Rails Meetup 2020

Hoelzle AG
Gregor Wassmann

github.com/hoelzle/rails-meetup-2020

Agenda

- Decorators vs Concerns
- Surprise

You should use…

```
class Price < SimpleDelegator
end
```

Thoughtbot — Evaluating Alternative Decorator Implementations In Ruby

Decorators

- Module + Extend + Super
- Plain Old Ruby Object (PORO)
- Class + Method Missing
- **SimpleDelegator + Super + Getobj**

Pros

- Encapsulation
- Easily Testable
- Tranparent interface
- Simple
- No dependencies

It redefines `class`, but that may be the only drawback…

```
class Decorators::Price
class Decorators::SpecialPrice
class Decorators::CustomerDiscountPrice
class Decorators::PromotionPrice
class Decorators::ProductDiscountPrice
class Decorators::GroupDiscountPrice
class Decorators::RegularPrice
```

```
class ProductsController < ApplicationController
  def index
    @products = Decorators::CustomizationsPreload.new(
      @products, current_customer, Decorators::Price
    )
  end
end
```

Refinements

```ruby
module DiscountFactor
  refine Numeric do
    def factor
      1 - self / 100.0
    end
  end
end
```

```ruby
module Decorators
  class RegularPrice < Decorators::Context
    using DiscountFactor

    def net_price
      price * discount.factor
    end

    # Omitted code here
  end
end
```

See also

- Draper
- Trailblazer

github.com/hoelzle/rails-meetup-2020

Thanks Rails Community ♥

What I learned at the meetup was:

- https://crystal-lang.org/
- https://www.luckyframework.org/