Helpers

Implementation of rails style partials

Using partials in your views is a great way to keep them clean. Since Sinatra takes the hands off approach to framework design, you'll have to implement a partial handler yourself.

Here is a really basic version:

```
# Usage: partial :foo
helpers do
  def partial(page, options={})
    haml page, options.merge!(:layout => false)
  end
end
```

A more advanced version that would handle passing local options, and looping over a hash would look like:

```
# Render the page once:
# Usage: partial :foo
#
# foo will be rendered once for each element in the array, passing in a local variable named "foo"
# Usage: partial :foo, :collection => @my_foos

helpers do
  def partial(template, *args)
    options = args.extract_options!
    options.merge!(:layout => false)
    if collection = options.delete(:collection) then
      collection.inject([]) do |buffer, member|
        buffer << haml(template, options.merge(
                                  :layout => false,
                                  :locals => {template.to_sym => member}
                                )
                     )
      end.join("\n")
    else
      haml(template, options)
    end
  end
end
```