

# Getting Started with Ruby on Rails

---

Prof. Paul Krause, University of Surrey

Chapter 1: Lecture 1

Setting the Scene

# Objectives of the course

---

- We aim to bring you to a “high level of proficiency in developing interactive web applications using the Ruby on Rails framework”
- We would also like you to gain understanding of:
  - Agile methods for developing software
  - Best practice in the use of HTML5 and CSS3

# Objectives for today

---

- Why is Ruby on Rails interesting?
- Basic introduction to Ruby
  - But more programming exercises next time
- Introduce Rails
  - But we will start to explore it in much more detail in the next Chapter

Why is Ruby on Rails Interesting?

# Is Ruby on Rails really that special?

---

- From Obie Fernandez' introduction to *The Rails Way*:
  - “That first Rails experience, over the course of a few weeks in February 2005, was life-altering”
  - “All the best practices I had learned over the years about building web apps had been distilled into a single framework, ...”
  - “My interest in Java died a sudden death.”

# Opinionated Software?

---

- Convention over configuration
- Elegance is not optional
- Do Not Repeat Yourself
- Developer motivation and productivity are primary factors in project success

# And the Icing

---

- Ruby Gems
- Ruby Gems
- and more Ruby Gems

Let's take a look at Ruby



# Key Features of Ruby

---

- Easy syntax
- Fully OO
- Highly dynamic
- Strong Web frameworks
  - Ruby on Rails

# Let's Try it!

---

- Open a Ruby shell
  - Oops, I need to show you how!
  - I will do that in the next lecture, but for the moment just sit back and watch it all happen!

# “Hello World” in Java

---

```
class HelloWorld {  
  
    public static void main(String[] args) {  
  
        System.out.println("Hello World!")  
  
    }  
  
}
```

# and in Ruby:

---

```
puts "Hello World!"
```

# Even Better

---

Enter: `3.times {puts "Hello World!"}`

# Ruby's Objects

---

- Enter: 4 (return)
- Enter: 4.567.round
- Find out the class of something you have entered...

# Ruby is Dynamically Typed

---

- Enter: `n=1`
- Enter: `n.class`
- Enter: `n="fishpaste"`
- Enter: `n.class`
- How about: `n+4; n.size; n.methods`

# Regular Expressions

---

- Regular expressions are written between slashes
- Enter: `regex = /better/`
- Find out what class it is.
- Enter: `"Mine is bigger" =~ regex`
- Enter: `"Mine is better" =~ regex`



# Containers

---

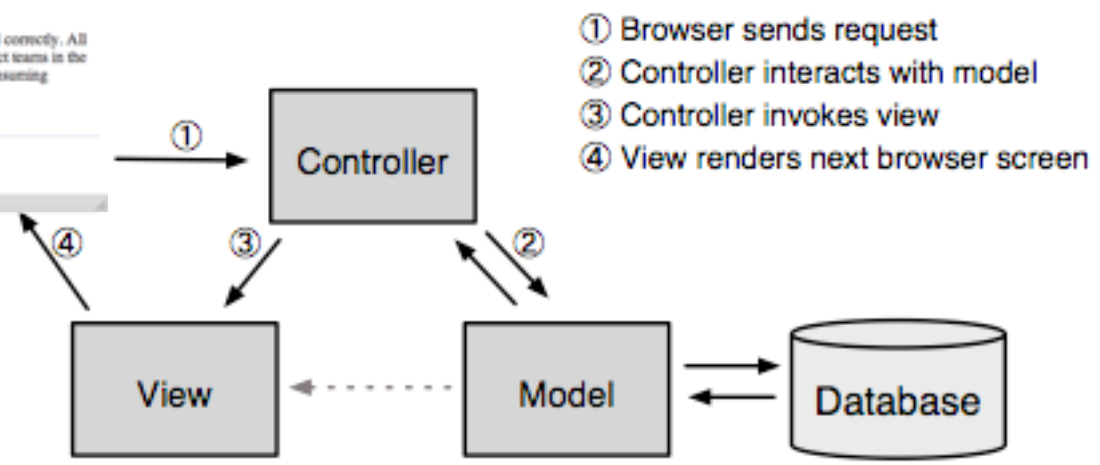
- Enter: `stack = [1, 2, 3]`
- Enter: `stack.push "cat"`
- Enter: `stack.pop`
- Enter: `stack`

# That's very nice, but

---

- Ruby was released by its designer *Yukihiro Matsumoto* in 1995
- Why is it only now, that it is generating so much interest?
- Let's explore Rails a little bit ...

# The Rails Framework



And how do we ensure we are creating quality products?

# The Answer

---

- All Agile Samurai use Test Driven Development



## The Agile Samurai

How Agile Masters  
Deliver  
Great Software



**Jonathan Rasmusson**

*Edited by Susannah Davidson Pfister*

# Process, process, process

---

- Release frequent increments
- Write your test cases before you code
- Manage your versions in Git and GitHub
- Deploy into Heroku



# Break Time

---

- We have covered some basics of Ruby
- We have very briefly introduced the Ruby on Rails framework
- Let's take a short break and then come back to exploring Ruby