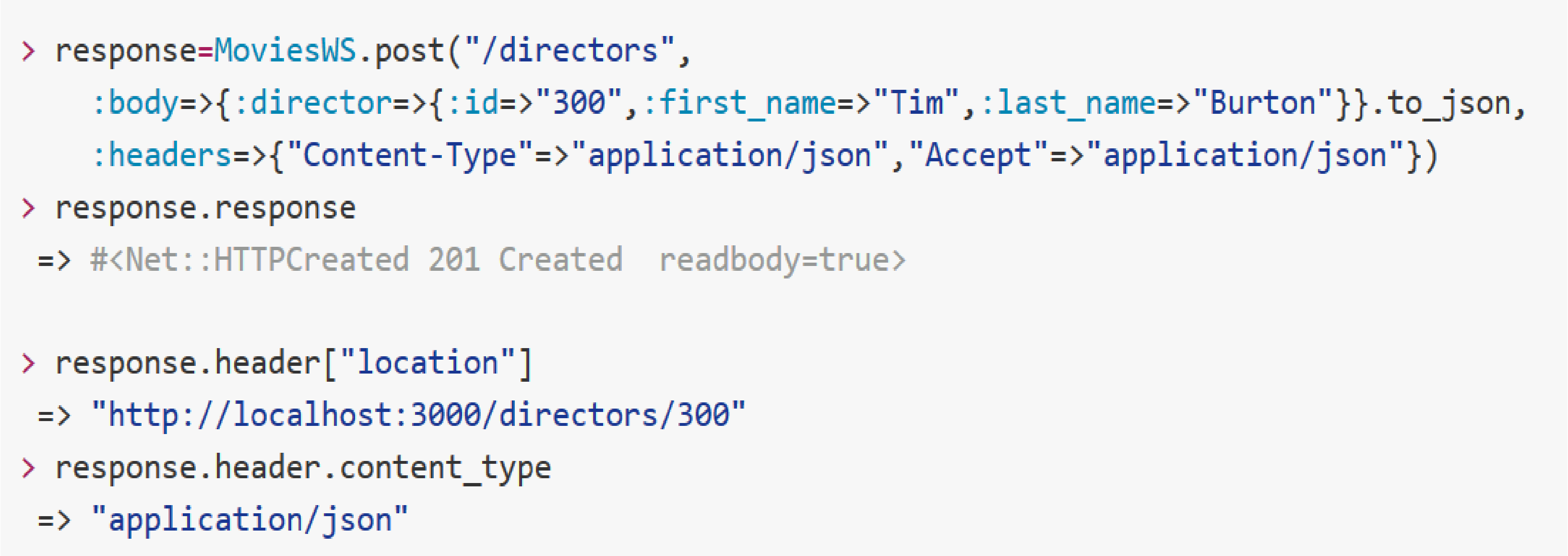
In this lecture, we will discuss…

- ✧ Content Negotiation
- ✧ Accept and Accept-Encoding Headers

Content Negotiation

- ✧ Leave content type out of the URI
- ✧ Use HTTP Headers ("Accept" and "Content-Type") to express formats and encodings
  - `Content-Type:` what we are sending in
  - `Accept:` what we are willing to accept

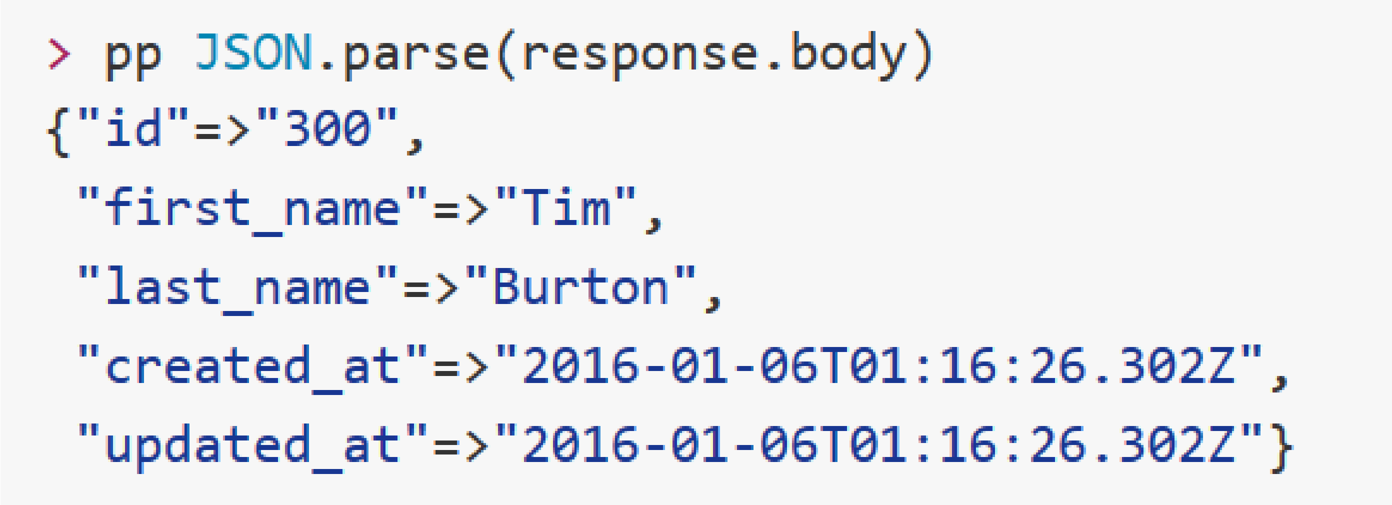
Content Negotiation - Example



```
> response=MoviesWS.post("/directors",
    :body=>{:director=>{:id=>"300",:first_name=>"Tim",:last_name=>"Burton"}}.to_json,
    :headers=>{"Content-Type"=>"application/json","Accept"=>"application/json"})
> response.response
 => #<Net::HTTPCreated 201 Created  readbody=true>

> response.header["location"]
 => "http://localhost:3000/directors/300"
> response.header.content_type
 => "application/json"
```

Content Negotiation - Example



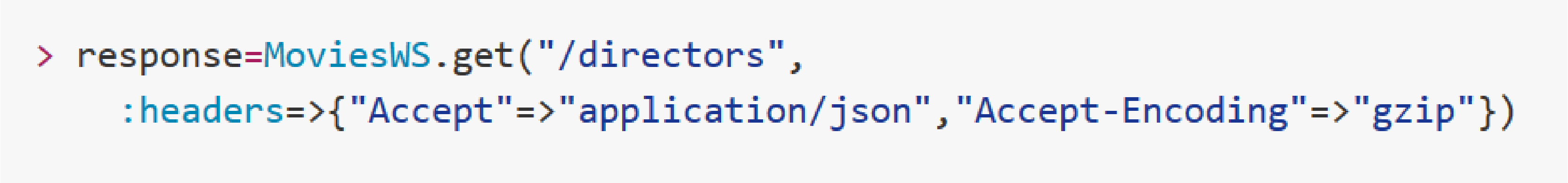
```
> pp JSON.parse(response.body)
{"id"=>"300",
 "first_name"=>"Tim",
 "last_name"=>"Burton",
 "created_at"=>"2016-01-06T01:16:26.302Z",
 "updated_at"=>"2016-01-06T01:16:26.302Z"}
```

Headers

- Accept: end-format we are willing to accept
- Accept-Encoding: intermediate form encoded on wire
- config/application.rb
  - `config.middleware.use Rack::Deflater`

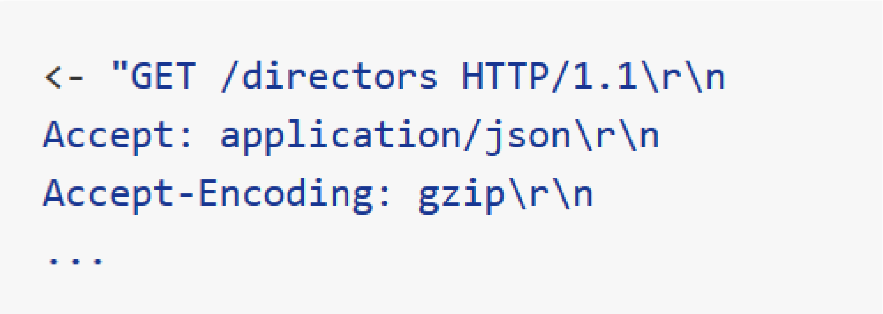
Headers - Demo

✧ API Command



```
> response=MoviesWS.get("/directors",
      :headers=>{"Accept"=>"application/json","Accept-Encoding"=>"gzip"})
```

✧ HTTP Request



```
<- "GET /directors HTTP/1.1\r\n
Accept: application/json\r\n
Accept-Encoding: gzip\r\n
...
```

Summary

- ✧ Content Negotiation – server-based content negotiation to serve up the appropriate content

What's Next?

- ✧ Headers and Status codes