Developing Web Apps
with Ruby on Rails

Prof. Paul Krause, University of Surrey
Chapter 4, Lecture 1
Using Active Record

Objectives for today
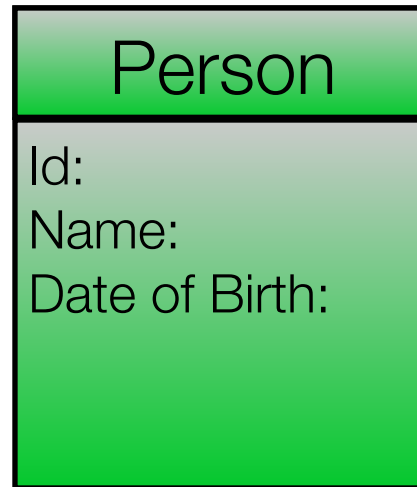
- Remove our static Events page and replace it with an index page of dynamically generated Events

- This will introduce Rails' Active Record and its:

  - Object-Relation Mapping

  - Support for Database Migrations

  - Scaffold for generating everything you need to support an Events resource

Object-Relational Mapping

- Classes map to Tables

- Attributes map to Columns in the respective Table

- Objects map to Rows

- Rails will handle the pluralisation automatically for you

  - (and usually gets it right)

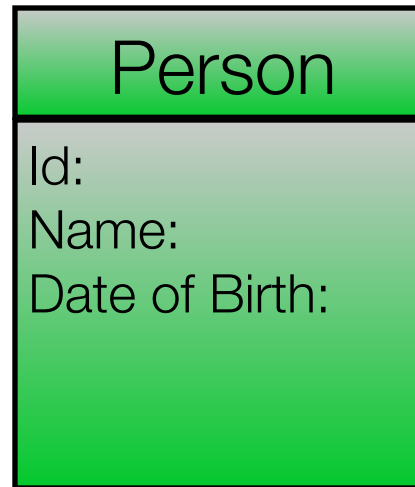| Person |
|---|
| Id:
Name:
Date of Birth: |

Object-Relational Mapping

- Classes map to Tables

- Attributes map to Columns in the respective Table

- Objects map to Rows

- Rails will handle the pluralisation automatically for you

  - (and usually gets it right)

Person
Id: Name: Date of Birth:

People		

Object-Relational Mapping

- Classes map to Tables

- Attributes map to Columns in the respective Table

- Objects map to Rows

- Rails will handle the pluralisation automatically for you
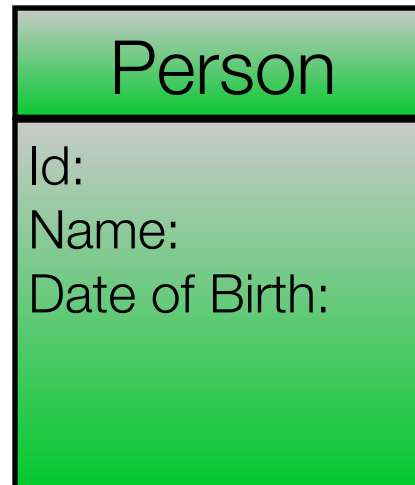
    - (and usually gets it right)

Person
Id: Name: Date of Birth:

People		
ID	Name	DoB

Object-Relational Mapping

- Classes map to Tables

- Attributes map to Columns in the respective Table

- Objects map to Rows

- Rails will handle the pluralisation automatically for you
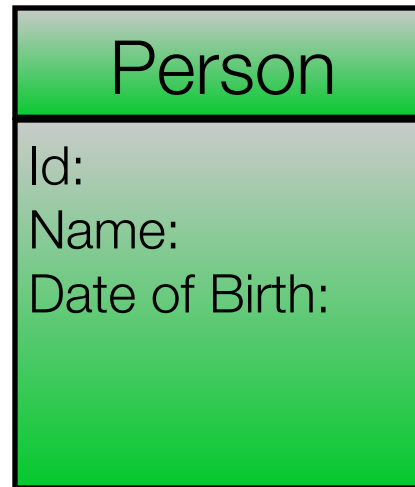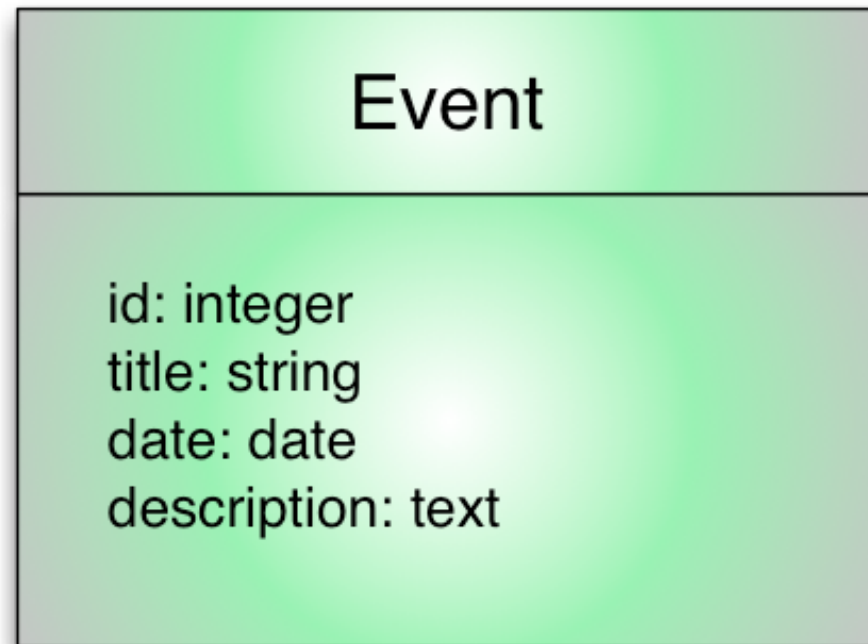
  - (and usually gets it right)

Person
Id: Name: Date of Birth:

People		
ID	Name	DoB
1	Paul	c1900

The Event Model

Event

id: integer
title: string
date: date
description: text

What do we want to do (with Events)?

- Create

  - a new Event

- Read

  - an existing Event

- Update

  - an existing Event

- Destroy

  - an existing Event

How do we set up the database?

- Do nothing!

- Rails already has SQLite3 bundled and will generate the database for you

The Prolog

- Start a new branch in your Git repository

- Strip out the:

  - Events action in the application controller

  - Events view

  - /events route in config/routes.rb

Build a CRUD interface for the Event model

rails generate scaffold Event title:string date:date description:text

Build a CRUD interface for the Event model

rails generate scaffold Event title:string date:date description:text

All
sorts of things
are going
on here!

Mapping Migration types to Database types

Migration type	MySQL	SQLite	Ruby Class
:binary	blob	blob	String
:boolean	tinyint(1)	boolean	Boolean
:date	date	date	Date
:datetime	datetime	datetime	Time
:decimal	decimal	decimal	BigDecimal
:float	float	float	Float
:integer	int(11)	integer	Fixnum
:string	varchar(255)	varchar(255)	String
:text	text	text	String
:time	time	time	Time
:timestamp	datetime	datetime	Time

Run the database migration

rake db:migrate

A set of routes for events in one statement
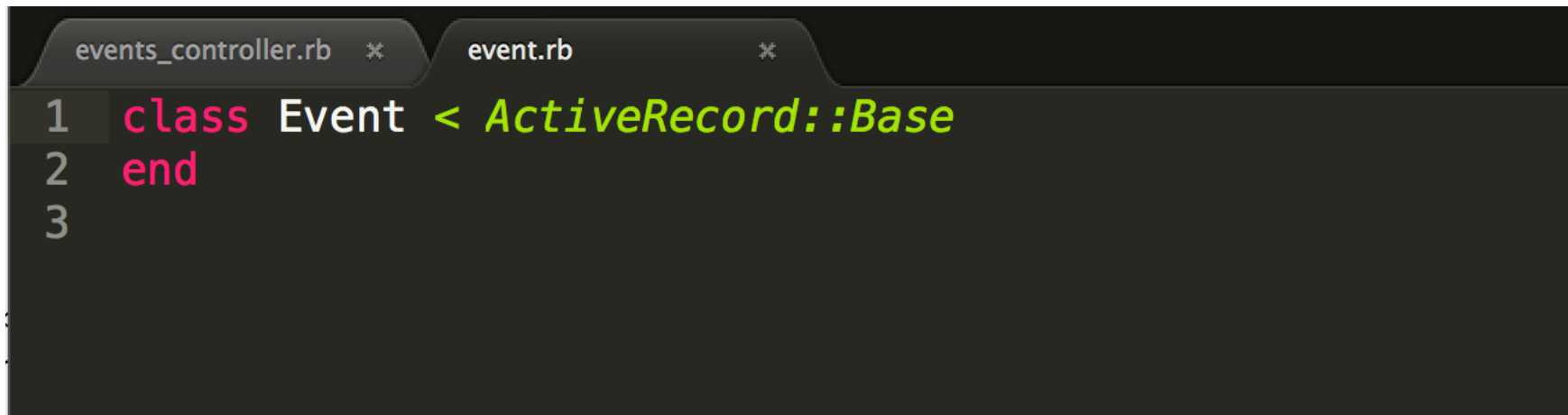
```ruby
events_controller.rb    event.rb    routes.rb

 1  AAAWebsite::Application.routes.draw do
 2    resources :events
 3
 4    match "/index" => "aaa_core#index", via: :get
 5    match "/shop" => "aaa_core#shop", via: :get
 6    match "/discussion" => "aaa_core#discussion", via: :get
 7    # The priority is based upon order of creation: first created -> highest priority.
 8    # See how all your routes lay out with "rake routes".
 9
10    # You can have the root of your site routed with "root"
11    root 'aaa_core#index'
12
```

A little surprise!

- models/event.rb

```
class Event < ActiveRecord::Base
end
```

- *Metaprogramming* enables Active Record to automatically add features to the model classes: e.g. it adds attributes to the model classes based on the columns in the corresponding database tables.

White list for parameters from the scary Internet
In: app/controllers/events_controller.rb

```ruby
64  private
65    # Use callbacks to share common setup or constraints between actions.
66    def set_event
67      @event = Event.find(params[:id])
68    end
69
70    # Never trust parameters from the scary internet, only allow the white list through.
71    def event_params
72      params.require(:event).permit(:title, :date, :description)
73    end
```

What we have done

- This has introduced Active Record

    - Probably the most impressive feature of Rails

- And we didn't even write one line of code!