



Kellogg  
School of Management

Welcome to KIEI-925

Brian Eng

Lecturer

b-eng@kellogg.northwestern.edu

@eng

Garrett Martin

Teaching Assistant

garrett.martin@kellogg.northwestern.edu



Course Website: [golearntocode.com/kiei925-winter16](http://golearntocode.com/kiei925-winter16)  
“Office Hours”: [community.golearntocode.com/c/kiei-925-winter-2016](http://community.golearntocode.com/c/kiei-925-winter-2016)  
“Real” office hours by appointment

# Grading

- Homework = 60% (6 @ 10% each)
- Midterm Exam (Week 6) = 15%
- Final Project = 25%

# Final Project

- MVP
- Your own idea – teams strongly encouraged!
- Ideation Spreadsheet

# MVP

Minimum Viable Product

# Traditional Product Management

- Assume you know it all
- Or make some stuff up
- Work really hard
- Spend all the money

*Land*



# Building Materials

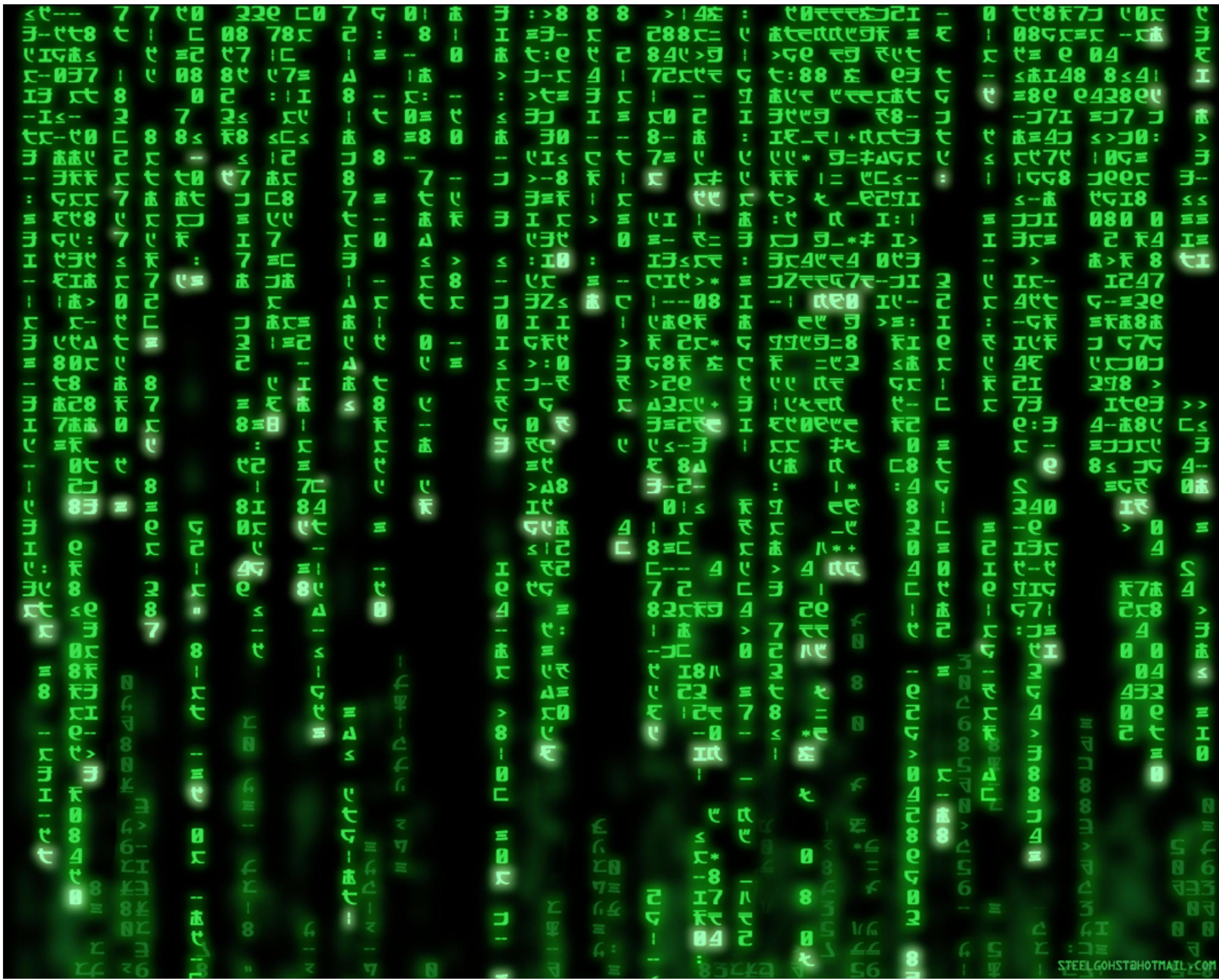


*Labor*



I think I can!





- Land = Hosting/  
Storage/Bandwidth
- Building Materials =  
Code
- Labor = Coders

**Fire, then aim**

# Lean Product Management

- Minimum feature set
- Measure & Learn
- Pivot or persevere



# Welcome to Amazon.com Books!

*One million titles,  
consistently low prices.*

(If you explore just one thing, make it our personal notification service. We think it's very cool!)

## SPOTLIGHT! -- AUGUST 16TH

These are the books we love, offered at Amazon.com low prices. The spotlight moves **EVERY** day so please come often.

## ONE MILLION TITLES

Search Amazon.com's [million title catalog](#) by author, subject, title, keyword, and more... Or take a look at the [books we recommend](#) in over 20 categories... Check out our [customer reviews](#) and the [award winners](#) from the Hugo and Nebula to the Pulitzer and Nobel... and [bestsellers](#) are 30% off the publishers list...

## EYES & EDITORS, A PERSONAL NOTIFICATION SERVICE

Like to know when that book you want comes out in paperback or when your favorite author releases a new title? Eyes, our tireless, automated search agent, will send you mail. Meanwhile, our human editors are busy previewing galleys and reading advance reviews. They can let you know when especially wonderful works are published in particular genres or subject areas. Come in, [meet Eyes](#), and have it all explained.

## YOUR ACCOUNT

Check the status of your orders or change the email address and password you have on file with us. Please note that you **do not** need an account to use the store. The first time you place an order, you will be given the opportunity to create an account.

# User Stories

As a \_\_\_\_\_, I want to \_\_\_\_\_,

so that \_\_\_\_\_.

# Domain Model



**What is this? A center for ANTS!?**



**HAWAII** DRIVER  
LICENCE

NUMBER 01-47-87441

DOB 06/03/1981 EXP 06/03/2008

HT	WT	HAIR	EYES	SEX	CTY
5-10	150	BRO	BRO	M	0

ISSUE DATE CLASS RESTR ENDORSE  
06/18/1998 3

*McLovin*

McLOVIN  
892 MOMONA ST  
HONOLULU, HI 96820



## License

id

name

address

dob

expires

height

weight

more...

# User Stories

- As a user, I want to see the details (title, summary, year released, etc) of a movie
- As a user, I want to see the cast of the movie
- As a user, I want to read and write reviews of movies

# RCAV

Route, Controller, Action, View

# THE FILES

ARE IN THE COMPUTER?!



# Ruby

*A Programmer's Best Friend*



**Class**

**Methods**

**Data**

**String**

`upcase, downcase, length, ...`

**"Hello"**

Determine the *length* of the string  
“Northwestern”

Is 420 is an *even* number?

*Reverse* the letters in the word “stressed”



**Hey girl,  
you will always be  
my number #0.**

Support the URL <http://localhost:3000/dice/roll>

Roll the dice

/weather

The current temperature in Chicago is \_\_\_\_\_.  
Current conditions are \_\_\_\_\_ with winds of  
\_\_\_\_\_ mph.

A photograph of a man with light brown hair and blue eyes, wearing a plaid shirt. He is sitting on a brown leather couch, resting his head on his right hand which is propped up. He is looking directly at the camera with a neutral expression. The background is a warm-toned wall.

Hey girl,

In case you were wondering,  
Upper, lower, or camel...it really doesn't matter.

I'll always be sensitive.

Do, or do not. There is no \_\_\_\_\_.



Do, or do not. There is no \_\_\_\_\_.



WRONG

# **CRUD**

# Create

```
movie = Movie.new  
movie.name = "Apollo 13"  
movie.year_released = "1995"  
movie.summary = "Spoiler alert: we land on the moon"  
movie.save
```

```
Movie.create(name: "Apollo 13", year_released: "1995",  
summary: "Spoiler alert: we land on the moon")
```

# Read

```
movies = Movie.all
```

```
movie = Movie.find_by(id: 1)
```

```
movie = Movie.find(1)
```

# Update

```
movie = Movie.find_by(id: 1)
```

```
movie.name = "Star Wars IV: A New Hope"
```

```
movie.save
```

```
movie = Movie.find_by(id: 1)
```

```
movie.update(name: "Star Wars IV: A New Hope")
```

# Delete

```
movie = Movie.find_by(id: 1)  
movie.delete
```

# The Dance - Buyflix

git checkout -f .

git clean -df .

git pull

bundle install

rake db:migrate

rake db:seed

# URL

<http://google.com/>

# Querystring Parameters

<http://maps.google.com/?q=Paris>

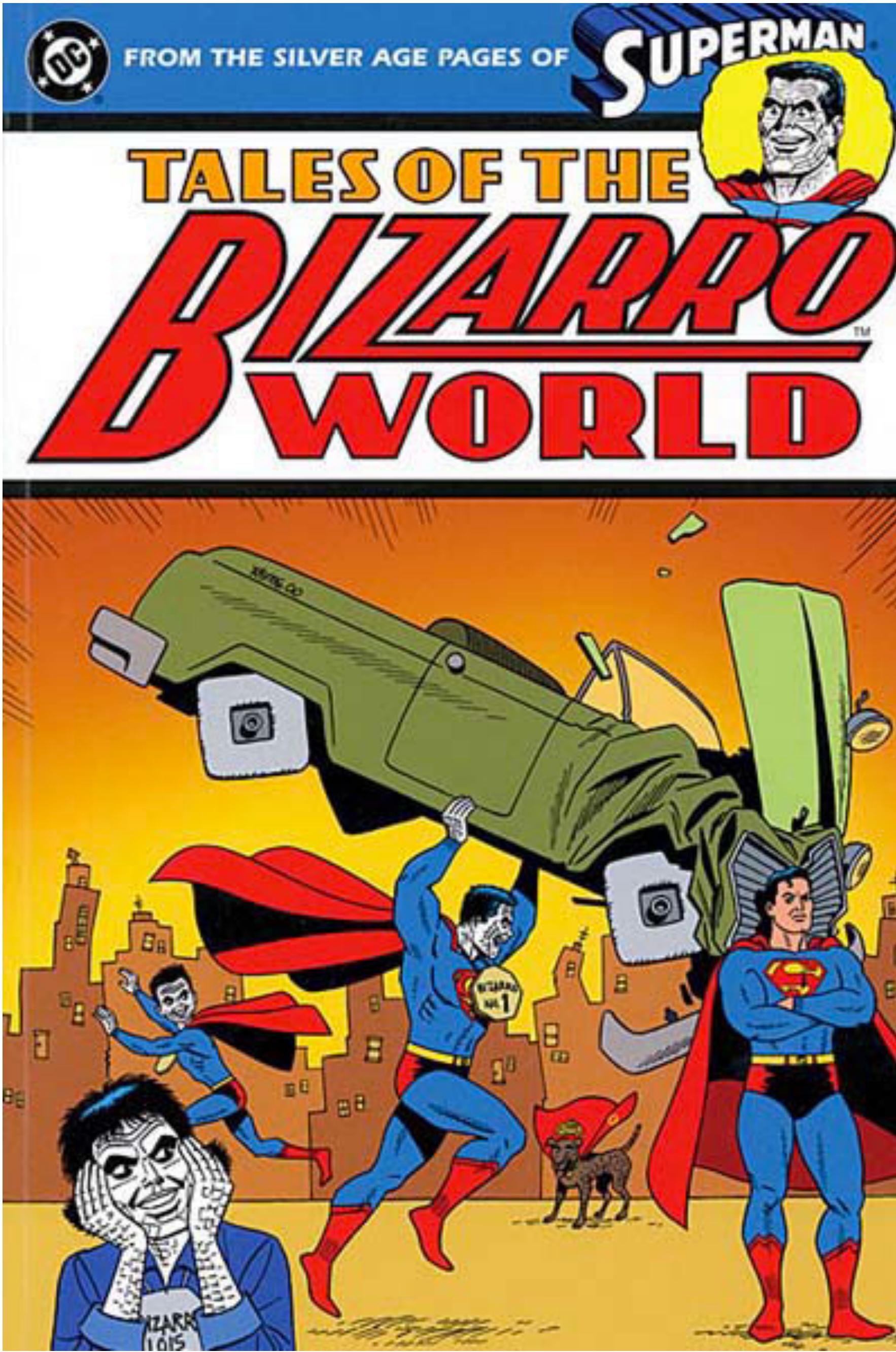
# HTTP Method

GET

POST

PATCH

DELETE



# CRUD

(Database)

HTTP

(Web App)

# CRUD

Read

Create

Update

Delete

# HTTP

GET

POST

PATCH

DELETE

# Tasks

**GET**

Feed the  
dog!

Don't forget  
to buy milk!

Take out  
the trash!

**POST**

Feed the  
dog!

Don't forget  
to buy milk!

Take out  
the trash!

Paint the  
garage!

Feed the  
dog  
cat!

Paint the  
garage!

Don't forget  
to buy milk!

Take out  
the trash!

PATCH

Feed the  
dog  
cat!

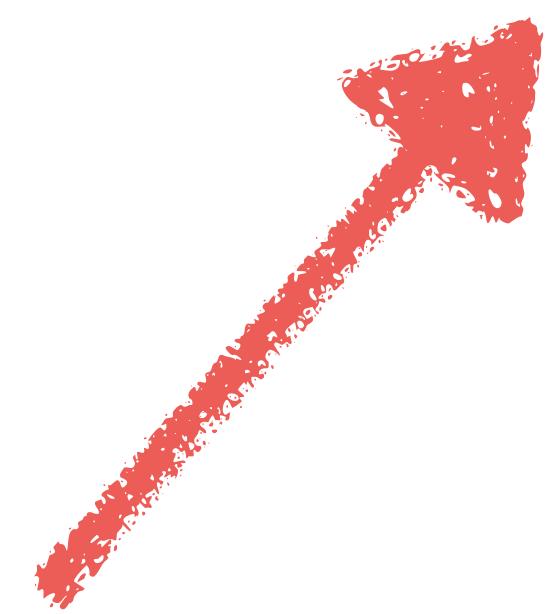
Paint the  
garage!

Don't forget  
to buy milk!

Take out  
the trash!

**DELETE**

# URL



# Tasks

**Resources = nouns**

**Methods = verbs**

# Resources (nouns)

Movies

Actors

Users

Reviews



Plural!

Movies

Actors

Users

Reviews

Directors

Producers

Theaters

Showings

# Methods (verbs)

GET

POST

PATCH

DELETE

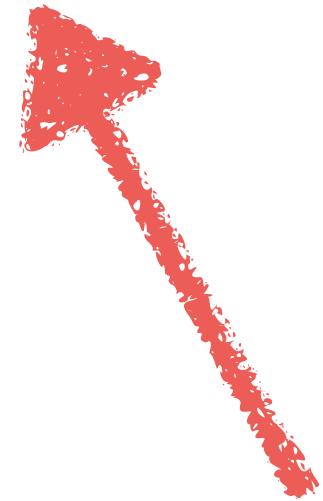
CRUD

REST

HTTP

SSL

# RCAV



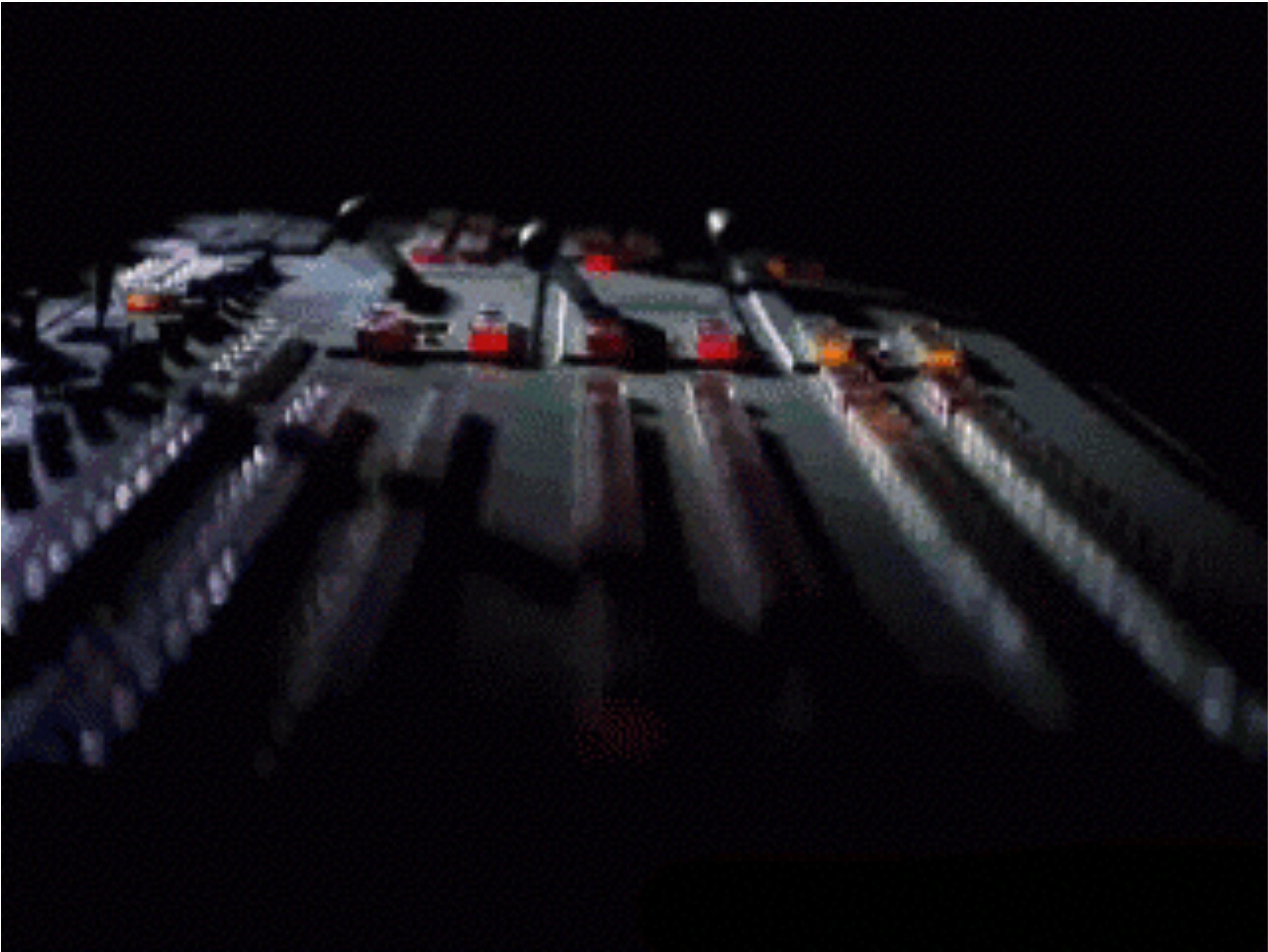
Action

# Movie Resource

Action	What It Does
index	Display a list of movies
show	Show one movie
new	Form for creating a new movie
create	Create the new movie
edit	Form for editing a movie
update	Update the movie
destroy	Delete the movie

# Movie Resource

Action	What It Does	URL	Method
index	Display a list of movies	/movies	GET
show	Show one movie	/movies/1	GET
new	Form for creating a new movie	/movies/new	GET
create	Create the new movie	/movies	POST
edit	Form for editing a movie	/movies/1/edit	GET
update	Update the movie	/movies/1	PATCH
destroy	Delete the movie	/movies/1	DELETE





# The Dance (Buyflix)

git checkout -f .

git clean -df .

git pull

bundle install

rake db:migrate

rake db:seed

# Buyflix Stories - Week 5

- As a user, I want to browse a list of movies to choose from
- As a user, I want to view the details of a single movie, so that I can see the summary, rating, and cast
- As an admin, I want to create a new actor
- As an admin, I want to edit the details of an existing actor
- As an admin, I want to delete an existing actor



# The Dance (Buyflix)

git checkout -f .

git clean -df .

git pull

bundle install

rake db:migrate

rake db:seed

# Agile Development

# Traditional (Waterfall) Project Management

Step 1: Plan plan plan plan plan plan plan

Step 2: Make some Gantt charts or  
something

Step 3: Make or miss deadline or budget

Step 4: Blame someone



# Agile Development

- Constant communication (stand-up meetings, Slack, etc)
- Accept uncertainty and failure
- Small iterations (sprints) with extremely narrow focus
- Retrospective

# Retrospective

- Gather data
- Generate insights
- Decide what to do next

# **Model Associations – DSL**

# Domain-Specific Language

**Grande, Iced, Sugar-Free, Soy Vanilla Latte**

**Tall, Half Sweet, Non-Fat, Caramel Macchiato**

**Venti Iced Skinny Hazelnut Macchiato**

Cleared to New York Kennedy, Compton 3 Foxtrot  
departure, Squawk 6244, QNH 998 Hectopascals

Turn right on Bravo, Link 21 join Alpha, hold at MORRA

Speedbird 472, LOC/DME approach, Runway 27 Left

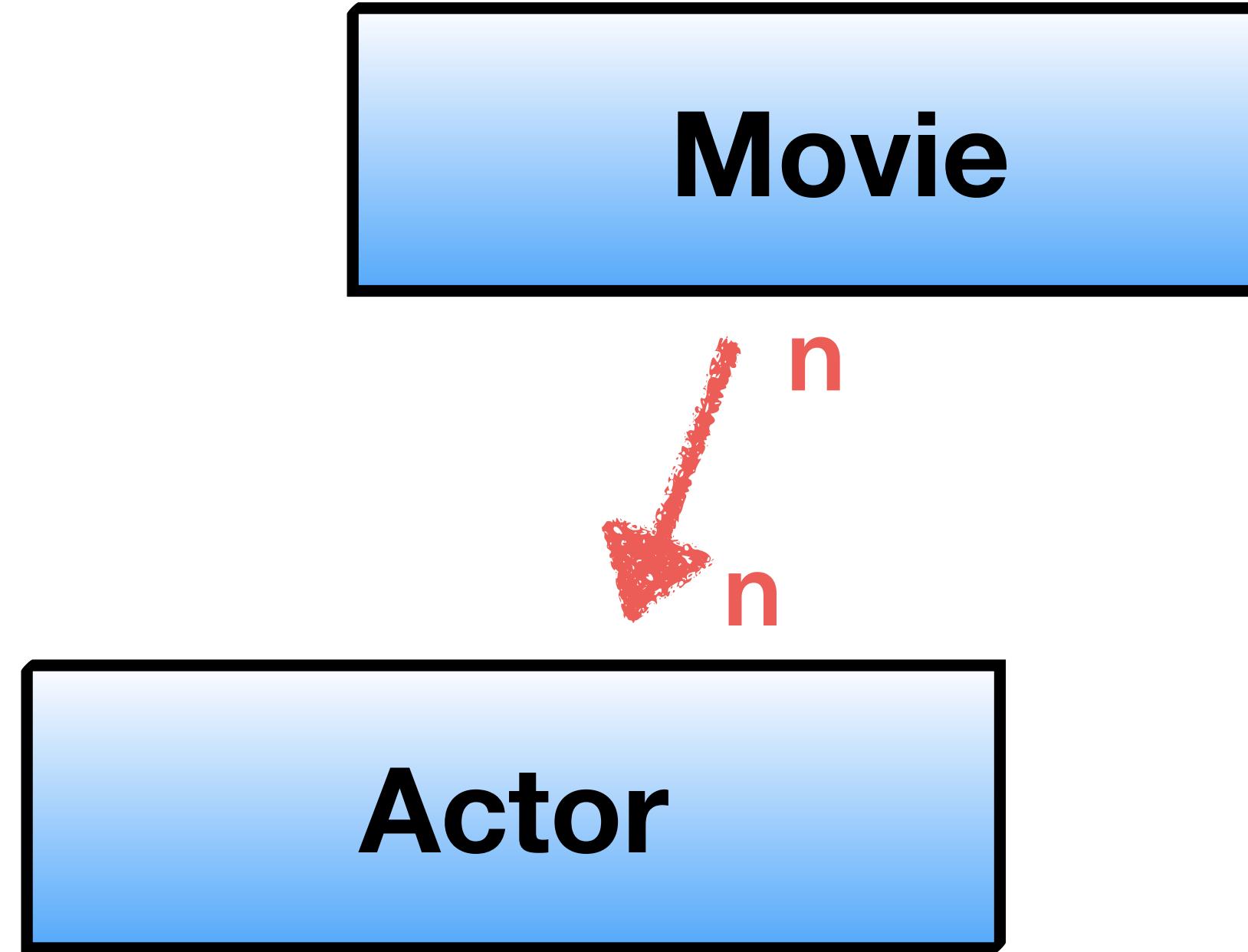


<b>id</b>	<b>name</b>
1	Paramount
2	Lucasfilm
3	Universal

<b>id</b>	<b>studio_id</b>	<b>title</b>
88	1	Raiders of the Lost Ark
89	3	Apollo 13
90	2	Jurassic Park
91	2	Star Wars

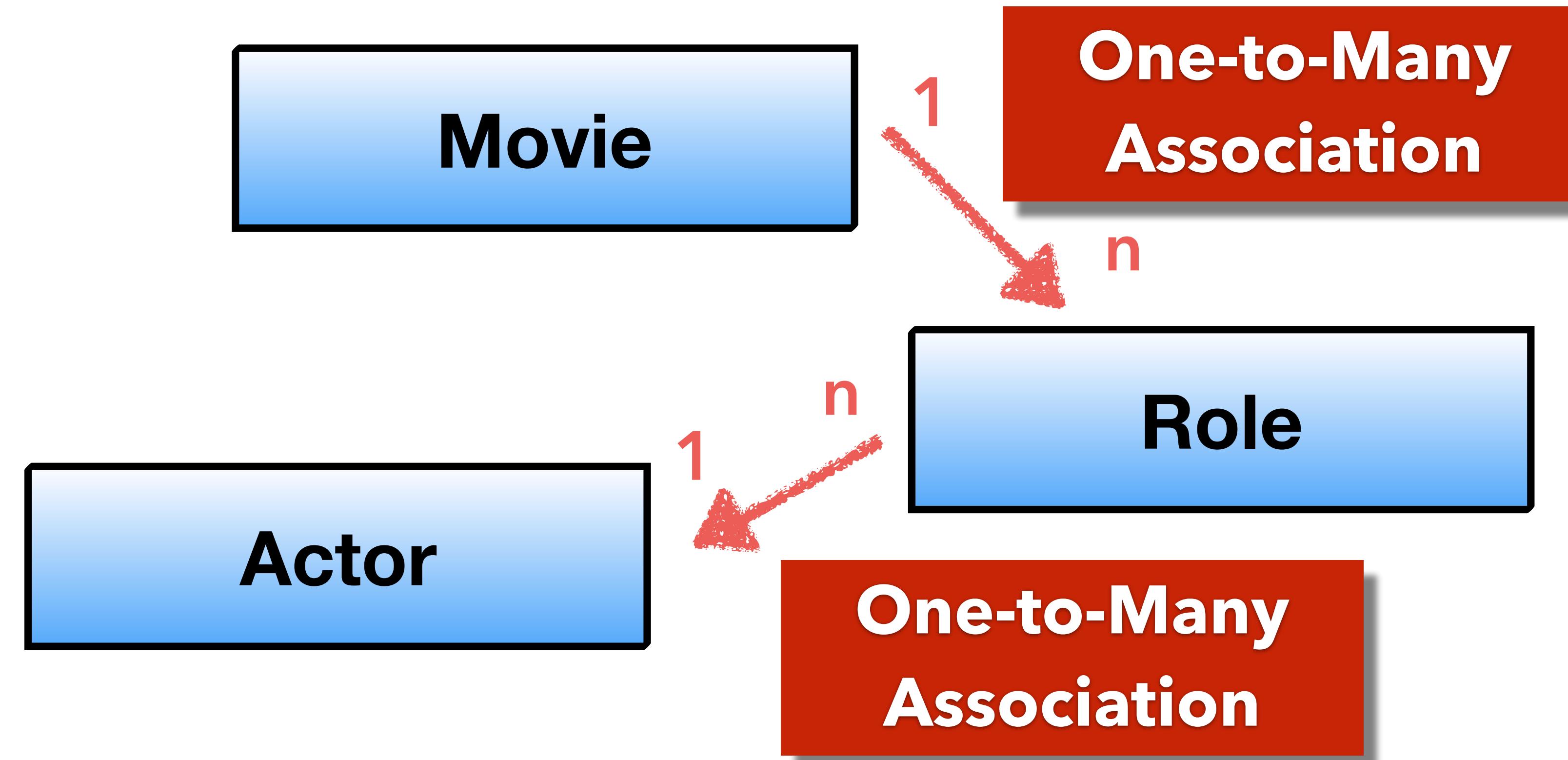
**Movie**

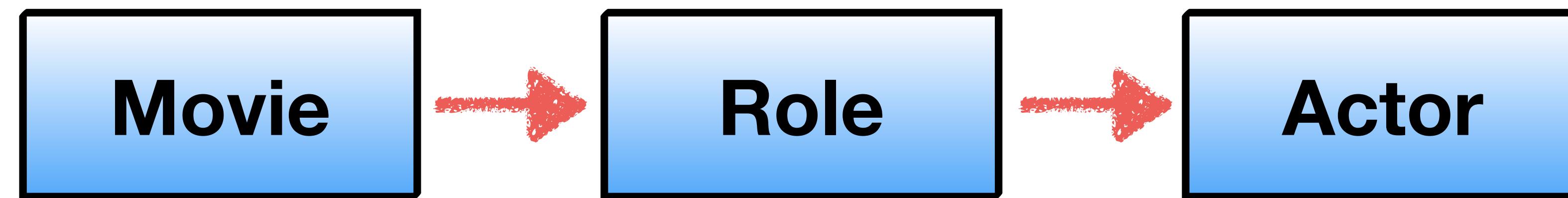
**Actor**



## Many-to-Many Association

*The connection becomes another model.*





<b>id</b>	<b>title</b>
88	Raiders of the Lost Ark
89	Apollo 13
90	Jurassic Park
91	Star Wars

<b>id</b>	<b>movie_id</b>	<b>actor_id</b>
14	88	51
15	91	54
16	91	51
17	91	53

<b>id</b>	<b>name</b>
51	Harrison Ford
52	Tom Hanks
53	Carrie Fisher
54	Mark Hamill



A Movie belongs\_to a Studio

A Movie has\_many Roles

A Movie has\_many Actors through Roles

A Studio has\_many Movies

A Role belongs\_to a Movie

A Role belongs\_to an Actor

An Actor has\_many Roles

An Actor has\_many Movies through Roles



# The Dance - Buyflix

git checkout -f .

git clean -df .

git pull

bundle install

rake db:migrate

rake db:seed

# Authentication and Authorization

Authentication = Let me in

Authorization = Let me do stuff

# Authentication

# Authentication

- Username and password
- OAuth (i.e. Login With Facebook)

# Username and Password Strategy

- Is there a user with the email b-eng@kellogg.northwestern.edu?
- Is that user's password "secret"?

# Sessions Resource

`new` = Sign-in page

`create` = Do authentication

`destroy` = Sign-out

# Security LOL

Plain text passwords FTL

Encryption FTW

No amount of encryption will help if your password is  
“password”

# Cryptographic Hash Security

- Lots of options: SHA1, Bcrypt, MD5, PBKDF2, LMAO, OMG
- Don't invent your own
- Speed is the enemy

# **TL;DR**

**Use Bcrypt**

# Session State

- HTTP is *stateless*
- We (developers) trick the browser



DANCE WITH ME, LOIS.  
DANCE THE DANCE OF LIFE.

# The Dance (Buyflix)

git checkout -f .

git clean -df .

git pull

bundle install

rake db:migrate

rake db:seed

# Authorization

# Git

Development  
Environment

Github

# Development

Git Local Repository

# Github

Git Remote

# Development

Git Local Repository

~/code/buyflix

# Github

Git Remote

<https://github.com/kie1925/imdb>

# Development

Git Local Repository  
(master)

~/code/buyflix

# Github

Git Remote  
(origin)

<https://github.com/kie1925/imdb>

**git add -A  
git commit**

Add and commit changes to local repository (master)

**git push origin master**

Push my local repository to the remote



Put ur stuff on the  
intarwebs

# DNS

Domain Name System

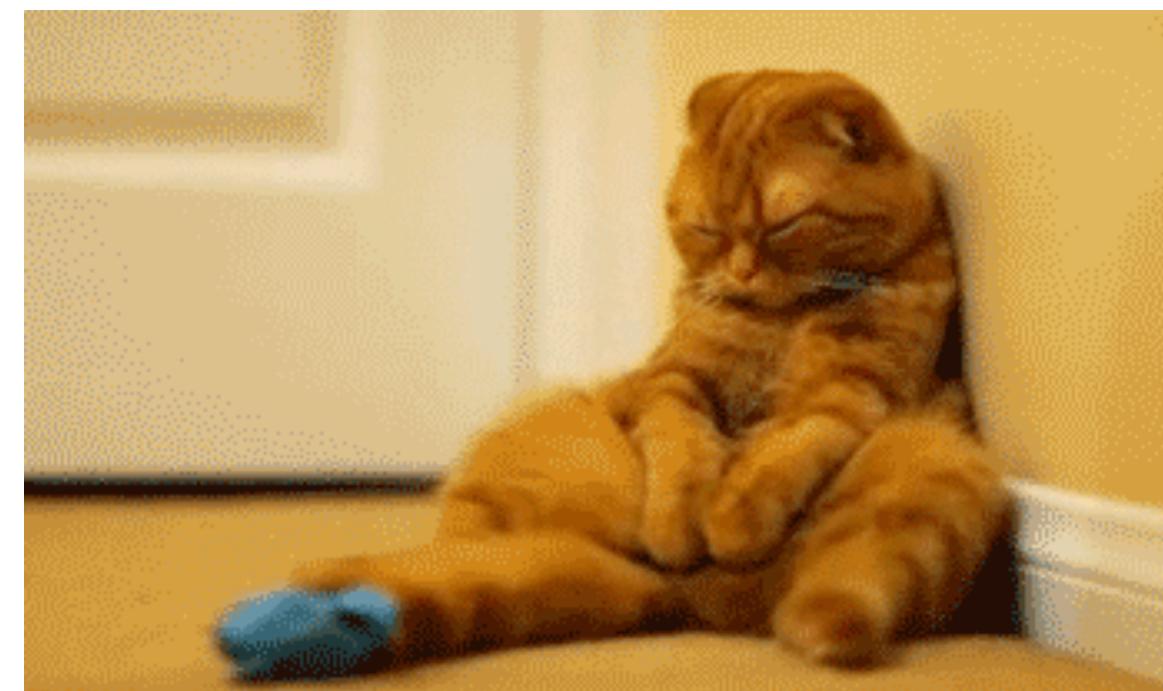
# How does the phone book work?

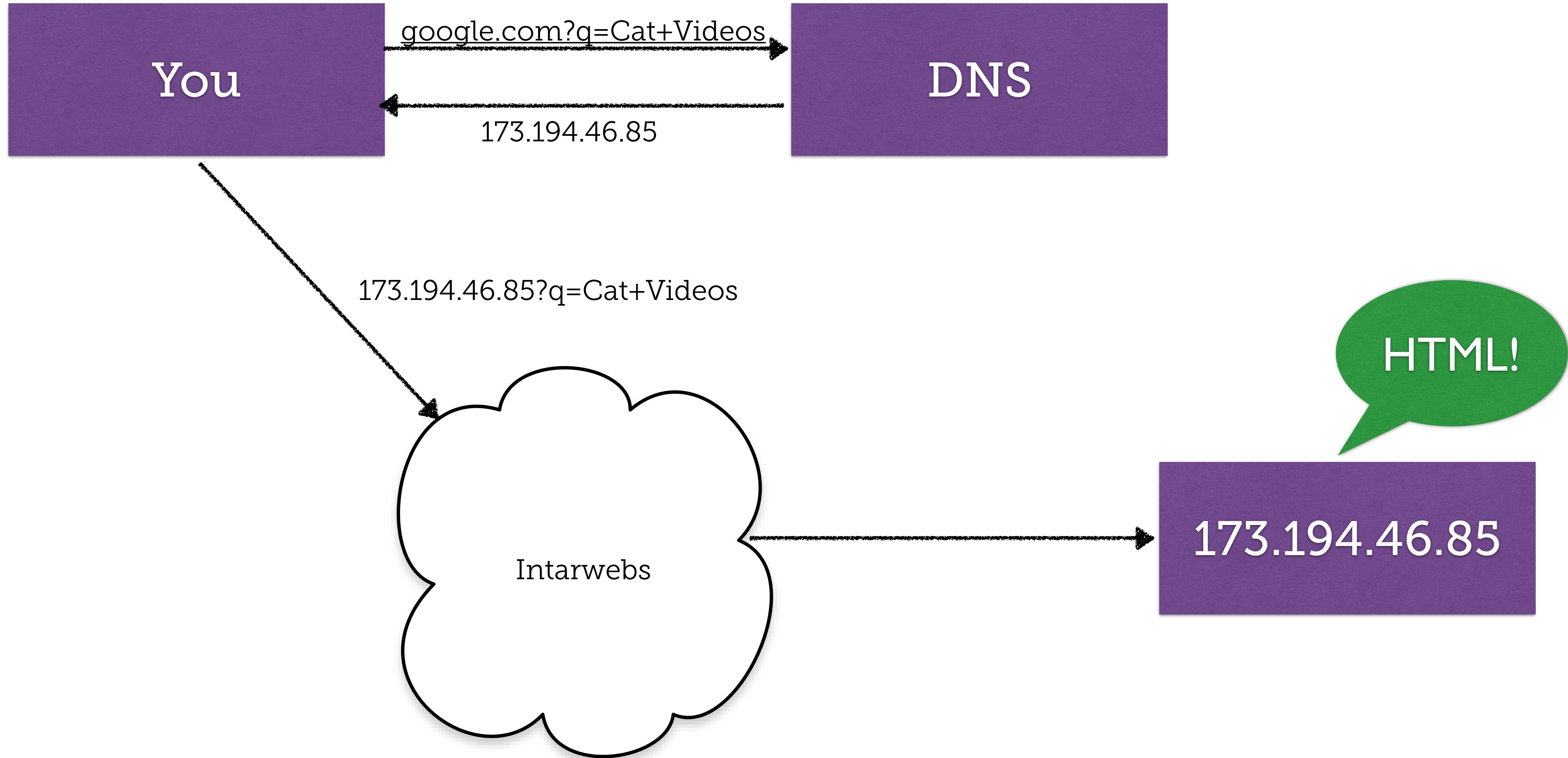
(That is, how did they work when people still used phone books?)

A day in the life of a web  
request

You

"Cat Videos"





# Hosting

Not ur laptop

The cloud = computers on  
the ground





*Your entire business*

# Stack

Operating System

Web Server

Application Server

Database Server

More...

# Development Stack

Mac/Windows

WEBrick

SQLite

# Production Stack

Ubuntu Linux

Apache/nginx/Unicorn

PostgreSQL/mySQL

# Heroku

# Platform-as-a-Service

Heroku

Command-line tools  
Git repository  
Server provisioning  
Etc...

Amazon EC2

Hosting

# My Laptop

Git Local Repository  
(master)

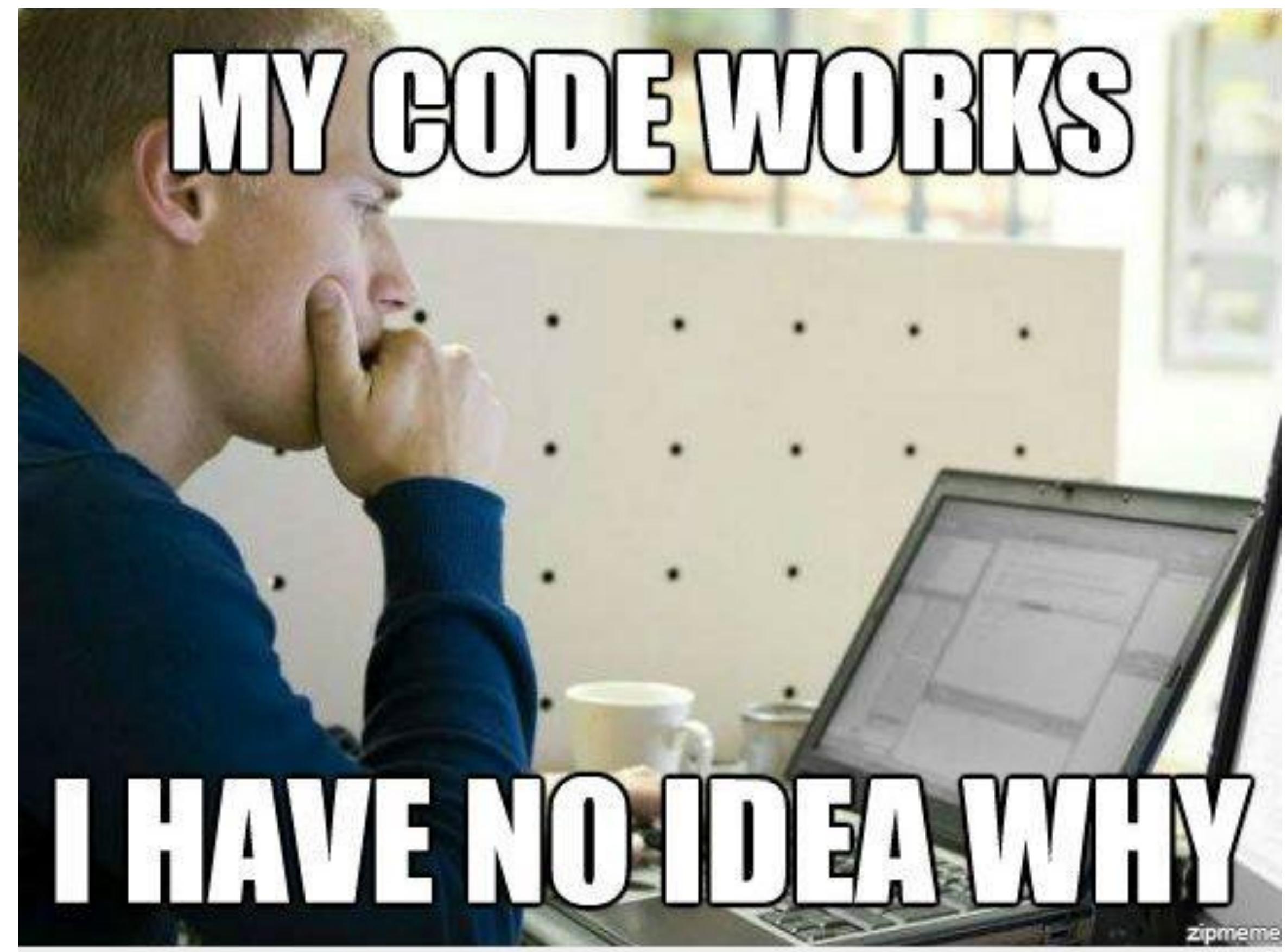
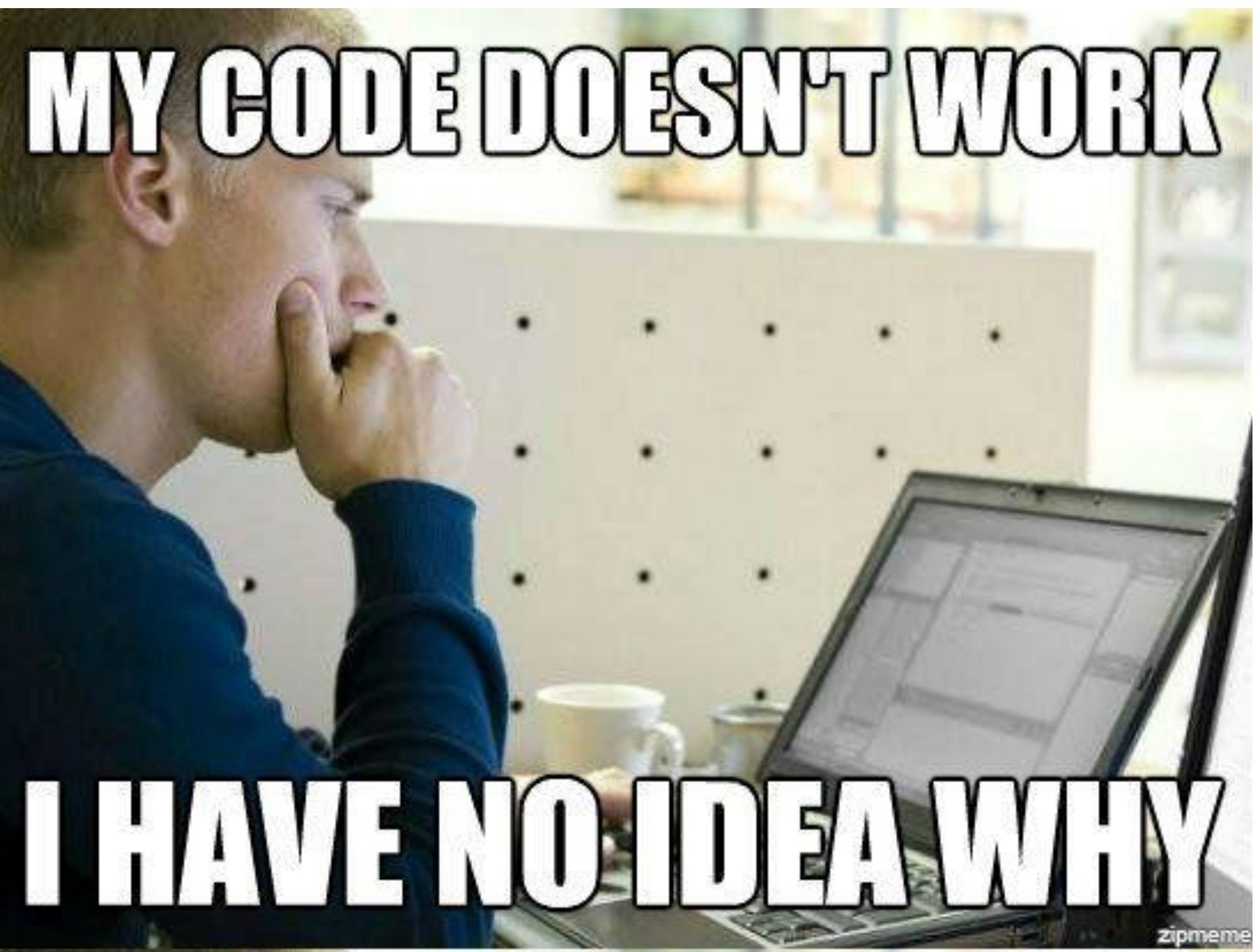
# Github

Git Remote  
(origin)

# Heroku

Git Remote  
(heroku)

**Let's do this.**



# The Dance (Buyflix)

git checkout -f .

git clean -df .

git pull

bundle install

rake db:migrate

rake db:seed

# Code Kata: Part I

- Modify `MoviesController#create` to create actors/roles along with the movie
- Create two movies with the same actor; make sure this doesn't create two actors with the same name
- Hint: Use `String#split` to turn a string into an Array

# Code Kata: Part II

- Modify `MoviesController#create` to create genres/categories along with the movie
- Create two movies in the same category (genre); make sure this doesn't create two genres with the same name

**YAGNI**

# YAGNI

(You Ain't Gonna Need It)

**DRY**

# The Dance (Buyflix)

git checkout -f .

git clean -df .

git pull

bundle install

rake db:migrate

rake db:seed

# E-Commerce

# Core Concepts

- How credit card processing works
- Compare approaches
- Case studies/integration with Rails

# Credit card processing

Your web app

Merchant account

Payment gateway

**WE BE MAKIN FAT STACKS**



Authorize

Capture

Purchase (Authorize + Capture)

Void

Credit

# Use Cases

Sell things

# Sell things

- Charge the card
- Handle declines
- Save customer/payment information for future purchases
- Handle refund/credit
- Handle chargebacks (disputes)

# Subscriptions

# Subscriptions

- Store customer details
- Subscribe customer to a plan – specify billing frequency and amount
- Handle trial period
- Handle declines
- Handle credit card expiration
- Handle change/cancel
- Handle coupons/discounts
- Metered billing
- One-time charges
- Per-user pricing
- Chargebacks



PCI Compliance!

# PCI Compliance

If you accept payments – need to maintain  
compliance at all times

# PCI Compliance

- Standard established in 2006 by all the major companies (Mastercard, Visa, American Express, JCB, etc.) in an effort to combat credit card fraud
- Needed if you:
  - Accept payment online
  - Credit card information posted to your server
  - Store credit card information

# PCI Levels

- Level 1: Over 6 million transactions per year
- Level 2: 1-6 million
- Level 3: 20,000-1,000,000
- Level 4: Less than 20,000

# PCI Data Security Standards (DSS)

## 12 Requirements

# PCI DSS Requirements

- Cannot store any card information including CVV, PIN, mag strip data
- Use SSL (https)
- Host behind a firewall
- Keep good logs
- Apply security patches

# PCI DSS Level 4

- Annual PCI Self-assessment Questionnaire
- Optional – quarterly network scan

# What you need to start charging

- ✓ Incorporate your business
- ✓ Get an EIN (tax ID number)
- ✓ State business license
- ✓ Bank account
- ✓ Merchant account/payment gateway
- ✓ Ensure PCI Level 4 compliance
- ✓ Code your app

# Costs and Options

# Easy ways!

- Customer never touches your server
- PayPal
- Amazon Payments
- Trusted by customers
- UX – meh
- Kickstarter – ~\$300 million in 2012
- 2.9% + \$0.30

# Medium Difficulty

- PayPal Payments Advanced
- PayPal Payments Pro - Virtual Terminal
- Braintree - Ignition
- Stripe
- 2.9% + \$0.30

# Benefits

- All-inclusive
- Developer-friendly
- Simplifies PCI compliance

# Roll your own

- Your own merchant account/payment gateway (e.g. Authorize.net, TrustCommerce, etc.)
- Lower fees (2.19% and lower)
- Handle all the things

# Benefits

- Fee structure
- Negotiable
- Total control

# ActiveMerchant

gem 'activemerchant'  
bundle install

```
# Send requests to the gateway's test servers
ActiveMerchant::Billing::Base.mode = :test

# Create a new credit card object
credit_card = ActiveMerchant::Billing::CreditCard.new(
  :number      => '4111111111111111',
  :month       => '8',
  :year        => '2009',
  :first_name  => 'Tobias',
  :last_name   => 'Luetke',
  :verification_value => '123'
)

if credit_card.valid?
  # Create a gateway object to the TrustCommerce service
  gateway = ActiveMerchant::Billing::TrustCommerceGateway.new(
    :login      => 'TestMerchant',
    :password   => 'password'
  )

  # Authorize for $10 dollars (1000 cents)
  response = gateway.authorize(1000, credit_card)

  if response.success?
    # Capture the money
    gateway.capture(1000, response.authorization)
  else
    raise StandardError, response.message
  end
end
```

Spreedly

# Spreedly

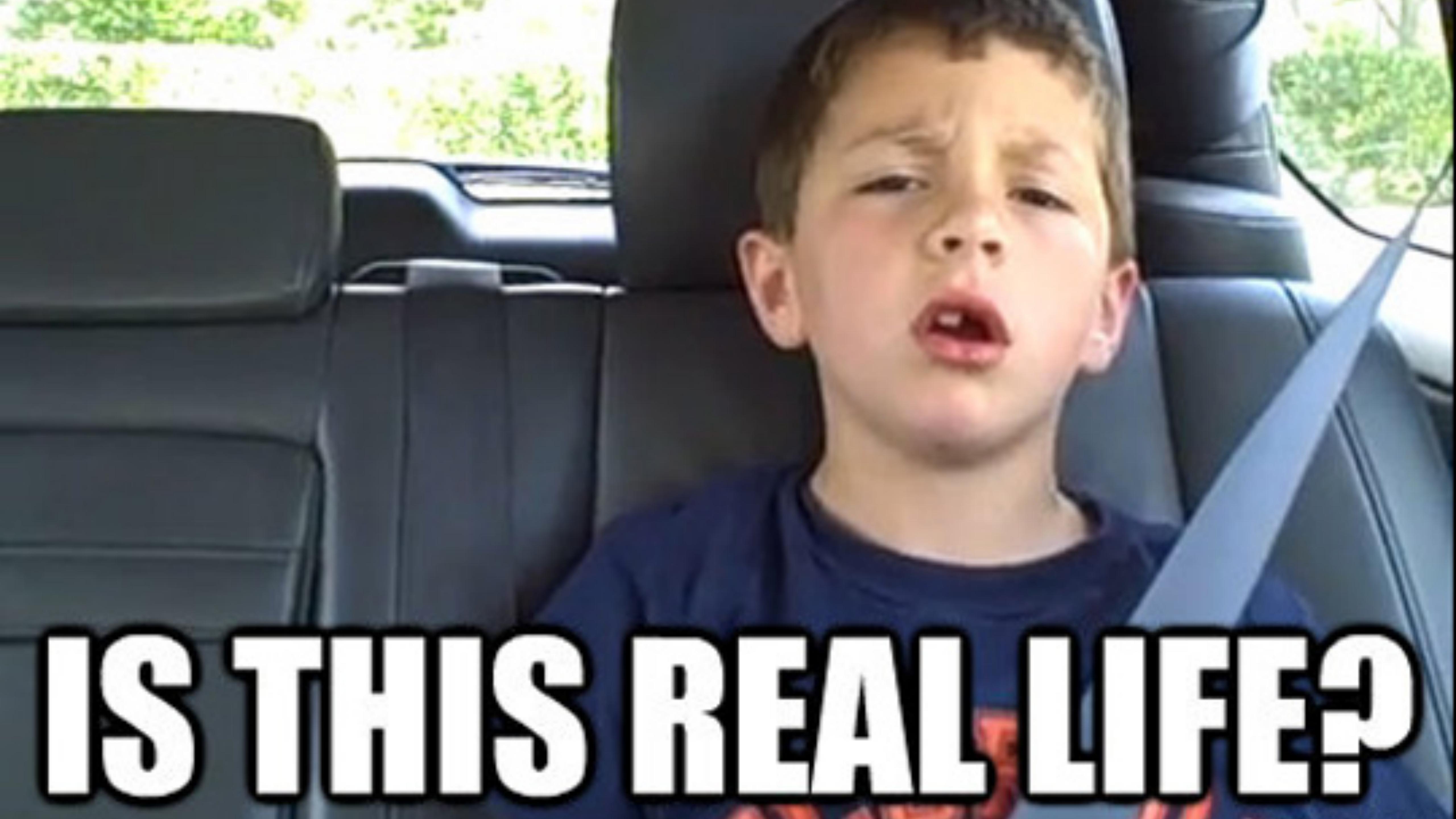
- Data “vault”
- Unified API to 50+ payment gateways
- Pricey (\$150/mo++)

# Stripe

# Stripe Benefits

- Standard pricing (2.9% + \$0.30)
- API for all the use cases
- Hand-holding through PCI compliance
- Built for developers

# Stripe Demo

A young boy with short brown hair is sitting in a dark-colored car seat. He is wearing a blue t-shirt and has a surprised or excited expression, with his mouth slightly open. The background shows the interior of a car with a window and some greenery outside.

**IS THIS REAL LIFE?**

Name	Lines	LOC	Classes	Methods	M/C	LC
Controllers	6992	4731	119	641	5	
Helpers	1977	787	1	99	99	
Models	9433	5696	125	622	4	
Mailers	65	55	2	8	4	
Javascripts	1814	1525	0	246	0	
Libraries	762	519	11	66	6	
Controller tests	312	36	12	0	0	
Helper tests	168	126	42	0	0	
Model tests	4092	1563	76	38	0	
Mailer tests	7	3	1	0	0	
Integration tests	0	0	0	0	0	
Total	25622	15041	389	1720	4	

# Rails IRL

- Asset Pipeline
- No “ez” gem - migrations and versioning
- More sophisticated development environment

4 20 00

HEATHKIT

