

GRAILS TUTORIAL

IS2150

Assoc Prof Kaushik Dutta
Department of Information Systems
School Computing
National University of Singapore

Main CSS

We will continue the tutorial using **EmployeeManager** project.

Open the **main.css** file located in the **Web Application > css** folder.

This is where most CSS is defined for your web application.

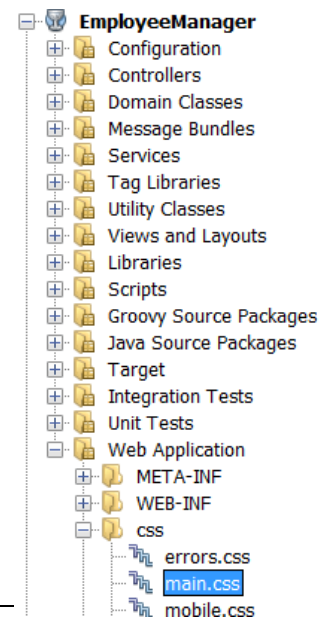
We will now change the code in **main.css** to do the following:

1. Give the website a black gradient background.
2. Give the header and footer a greyish gloss.

In the **main.css** file, change the **html** block to reflect the following:

```
html {
    /*
        background-color: #ddd;
        background-image: -moz-linear-gradient(center top, #aaa, #ddd);
        background-image: -webkit-gradient(linear, left top, left bottom, color-stop(0, #aaa),
        color-stop(1, #ddd));
        background-image: linear-gradient(top, #aaa, #ddd);
        filter: progid:DXImageTransform.Microsoft.gradient(startColorStr = '#aaaaaa',
        EndColorStr = '#dddddd');
        background-repeat: no-repeat;
        height: 100%;
    */
    background: rgb(125,126,125);
    background: -moz-linear-gradient(top, rgba(125,126,125,1) 27%, rgba(14,14,14,1) 100%);
    background: -webkit-gradient(linear, left top, left bottom, color-
    stop(27%,rgba(125,126,125,1)), color-stop(100%,rgba(14,14,14,1)));
    background: -webkit-linear-gradient(top, rgba(125,126,125,1) 27%,rgba(14,14,14,1) 100%);
    background: -o-linear-gradient(top, rgba(125,126,125,1) 27%,rgba(14,14,14,1) 100%);
    background: -ms-linear-gradient(top, rgba(125,126,125,1) 27%,rgba(14,14,14,1) 100%);
    background: linear-gradient(top, rgba(125,126,125,1) 27%,rgba(14,14,14,1) 100%);
    filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='#7d7e7d',
    endColorstr='#0e0e0e',GradientType=0 );
    height: 200%;
    background-repeat: no-repeat;

    /* change the box model to exclude the padding from the calculation of 100% height (IE8+)
    */
    -webkit-box-sizing: border-box;
    -moz-box-sizing: border-box;
    box-sizing: border-box;
}
```



Run the project. You should see that a black gradient for the background.

Next, change the **#grailsLogo** block to reflect the following:

```
#grailsLogo {
  /*background-color: #abbf78;*/
  background: rgb(226,226,226);
  background: -moz-linear-gradient(top, rgba(226,226,226,1) 0%, rgba(219,219,219,1) 50%,
  rgba(209,209,209,1) 51%, rgba(254,254,254,1) 100%);
  background: -webkit-gradient(linear, left top, left bottom, color-
  stop(0%,rgba(226,226,226,1)), color-stop(50%,rgba(219,219,219,1)), color-
  stop(51%,rgba(209,209,209,1)), color-stop(100%,rgba(254,254,254,1)));
  background: -webkit-linear-gradient(top, rgba(226,226,226,1) 0%,rgba(219,219,219,1)
  50%,rgba(209,209,209,1) 51%,rgba(254,254,254,1) 100%);
  background: -o-linear-gradient(top, rgba(226,226,226,1) 0%,rgba(219,219,219,1)
  50%,rgba(209,209,209,1) 51%,rgba(254,254,254,1) 100%);
  background: -ms-linear-gradient(top, rgba(226,226,226,1) 0%,rgba(219,219,219,1)
  50%,rgba(209,209,209,1) 51%,rgba(254,254,254,1) 100%);
  background: linear-gradient(top, rgba(226,226,226,1) 0%,rgba(219,219,219,1)
  50%,rgba(209,209,209,1) 51%,rgba(254,254,254,1) 100%);
  filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='#e2e2e2',
  endColorstr='#fefefe',GradientType=0 );
}
```

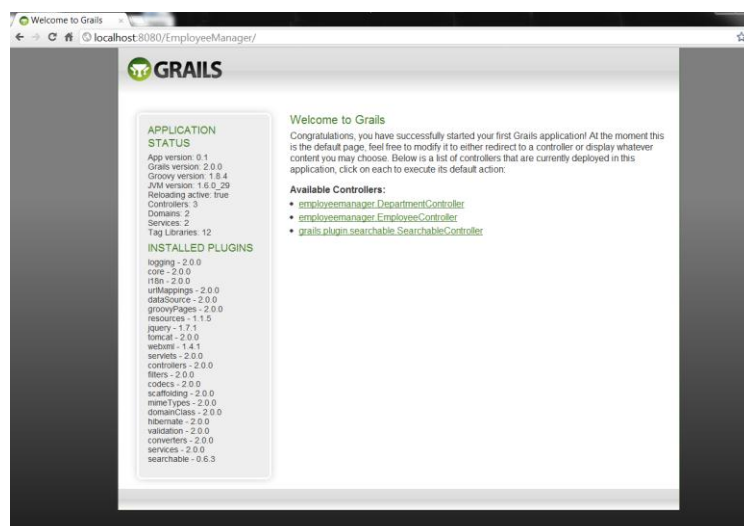
And the **.footer** block to the following:

```
.footer {
  /*background: #abbf78;
  color: #000;*/

  background: rgb(226,226,226);
  background: -moz-linear-gradient(top, rgba(226,226,226,1) 0%, rgba(219,219,219,1) 50%,
  rgba(209,209,209,1) 51%, rgba(254,254,254,1) 100%);
  background: -webkit-gradient(linear, left top, left bottom, color-
  stop(0%,rgba(226,226,226,1)), color-stop(50%,rgba(219,219,219,1)), color-
  stop(51%,rgba(209,209,209,1)), color-stop(100%,rgba(254,254,254,1)));
  background: -webkit-linear-gradient(top, rgba(226,226,226,1) 0%,rgba(219,219,219,1)
  50%,rgba(209,209,209,1) 51%,rgba(254,254,254,1) 100%);
  background: -o-linear-gradient(top, rgba(226,226,226,1) 0%,rgba(219,219,219,1)
  50%,rgba(209,209,209,1) 51%,rgba(254,254,254,1) 100%);
  background: -ms-linear-gradient(top, rgba(226,226,226,1) 0%,rgba(219,219,219,1)
  50%,rgba(209,209,209,1) 51%,rgba(254,254,254,1) 100%);
  background: linear-gradient(top, rgba(226,226,226,1) 0%,rgba(219,219,219,1)
  50%,rgba(209,209,209,1) 51%,rgba(254,254,254,1) 100%);
  filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='#e2e2e2',
  endColorstr='#fefefe',GradientType=0 );

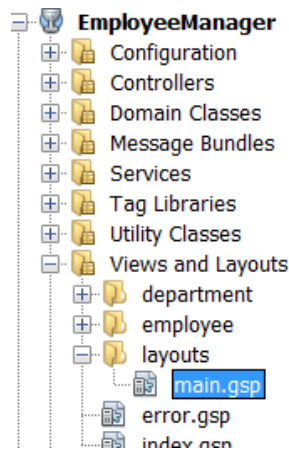
  clear: both;
  font-size: 0.8em;
  margin-top: 1.5em;
  padding: 1em;
  min-height: 1em;
}
```

Return to the browser and hit **F5**. The site should look like the following:



Main Layout

Open the **main.gsp** file located in the **Views and Layouts > layouts** folder.



This file contains the layout for every page using it.

Note that the header and footer are defined in this very file.

The **<g:layoutBody/>** tag defines the content of the pages using this layout.

```
<body>
  <div id="grailsLogo" role="banner"><a href="http://grails.org"></a></div>

  <g:layoutBody/>

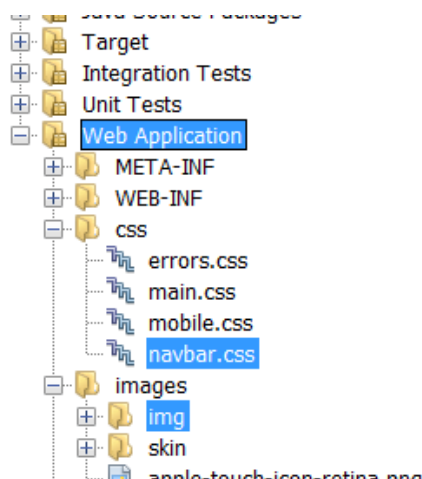
  <div class="footer" role="contentinfo"></div>
  <div id="spinner" class="spinner" style="display:none;"><g:message code="spinner.alt"
default="Loading&hellip;" /></div>
  <g:javascript library="application"/>
  <r:layoutResources />
</body>
```

We will define a simple jQuery powered navigation menu in **main.gsp**. Since it is a layout file, the navigation menu will appear across all pages.

First, paste the **navbar.css** into the **Web Application > css** folder.

Next, paste the **img** folder into the **Web Application > images** folder.

These are the resources needed for our navigation menu.



Next, in main.gsp, change the code between the <body></body> tags to:

```
<body>
  <div id="grailsLogo" role="banner"><a href="http://grails.org"></a></div>

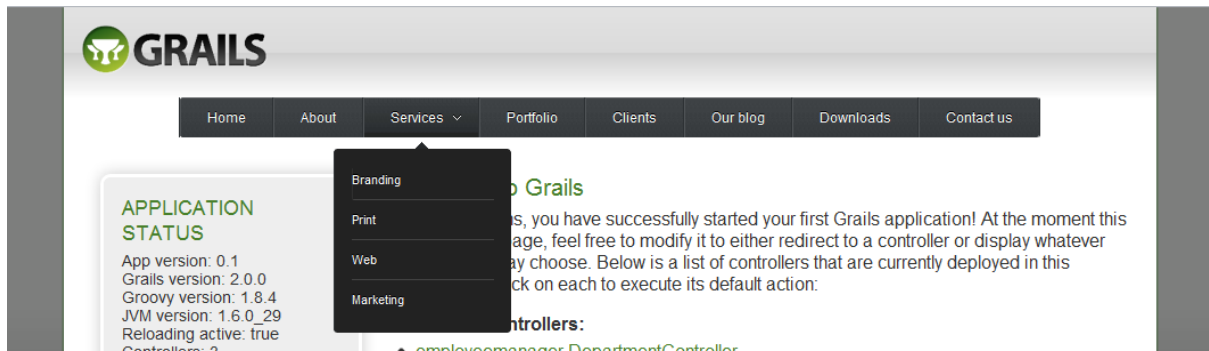
  <div class="wrapper">
    <nav class="dark">
      <ul class="clear">
        <li><a href="#">Home</a></li>
        <li><a href="#">About</a></li>
        <li><a href="#">Services</a>
          <ul>
            <li><a href="#">Branding</a></li>
            <li><a href="#">Print</a></li>
            <li><a href="#">Web</a></li>
            <li><a href="#">Marketing</a></li>
          </ul>
        </li>
        <li><a href="#">Portfolio</a></li>
        <li><a href="#">Clients</a></li>
        <li><a href="#">Our blog</a></li>
        <li><a href="#">Downloads</a></li>
        <li><a href="#">Contact us</a></li>
      </ul>
    </nav>
  </div>

  <g:layoutBody/>
  <div class="footer" role="contentinfo"></div>
  <div id="spinner" class="spinner" style="display:none;"><g:message code="spinner.alt"
default="Loading&hellip;"/></div>
  <g:javascript library="application"/>
  <r:layoutResources />
</body>
```

After that, change the code between the <head></head> tags to:

```
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
  <title><g:layoutTitle default="Grails"/></title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="shortcut icon" href="${resource(dir: 'images', file: 'favicon.ico')}}"
type="image/x-icon">
  <link rel="apple-touch-icon" href="${resource(dir: 'images', file: 'apple-touch-
icon.png')}}">
  <link rel="apple-touch-icon" sizes="114x114" href="${resource(dir: 'images', file: 'apple-
touch-icon-retina.png')}}">
  <link rel="stylesheet" href="${resource(dir: 'css', file: 'main.css')}}" type="text/css">
  <link rel="stylesheet" href="${resource(dir: 'css', file: 'mobile.css')}}" type="text/css">
  <link rel="stylesheet" href="${resource(dir: 'css', file: 'navbar.css')}}" type="text/css">
  <g:layoutHead/>
  <script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/1/jquery.min.js"></script>
  <script type="text/javascript">
    $(document).ready(function() {
      // Required: Navigation bar drop-down
      $("nav ul li").hover(function() {
        $(this).addClass("active");
        $(this).find("ul").show().animate({opacity: 1}, 400);
      },function() {
        $(this).find("ul").hide().animate({opacity: 0}, 200);
        $(this).removeClass("active");
      });
      // Required: Additional styling elements
      $('nav ul li li:first-child').prepend('<li class="arrow"></li>');
      $('nav ul li:first-child').addClass('first');
      $('nav ul li:last-child').addClass('last');
      $('nav ul li ul').parent().append('<span
class="dropdown"></span>').addClass('drop');
    });
  </script>
  <r:layoutResources />
</head>
```

Return to the browser and hit **F5**. We will now have a navigation bar across every page.



MODIFYING OTHER GSP FILES

Changing the look and feel of the application – the front page of your application can be customized by changing the index.gsp file under layouts folder. Change the content of your index.gsp to the below and see how the front page changes.

```
<!doctype html>
<html>
<head>
<title>Employee Manager Application</title>
<meta name="layout" content="main" />
</head>
<body>
  <center>
    <p> This is a simple application to manage Employees!! </p>
    <br>
    <br>
    <div class="homeCell" >
      <h3>Employee</h3>
      <br>
      <span class="buttons" >
        <g:link controller="employee" action="list" >Employee List</g:link>
      </span>
    </div>
    <br>
    <div class="homeCell" >
      <h3>Department</h3>
      <br>
      <span class="buttons" >
        <g:link controller="department" action="list" >Department List</g:link>
      </span>
      <br>
      <br>
      <div class="homeCell" >
        <h3>Search in the application</h3>
        <br>
        <span class="buttons" >
          <g:link controller="searchable" action="index" >Search</g:link>
        </span>
      </div>
    </center>
  </body>
</html>
```

CUSTOMIZING SEARCH RESULTS:

We installed the searchable plugin a couple of weeks back. The page which shows search results can also be customized to look the way we want to.

Location of searchable controller and index pages –

C:\users\USERNAME\grails\VERSION\projects\EmployeeManager\plugins\searchable

Modify the portion in index.gsp between the `<g:if test="{haveResults}">` and `</g:if>` tags to the below given code.

```
<table class="tt" cellpadding="0" align="center">
  <thead>
    <tr>
      <g:sortableColumn property="title" title="Link" style="width: 150px" />
      <g:sortableColumn property="title" title="Name" style="width: 100px" />
    </tr>
  </thead>
  <tbody>
    <div class="results">
      <g:each var="result" in="{searchResult.results}" status="index">
        <tr>
          <div class="result">
            <g:set var="className" value="{ClassUtils.getShortName(result.getClass())}" />
            <g:set var="link" value="{createLink(controller: className[0].toLowerCase() + className[1..-1],
action: 'show', id: result.id)}" />
            <pre><td><div class="name"><a href="{link}">{className} #{result.id}
</a></div></td></pre>
            <g:set var="desc" value="{result.toString()}" />
            <g:if test="{desc.size() > 120}"><g:set var="desc" value="{desc[0..120] + '...'}" /></g:if>
            <td><div class="desc">{desc.encodeAsHTML()}</div></td>
          </div>
        </tr>
      </g:each>
    </div>
  </tbody>
</table>
```

This displays the results in a table format. This is just a simple change. The search results can be customized in any way based on our needs.

TEMPLATE ADDITION – render tag

Let us assume that want to add an additional footer for the above two classes. (Employee and Department)

We can do it by Templates. Like most modern view technologies, Grails uses a layout and template-based approach to assemble the view/UI. A template is a view fragment that resides in the `grails-app/views` directory and starts with an underscore. The underscore is Grails' convention for signifying that GSP is a template. Best practices dictate that you should put templates that are associated with a specific domain, such as Employee, in the domain's view directory, like **`grails-app/views/employee/_someTemplate.gsp`**. You should put templates that are more generic or shared across views in a common place, such as `grails-app/views/common`.

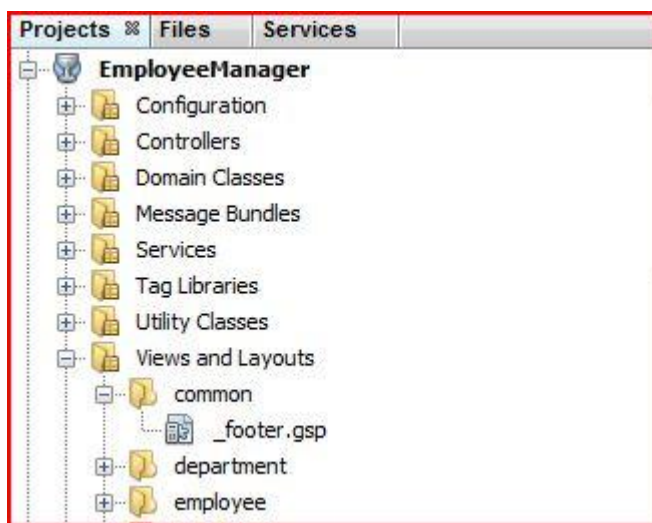
Creating the Footer

The goal is for a simple copyright notice to be displayed at the bottom of every page in the web site.

1. You need to create a GSP fragment named `_footer.gsp` in the `employeemanager` `/views/common` directory.
2. Then right click on the “Views and Layouts” to create the “`_footer.gsp`” with the following code in it.

The Footer Template (`_footer.gsp`)

```
<span class="copyright">&copy;
This is my first Grails application
<br />
</span>
```



3. You need to add the copyright to the main layout (`main.gsp`) so that it's included on every page. This is where the `<g:render>` tag comes to your aid. You use the `<g:render>` tag, which has a `template` attribute, to insert templates into GSPs. All you have to do is add `<g:render template="/common/footer" />` to the `main.gsp` like this

```
<g:layoutBody/>
<div class="footer" role="contentinfo">
    <g:render template="/common/footer"/>
</div>
<div id="spinner" class="spinner" style="display:none;"><g:message
code="spinner.alt" default="Loading&hellip;" /></div>
<g:javascript library="application"/>
<r:layoutResources />
```

4. The styling for this footer was already specified in the previous section. (this can also be customized to suit your needs)
5. Run the Project and see you will get the following footer display in all the pages.