Week 3 Lecture 8

Applied

What’s in this lecture?

- HTTP methods: POST, GET
- Advanced Rails Controllers

HTTP Methods:

- **GET**
- **POST**
- **PUT**
- **DELETE**

- HEAD
- TRACE
- OPTIONS
- CONNECT
- PATCH

HTTP GET

- Required:
  - URL
  - Headers
- Example: http://www.google.com

HTTP GET

- Rule of thumb:

  A GET request is used if you need to access a resource on the server without changing it

HTTP GET

- You can pass parameters:

http://localhost:3000/listings?filter=price&year=2011

HTTP GET

- Called the ‘query_string’
- Takes the general form:

http://localhost:3000/search?field1=value1&field2=value2

HTTP GET

- Why? Huh? Parameters? Queries?

- Think of /listing and /search as procedures
- We’re passing them a hash of args
- The args define our parameter space
- ‘query_strings’ doesn’t imply search

Example 01 -- The Deli

- We have a program located at:
  http://localhost:3000/subs
- that returns all the subs we make
- We can pass it two pieces of filtering information:
  type => hot; cold
  vegetarian => true; false
- What is GET request for a hot vegetarian sub?

Example 01 -- The Deli

Answer:

http://localhost:3000/subs?type=hot&vegetarian=true

Accessing query_string

```
class SubsController < ApplicationController

  def index # the action
    type = params[:type]         # hot
    vega = params[:vegetarian]   # true
    @sub = Sub.find(:first, :conditions => {
      :type => type,
      :vegetarian => vega
     })
    render :json => @sub
  end

end
```

Helpful Hint:

```
Rails.logger.debug “#############”
Rails.logger.debug “#{params.inspect}”
Rails.logger.debug “#############”
```

HTTP POST

- Requires:
  - URL
  - Headers
  - message body

  (GET with a body)

HTTP POST

- Body:
  Equivalent to GET's query_string, but can include JSON, XML, plain text...

HTTP POST

- Rule of thumb:

  A POST request is used when an object on the server gets created

HTTP POST

Example:

URL: http://localhost:3000/subs
body: type=cold&name=italian

Connecting to a form:

GET:

```
<form action="check_pulse" method="get">
  Name:           <input type="textbox" name="name" /> <br />
  Are you alive:  <input type="checkbox" name="alive" /> <br />
  <input type="submit" name="submit" />
</form>
```

Connecting to a form:

POST:

```
<form action="check_pulse" method="post">
  Name:           <input type="textbox" name="name" /> <br />
  Are you alive:  <input type="checkbox" name="alive" /> <br />
  <input type="submit" name="submit" />
</form>
```