

## Simple A to Z Navigation in Rails

This quick tutorial will show you how to make a navigation bar of letters so you can display results alphabetically.

I'm going to assume you have scaffolded a controller and views for a **User** model which has the following migration:

```
class CreateUsers < ActiveRecord::Migration
  def self.up
    create_table :users do |t|
      t.column :username, :string
      t.column :last_name, :string
      t.column :first_name, :string
      t.column :middle_name, :string
      t.column :created_at, :datetime
    end

    User.create :username => 'hoganbp'
    User.create :username => 'blakeje'
    User.create :username => 'kingsa'

  end

  def self.down
    drop_table :users
  end
end
```

The first step is to build your own custom finder in your User model.

```
def self.find_all_last_names_by_letter(letter)
  letter = "#{letter}%"
  User.find :all,
    :conditions=>['users.last_name like ?', letter],
    :order => "last_name ASC"
end
```

While you're in there, add a **full\_name** method if you don't already have one.

```
def full_name
  "#{self.last_name}, #{self.first_name} #{self.middle_name}"
end
```

Be sure to write some unit tests for these!

Next, we'll build the helper method that will display the navigation links. Open up **app/helpers/application\_helper.rb** and add this method

```
def show_a_to_z_links(action = "list")
  res = Array.new
  "A".upto("Z") do |l|
    res << "#{link_to_unless_current(l, :action=>action, :filter => l)} "
  end
  return '<span id="link_list">' + res.join(" | ") + "</span>"
end
```

Our controller needs to call our new finder when the **list** action is invoked. Modify your **list** action in **/app/controllers/user\_controller.rb** so it looks like this:

```
def list
  @users = User.find_all_last_names_by_letter(params[:filter])
end
```

## Simple A to Z Navigation in Rails

end

The final piece is going to be the view for the list page. Open up **app/views/user/list.rhtml** and modify it as follows:

```
<p><%= show_a_to_z_links "list_experts" %></p>
<div id="users" >
  <table>
    <%= @users.each do |user| %>
      <tr>
        <td<%= link_to h(user.full_name), :action => 'edit', :id =>user %></td>
      </tr>
    <%= end %>
  </table>
  <%= link_to 'New User', :action => 'new' %>
</div>
```

When you call the **list** action and pass it a parameter called 'filter', it should now filter the results.

```
user/list?filter=A
user/list?filter=B
```

Of course, a route would make that look a bit nicer, wouldn't it? Add this to your **config/routes.rb** file above the default route.

```
map.user_list 'user/list/:filter', :action=>'list', :controller=>'user'
```

That should give us

```
/user/list/A
/user/list/B
```

That's it!

## About the Author

Brian Hogan is a developer, designer, trainer, and an active member of the Ruby on Rails community. He provides training, mentoring, and consulting to small teams and enjoys sharing what he learns with others.