Controllers

Nerve Center

Handles **requests**

Generates **responses**

Determines what to **render**

or where to **redirect**

Conventions

Suffix with filename with *Controller*

ie. FooController

Resides in `grails-app/controllers/`

Actions

Named:

Any **public method** in a controller

or

A declared **closure** in a controller

Params

Params submitted to the controller action are available via the **params** object.

Params

Data Binding

Domain objects can be instantiated by passing the **params** object into the domains constructor.

```
Book book = new Book(params)
```

You can also assign the params to the Domain objects **properties** field.

```
book.properties = params
```

Action Method Params

Action Binding

Params can be bound and cast to typed params in the **action**s method signature.

```
def addAmount(Double amt) {
    ...
    account.add(amt)
}
```

The Free Stuff

Injected at runtime

- log
- any **Services** via convention

The Free Stuff

Available objects

```
servletContext
request
response
session
params
flash
```

Rendering Views

Implicit (convention)

- Simply return a map of params
- Uses a view that is named the same as the action.
- Map values will bind to the placeholder
  - values in the gsp

Rendering Views

Explicit

- Call the **render** method. Passing the **view** and the **model.**

Rendering Text

```
def showText() {
  render "Hello, World"
}
```

```
Hello, World
```

Rendering JSON

```
def showJson() {
  def map = [foo: 42, bar: 'Hi']
  render map as JSON
}
```

```
{"foo":42,"bar":"Hi"}
```

Rendering XML

```
def showXml() {
  def map = [foo: 42, bar: 'Hi']
  render map as XML
}
```

```
<map>
  <entry key="foo">42</entry>
  <entry key="bar">Hi</entry>
</map>
```

Rendering Templates

Templates are partial html pages.

Render them similar to **views** just use pass the **template** and **model** param in the **render** method.

```
render(template:'tableRow', model:[row: book])
```

Forward

Forward to the request to a different controller/action.

```
forward(controller: Book, action: list, id:3)
```

Redirect

Sends a 302 status message back to the browser along with the new URL that it should call.

```
redirect(controller:book, action: display, id: 3)
```

Workshop

Check out the introControllersStart branch

Create a controller named VoteController in the com.opi package

```
> grails generate-views com.opi.Question
> grails generate-views com.opi.Answer
```

Workshop Continued

Create 2 actions in the VoteController

- `voteUpQuestion()`
- `voteDownQuestion()`

Both actions should accept Question ID and redirect to the QuestionController.show action