Grails Taglibs

What is Taglib?

- A class which defines tags which aid in generating views.

- Grails supports built-in tags and custom tags.

- Grails tag library mechanism is simply, elegant and completely reload-able at runtime

GSP Tags

Grails provides more than 50 in-built useful tags.

All tags start with `<g: ....>` notation signifying the name-space 'g'.

See more tags: http://grails.org/doc/latest/

Type of Tags

Tags can be grouped under :

- Simple tags
- Logical tags
- Iterative tags
- Form Tags
- Custom tags

Simple tags

```
<g:link>
<g:createLink >
<g:resource>
```

Understand the differences ???

Some Logical tags

```
<g:if>

<g:else>

<g:elseif>

<g:unless>
```

Logical tags contd...

Example 1:

```
<g:if env="production" test="$
    {isAdmin}">

     Welcome, Admin

</g:if>
```

Example 2:

```
<g:if test="${name=='fred'}">
     Hello Fred!
</g:if>
<g:elseif test="${name== 'bob'}">
     Hello bob!
</g:elseif>
<g:elseif env="development">
     Hello unknown developer!
</g:elseif>
```

Logical Tags cont...

```
<g:unless test="${name == 'fred'}">
     Hello ${name}!
</g:unless>
```

The tag will always do the opposite of what the if tag would have done. Rarely used.

GSP iteration tags

```
<g:each>
 <g:while>
```

Ex:

```
<ul>
 <g:each var="i" in="${ (0..<100) }">
    <li>Item ${i}</li>
  </g:each>
</ul>
```

Iteration tags contd…

```
<g:while>
```

Ex:

```
<g:while test="${i < 5}">
        <%i++%>
        <p>Current i = ${i}</p>
  </g:while>
```

GSP Form tags

```
<g:form>

<g:textField>

<g:passwordField>

<g:textArea>

<g:countrySelect>

<g:select>

<g:hiddenField>
```

GSP Form tags cont..

```
<g:radioGroup>

<g:radio>

<g:checkBox>

<g:submitButton>
```

GSP Form tags cont..

- Handling multiple Submit buttons
- `<g:form controller='post'>`
  - `<g:actionSubmit value='Save' />`
  - `<g:actionSubmit value='Save' />`
- `</g:form>`

λ Creates a submit button that maps to a specific action, which allows you to have multiple submit buttons in a single form.

λ Note that it is still the form that determines the target controller.

GSP Form tags cont..

Handling multiple Submit buttons

```
<g:actionSubmit value="Update" />
```

The 'update' action is called.

```
<g:actionSubmit value="Update" action="upsert" />
```

The 'upsert' action is called, if explicitly specified

GSP Form tags cont..

Example:

```
<g:form controller='post'>

  ...

<g:actionSubmit value='Save'/>

<g:actionSubmit value='Cancel' action='cancelPost'

</g:form>
```

Other useful GSP tags

```
<g:message>

<g:hasErrors>

<g:renderErrors>

<g:fieldError>

<g:javaScript>

<g:paginate>

<g:formatDate/>
```

Custom Tags

- Provided through classes called tag libraries

- Can also be invoked as methods
  eg:- ns.myTag(name:”TestData”)

- Helps in nesting a tag inside another

Creating Tag Libraries

- grails create-tagLib tagLibName
- eg: grails create-tagLib GtunesTaglib

- Inorder to group tags, assign a package
- class GtunesTagLib {
  - static namespace = "gt"

  }

Tag Libraries : Namespaces

- Declared like: *static namespace=”gt”*

- To avoid naming conflicts

- In the GSP, the tag can be accessed using <*gt:tagName* /> in place of <*g:tagName*/>

Examples

-
  ```
  def emoticon = { attrs, body ->
   out << body() << attrs.happy == 'true' ? " :-)" : " :-("
    }
     <g:emoticon happy="true">Hi John</g:emoticon>
  ```

-
  ```
  def formatBook = { attrs, body ->
    out << "<div id="${attrs.book.id}">"
    out << "Title : ${attrs.book.title}"
    out << "</div>"
  }
  ```

Git repo

git@github.com:RichaIntelligrape/taglibDemo.git

Questions??