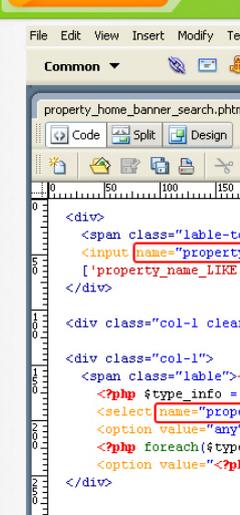
grater than SQL query. This keyword (>) is handy when you want to fetch all numbers which is higher / grater than given number.

- The Sign “_<=” means Maximum number value. The Sign “_<=” must be appended as suffix (The ending part of a field name) to run smaller then or equals SQL query. This keyword (<=) is handy when you want to fetch all numbers which is lower or equals than given number.
- The Sign “_>=” means minimum number value. The Sign “_>=” must be appended as suffix (The ending part of a field name) to run grater then or equals SQL query. This keyword (>=) is handy when you want to fetch all numbers which is higher or equals than given number.

Date Query:

- The Sign “_<” is to represent before given data. The Sign “_<” must be appended as suffix (The ending part of a field name) to run before date SQL query. This keyword (<) is handy when you want to fetch all result which is before than given date.
- The Sign “_>” is to represent after given date. The Sign “_>=” must be appended as suffix (The ending part of a field name) to run grater then or equals SQL query. This keyword (>=) is handy when you want to fetch all result which is after than given date.

For more clear visualization, please click on the below images for better Illustration.

<h2 style="text-align: left; margin: 0;">Some Samples of SQL Suffix</h2>	
<div style="background-color: #4CAF50; color: white; padding: 5px; display: flex; align-items: center;"> ★ <div style="flex-grow: 1;"> <p style="margin: 0;">STEP-6 This tutorial is explain how to manage property home banner search source code settings.</p> </div> </div> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 5px;">  <p style="text-align: center; color: green; font-weight: bold;">Code View of Property Name Field</p> <pre style="font-family: monospace; font-size: 0.9em;"> <div> <?php echo \$this->translator->translator('partial_property_search_name'); ?> <input name="property_name_LIKE" id="property_name_box" type="text" class="field-top" value="<?php if(!empty(\$this->postValue ['property_name_LIKE']))(echo \$this->postValue['property_name_LIKE'];)else(echo 'Any';) ?>" /> </div> <div class="col-1 clear"><?php echo \$this->translator->translator('partial_property_search_post_code'); ?>
 <div class="col-1"> <?php echo \$this->translator->translator('partial_property_search_type'); ?>
 <?php \$type_info = \$this->business_type_db->getOptions(\$this->group);?> <select name="property_type_" id="property_type_box" class="menu"> <option value="any"><?php echo \$this->translator->translator('common_any_language'); ?></option> <?php foreach(\$type_info as \$info){\$selected = (!empty(\$this->postValue['property_type_']) && \$this->postValue['property_type_'] == \$info['id']) ? 'selected="selected"' : ''; <option value="<?php echo \$info['id']; ?>"><?php echo \$selected; ?><?php echo \$this->view->escape(\$info['business type']); ?></option> </div> </pre> <p style="text-align: center; color: green; font-weight: bold;">Code View of Property Type Field</p> </div> <div style="background-color: #4CAF50; color: white; padding: 5px; display: flex; align-items: center;"> NOTES ! <div style="flex-grow: 1;"> <p style="margin: 0; font-size: 0.8em;">LOCATION PATH : </\$ServerPath> application/layouts/scripts/<\$realestate>/blocks/property_home_banner_search.phtml</p> </div> </div>	<div style="background-color: #4CAF50; color: white; padding: 5px; display: flex; align-items: center;"> ★ <div style="flex-grow: 1;"> <p style="margin: 0;">STEP-7 This tutorial is explain how to manage property home banner search source code settings.</p> </div> </div> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 5px;">  </div> <div style="background-color: #4CAF50; color: white; padding: 5px; display: flex; align-items: center;"> NOTES ! <div style="flex-grow: 1;"> <p style="margin: 0; font-size: 0.8em;">LOCATION PATH :</p> </div> </div>

STEP-11 This tutorial is explain how to manage property inner sidebar search source code settings.

```

<span class="table"><?php echo $this->translator->translator('property_list_page_search_price_max', '', 'Property'); ?>
</span><br />

<?php
$max_price_arr = $property_db->getPrice('DESC');
?>
<select name="property_price_<=>" id="property_price_<=>" class="menu">
<option value="any"><?php echo $this->translator->translator('common_any_language'); ?></option>
<?php
foreach($max_price_arr as $key=>$value)
{
if(!empty($value['property_price']))
{
$selected = (!empty($this->postValue['property_price_<=>']) && $this->postValue['property_price_<=>'] ==
$value['property_price']) ? ' selected="selected" ' : '';
?>

```

NOTES! LOCATION PATH : /<ServerPath> application/layouts/scripts/<realestate>/blocks/property_inner_sidebar_search.phtml

STEP-10 This

```

<div class="sidebar-search">
<select name="pr
<option valu
<?php
foreach(
{
$sel
post
' se
?>
<option
['propert
<?php
?>
</select>

```

NOTES! LOCATION PATH :

STEP-13 This tutorial is explain how to manage property inner sidebar search source code settings.

```

<span class="table"><?php echo $this->translator->translator('property_list_page_search_price_max', '', 'Property'); ?>
</span><br />

<?php
$max_price_arr = $property_db->getPrice('DESC');
?>
<select name="property_price_<=>"
<option value="any"><?php echo
<?php
foreach($max_price_arr as $key=
{
if(!empty($value['property_price']))
{
$selected = (!empty($this->postValue['property_price_<=>']) && $this->postValue['property_price_<=>'] ==
$value['property_price']) ? ' selected="selected" ' : '';
?>

```

The Sign `<=>` means Maximum number value. The Sign `<=>` must be appended as suffix (The ending part of a field name) to run smaller then or equals SQL query. This keyword (`<=>`) is handy when you want to fetch all numbers which is lower or equals than given number.

NOTES! LOCATION PATH : /<ServerPath> application/layouts/scripts/<realestate>/blocks/property_inner_sidebar_search.phtml

STEP-16 This

```

<td height="35">
<table width="99%"
<tr>
<td width="14%"><?php echo $this->translator->translator('Date Before :'); ?></td>
<td width="37%" >
<input type="text" name="property_date_<=>"
['property_date_<=>']; ?> </td>
<td width="10%"><?php echo $this->translator-
<td width="39%" >
<input type="text" name="property_date_>" id="property_date_after" value="<?php echo strip tags(trim($this->postValue
['property_date_>']); ?>" />
</td>
</tr>
</table>
</td>

```

NOTES! LOCATION PATH :

STEP-17 This tutorial is explain how to manage advanced search source code settings.

```

<td height="35">
<table width="99%" border="0" align="center" cellpadding="0" cellspacing="0">
<tr>
<td width="14%"><?php echo $this->translator->translator('Date Before :'); ?></td>
<td width="37%" >
<input type="text" name="property_date_<=>"
['property_date_<=>']; ?> </td>
<td width="10%"><?php echo $this->translator-
<td width="39%" >
<input type="text" name="property_date_>" id="property_date_after" value="<?php echo strip tags(trim($this->postValue
['property_date_>']); ?>" />
</td>
</tr>
</table>
</td>

```

The Sign `<=>` is to represent before given data. The Sign `<=>` must be appended as suffix (The ending part of a field name) to run before date SQL query. This keyword (`<=>`) is handy when you want to fetch all result which is before than given date.

The Sign `>` is to represent after given date. The Sign `>` must be appended as suffix (The ending part of a field name) to run grater then or equals SQL query. This keyword (`>`) is handy when you want to fetch all result which is after than given date.

NOTES! LOCATION PATH : /<ServerPath> /application/modules/<Property>/views/scripts/search/items.phtml

STEP-12 This

```

<div class="sidebar-search">
<select name="pr
<option valu
<?php
foreach(
{
$sel
post
' se
?>
<option
['propert
<?php
?>
</select>

```

NOTES! LOCATION PATH :

** Click on the above images for greater representation. **

The “fields ” are one of the most important aspects of a search form which plays key role fetching searched data. You MUST NOT define, add or remove “field(s)” erratically because field[s] has it’s own convention and maintains many references throughout the application including database. All fields of a form are predefined and field's attributes are encapsulated at application/modules/\$ModuleName/forms/ source/en_US.ProductForm.ini or \$any_other_form.ini. Therefore, DO NOT add or edit existing “field’s name” upon your choices that may crush the entire application down. You can remove any fields or add new fields to the search form as you wish. For adding new field, please follow / use the “field’s name” that are predefined at application/modules/\$ModuleName/forms/ source/ en_US.ProductForm.ini or \$any_of_your_form.ini or \$any_of_your_form.ini followed by SQL prefix explain above. You will find all attributes sets, validations, label’s name, attributes and other information encapsulated under this ini file. Click here to know how to manage a form through ini files.

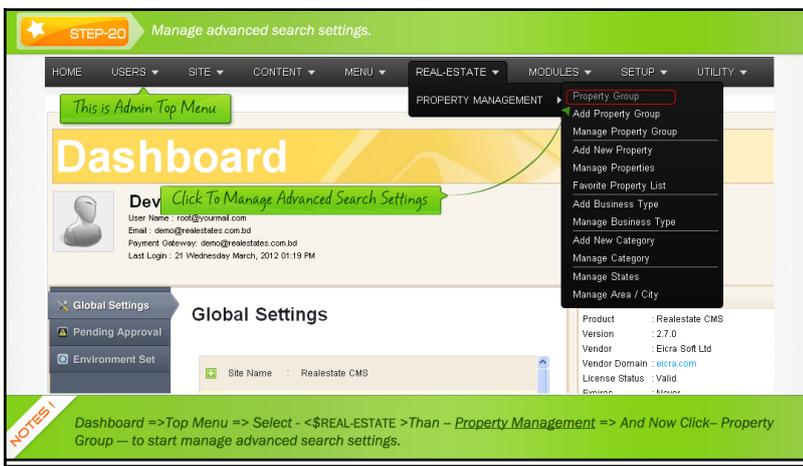
Step 02 - Mapping search fields with Application’s Module :

This is the second steps to join your customized search field with application. This step is important because following reasons:

- **Adding New Fields:** Program needs to know that, a new field is added which must be included in SQL query.
- **Editing Fields:** When a field is edited, program needs to know the medication to run correct query according to search pattern
- **Editing SQL suffix :** When a SQL query suffix has changed, program must match with field suffix to fetch correct information
- **Removing a Field:** When a field is removed, program must drop this field from its search pattern.
- **Matching Data Types:** Sql syntax changes based on Data types. If you change search Data types such as, text to date or number to text etc, the sql Data types must be mapped correctly. You have to indicate and synchronize correct data-type according to field that sets in search form.
- **Search Pattern:** When a field of a search form pattern is changed, the program must know and set the sql pattern accordingly in order to fetch desired result.
- Many more...

Since all searches belong to Modules, we must log on to corresponding module’s group setting for mapping. Let’s assume that, we are going to map search setting of Property modules (for your case module can be different), see the following image illustrates where to find corresponding module’s group search settings for mapping.

Step 01 - Selecting Group of a Module	Step 02 - Click on Search setting for mapping



** Click on the above images for greater representation. **

The following mapping interface will be displayed to configure your search fields according to the form we setup at step one. SQL uses many different data types, broken into three categories: numeric, date and time, and string types.

Numeric Data Types:

SQL uses all the standard ANSI SQL numeric data types, so if you're coming to MySQL from a different database system, these definitions will look familiar to you. The following list shows the common numeric data types and their descriptions.

- INT
- TINYINT
- SMALLINT
- MEDIUMINT
- BIGINT
- FLOAT(M,D)
- DOUBLE(M,D)
- DECIMAL(M,D)

Date and Time Types:

The SQL date and time datatypes are:

- DATE
- DATETIME
- TIMESTAMP
- TIME
- YEAR(M)

String or Text Types:

- CHAR(M)
- VARCHAR(M)
- BLOB or TEXT
- TINYBLOB or TINYTEXT
- MEDIUMBLOB or MEDIUMTEXT
- LONGBLOB or LONGTEXT
- ENUM

Find the below images, showing how can we indicate a field according to search form based on SQL data types.

STEP-22 Manage advanced text search settings.

NOTES ! The Sign **_LIKE** must be appended as suffix (The ending part of a field name) to run LIKE SQL query. This keyword (**_LIKE**) is handy when you search a text date which matched with input field s data.

STEP-26 Manage advanced text search settings.

NOTES ! The Sign **<** is to run before date SQL query.

STEP-23 Manage advanced text search settings.

NOTES ! The Sign **_=** must be appended as suffix (The ending part of a field name) to run equals SQL query. This keyword (**_=**) is handy when you want to fetch text or number which is equals to given input field s data.

STEP-24 Manage advanced text search settings.

NOTES ! The Sign **>=** means greater then or equal to given number.

STEP-25 Manage advanced neumaric search settings.

HOME USERS SITE CONTENT MENU REAL-ESTATE MODULES SETUP UTILITY

Search Settings Back Reset Save

NEUMARIC SEARCH *Click Here To Add Property Field Settings*

stars MINIMUM (≥) Value of respective field is Optional Mandatory

feature_bedroom Equals (=) **Add New**

Select Property Price Name MINIMUM (≥) Value of respective field is Optional Mandatory

property_price MAXIMUM (≤) *Select Price Maximum Value* **Remove This**

property_owner
property_code
property_type
property_primary_image
property_image
property_price
property_desc
property_date
property_order

Higher then (>)
Less then (<)
Any (1=1)
MINIMUM (≥)
MAXIMUM (≤)

Value of respective field is Optional Mandatory **Remove This**

NOTES ! The Sign `_<=` means Maximum number value. The Sign `_<=` must be appended as suffix (The ending part of a field name) to run smaller then or equals SQL query. This keyword (`_<=`) is handy when you want to fetch all numbers which is lower or equals than given number.

STEP-27 Manage advanced neumaric search settings.

HOME USERS SITE CONTENT MENU REAL-ESTATE MODULES SETUP UTILITY

Search Settings Back Reset Save

DATE & TIME SEARCH

property_build_year

property_build_year

property_build_year

Select Property

property_date
property_image
property_price
property_desc
property_date
property_order
property_est_payment
property_size

NOTES ! The Sign `_>` is to run greater then or equals SQL query. This keyword (`_>`) is handy when you want to fetch all numbers which is greater then or equals than given number.

Article ID: 106

Last updated: 09 May, 2012

Revision: 16

Modules Wise Documentation -> Manage Articles / Content (CMS) -> Customization of search and advance search fields.

<http://www.httpsdoc.com/entry/106/>