

- ❖ **Ruby: Classes, Instances, and Attributes**
- ❖ **Intro to Rails**
 - ❖ **RCAV**
 - ❖ **URLs**
 - ❖ **QueryStrings**
 - ❖ **Forms**

Ruby, Part 3

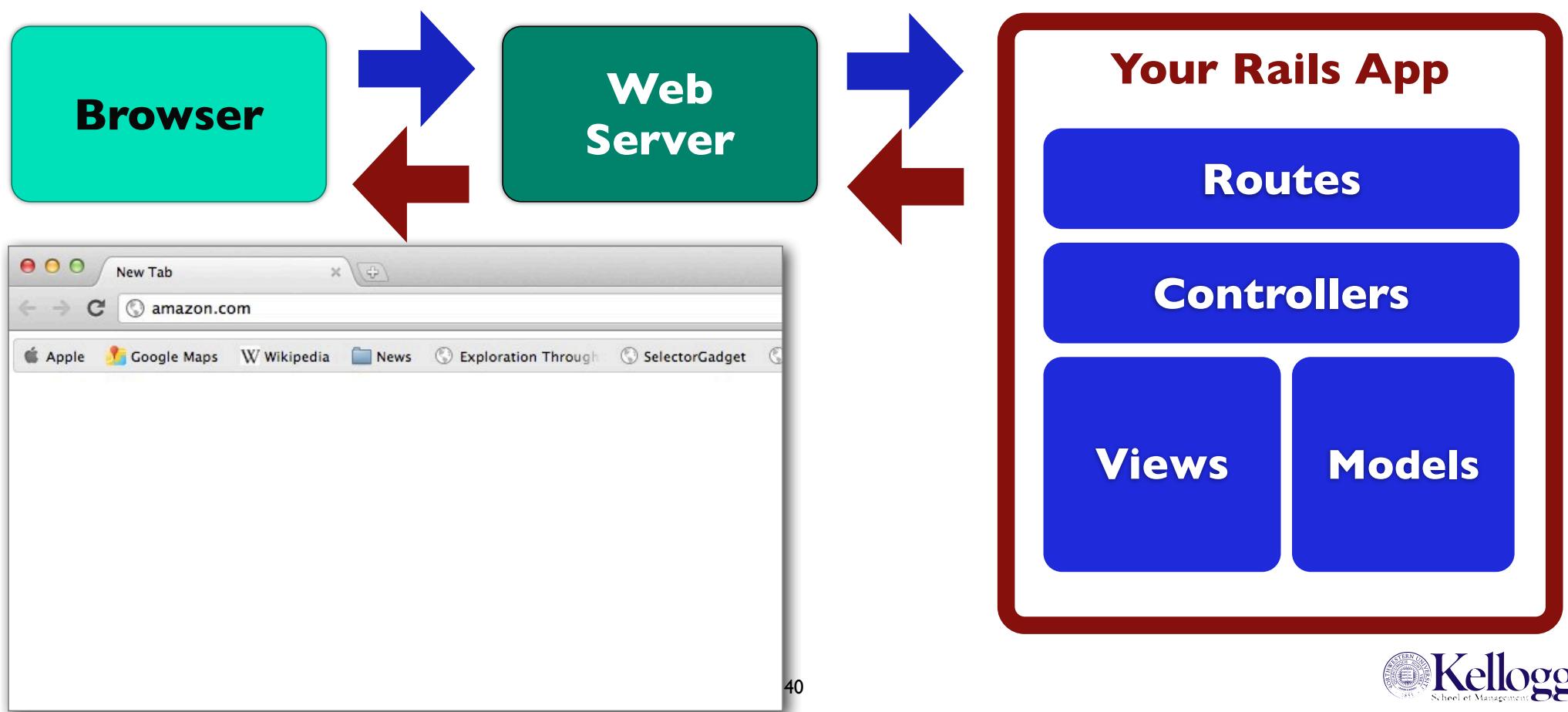
Classes

Instances

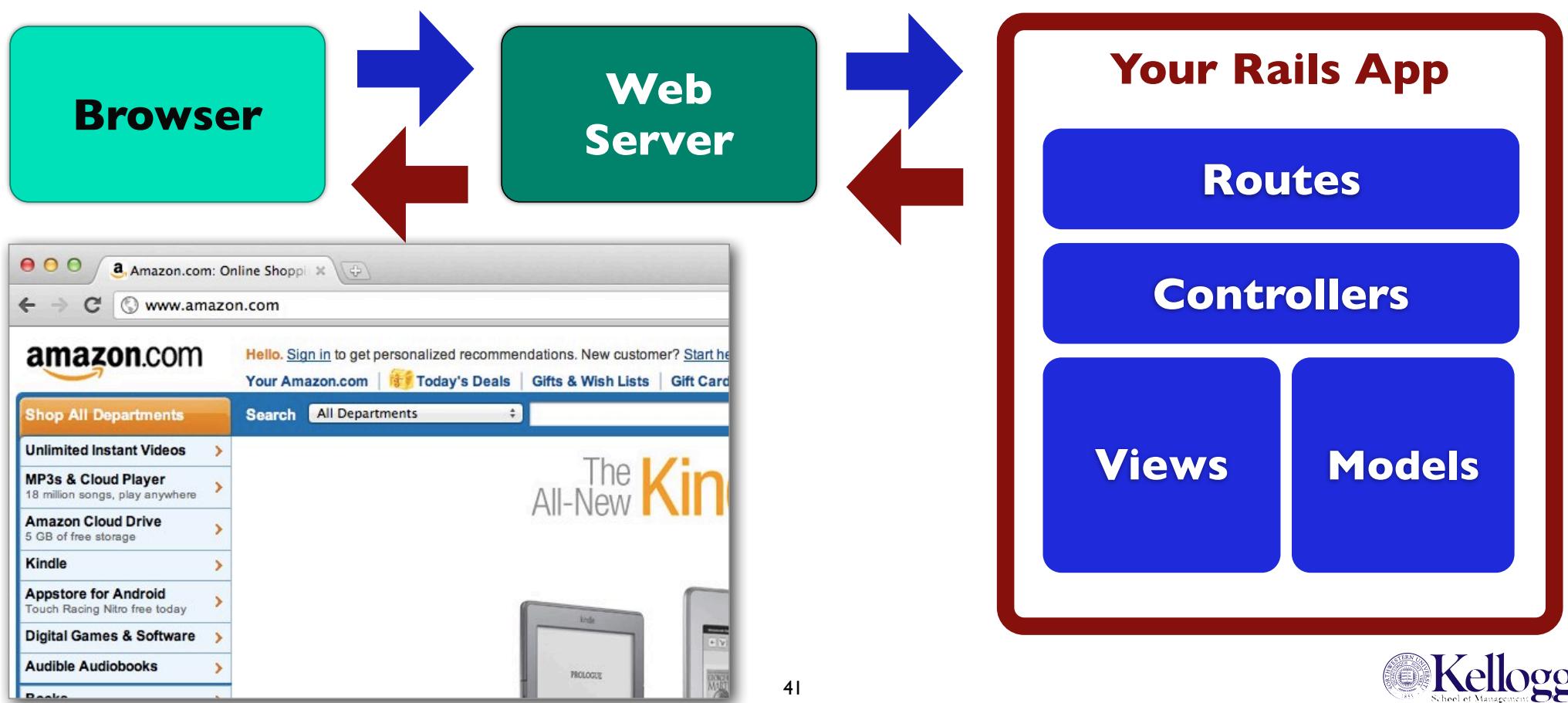
Attributes

HTTP

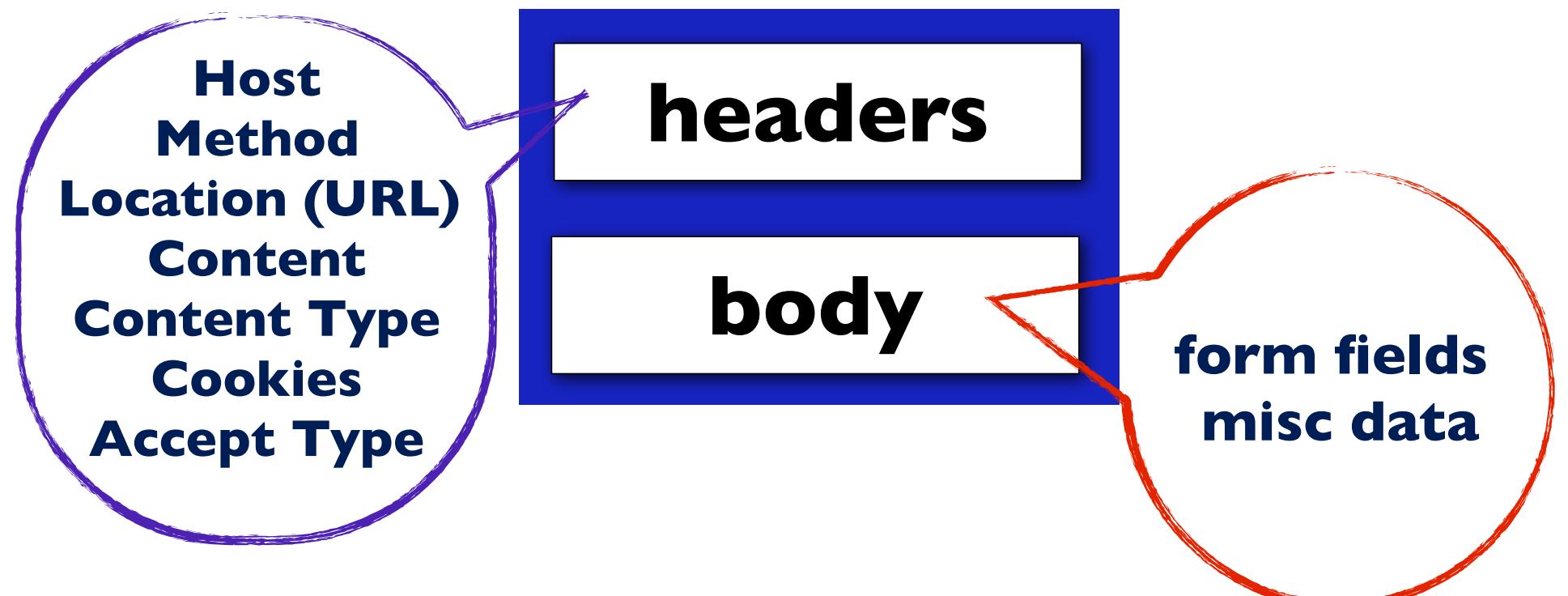
HTTP Overview



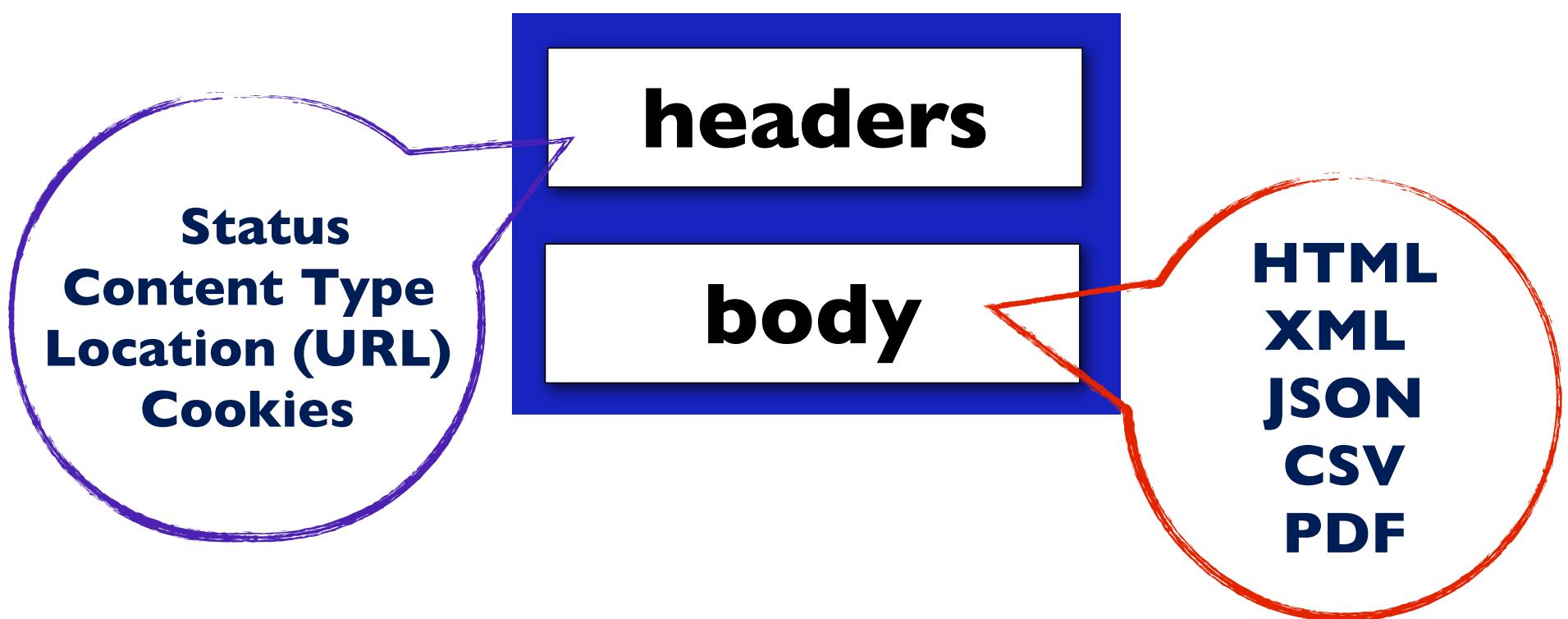
HTTP Overview



HTTP Request



HTTP Response



HTTP Playground

Playing is the best way to learn!

- ❖ curl
- ❖ Chrome Inspector
- ❖ *search for "http sniffer"*

Ruby on Rails



Why Rails?

**Database-Backed
Web Applications**

**Convention Over
Configuration**

Agile Development

Why Rails?

STTCPW

YAGNI

DRY

SRP

Why Rails?

Responsibility

Rails QuickStart

- 1. cd ~/week2**
- 2. rails new myapp**
- 3. cd myapp**
- 4. rails server**
- 5. Go to <http://localhost:3000>**

Rails: Routes



Rails 101: Routes

Basic Definition

A **route** connects
a **URL**
to a specific **web page**
in your app.

Rails 101: Routes

A Better Definition

A **route** connects
an **http request**
to a specific Ruby method
in your app.

Rails 101: Routes

Best Definition

A **route** connects
an **http request**
to a **Rails Action**
in your app.

Rails: Actions

HTTP Response Headers

HEADER	MEANING
Content-Type	Type of data
Location	Resource location
Status	Response code
Set-Cookie	Cookie data

HTTP Response Headers

An **action** generates a **response**
by using the **render** method
or an **HTML view template.**

Rendering in HTTP Response

```
def greeting
  render( :text => "Hello!", :status => 200)
end
```

Generates an HTTP response with a body and header status code.

Rendering in HTTP Response

```
def greeting
  render(:text => "Hello!",
         :status => 301,
         :location => "http://www.apple.com")
end
```

Generates an HTTP response with a body, header status code, and header location value.

Rails Views

Rendering a view template

```
def greeting  
  render 'greeting'  
end
```

Generates an HTTP response by using a *view template* named *greeting.html.erb*

Rendering conventions

```
def greeting  
  render  
end
```

The filename can be omitted if it's the same as the name of the action method.

Rendering conventions

```
def greeting  
end
```

The render statement can be omitted if you want to render a view with the same name as the action method.

Data-Driven Views

Instance variables in action methods can be used inside the view template for that action!

```
def greeting  
  @salutation = "Wazzzzup!"  
end
```

Data-Driven Views

We use embedded Ruby to insert logic and expression evaluation:

```
<h1><%= @salutation %></h1>
```

The RCAV Recipe

- 1. Define a route**
- 2. Create a controller class**
- 3. Create an action method**
- 4. Create a view template**