

# Therapeutic Refactoring

***Katrina Owen***  
*Jumpstart Lab*  
*@kytrinyx*







Illustration from **Am Adult** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/2010/03/am-adult.html>



Photo: Johanna Owen 2011

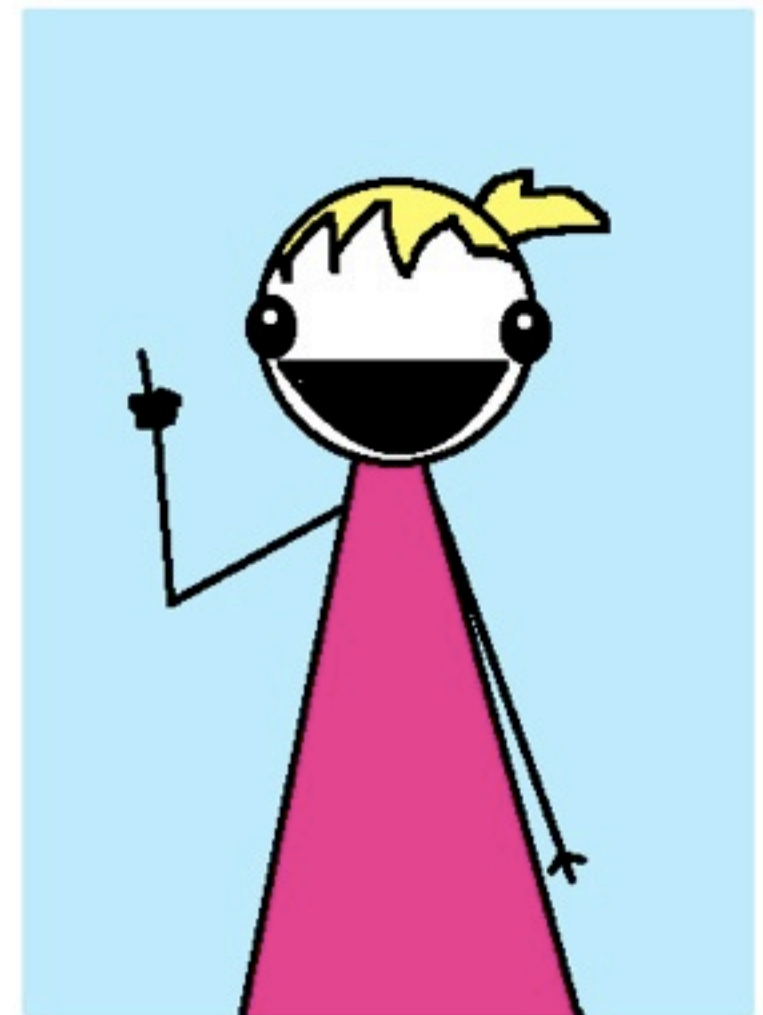
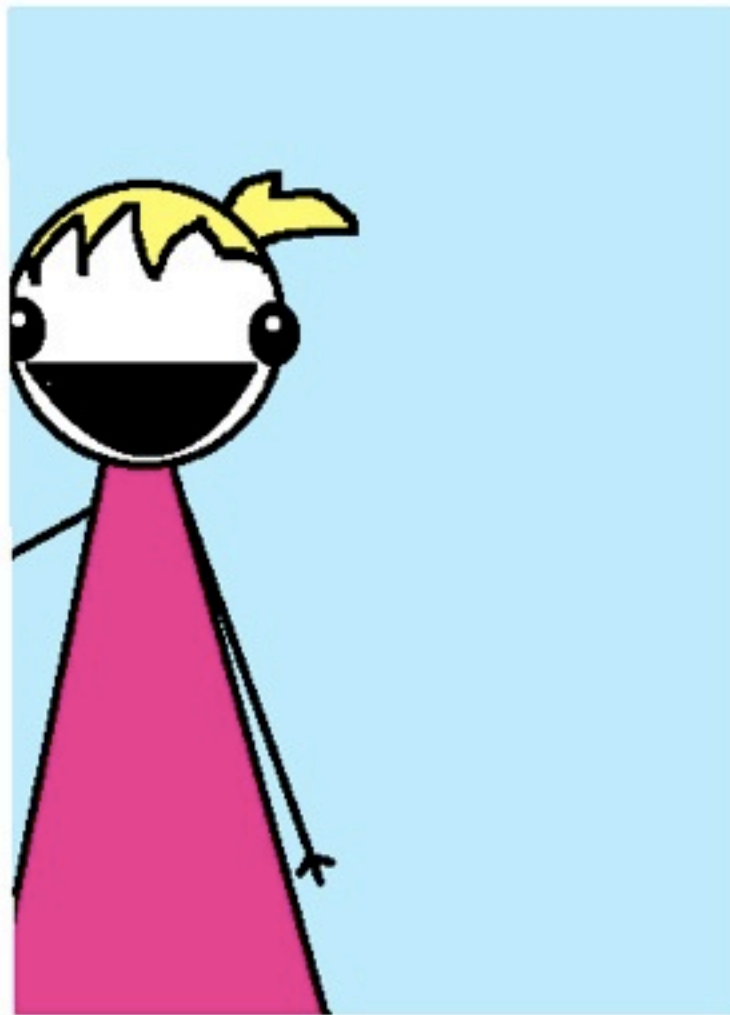
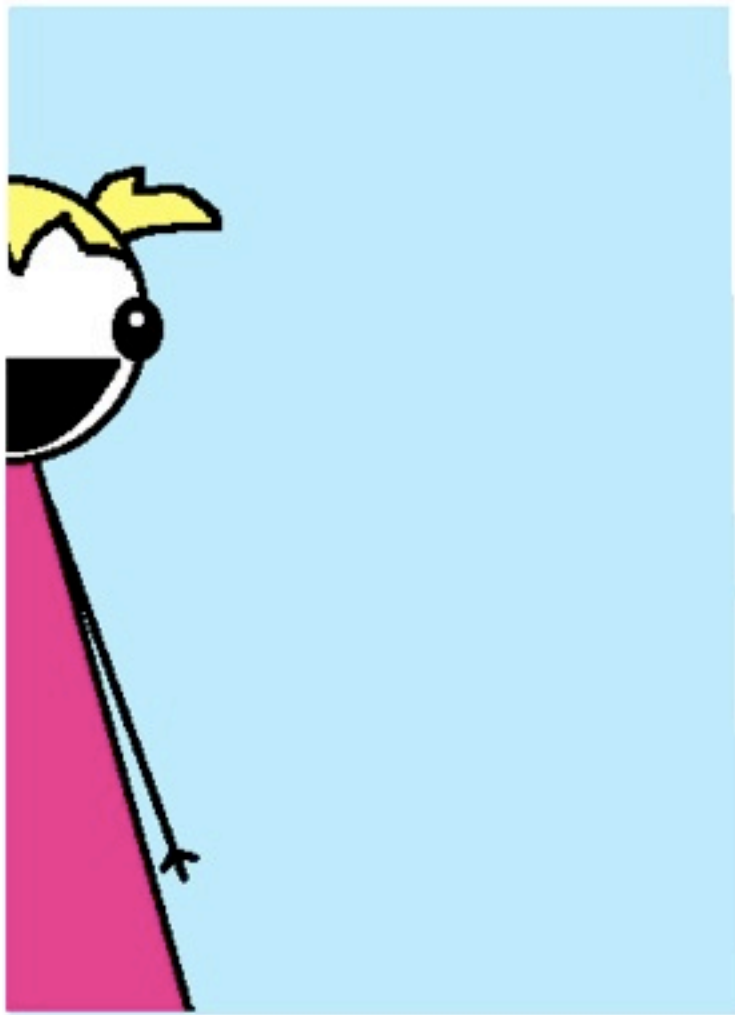
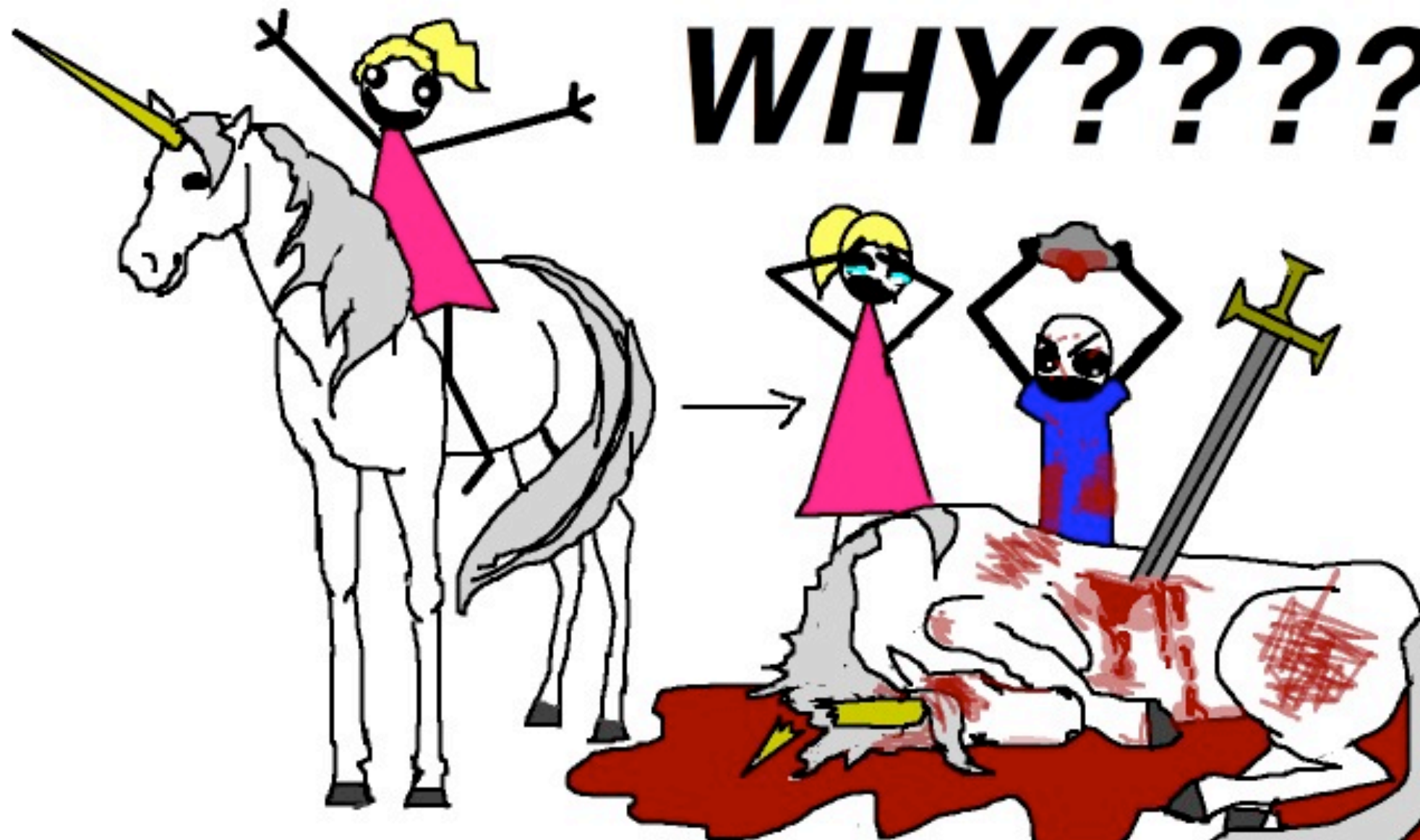


Illustration from **U B Hatin'** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/2010/03/u-b-hatin.html>



**refactoring**





***WHY????***

Illustration from **There's a Lesson in Here Somewhere, I Think** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/2010/01/theres-lesson-in-here-somewhere-i-think.html>

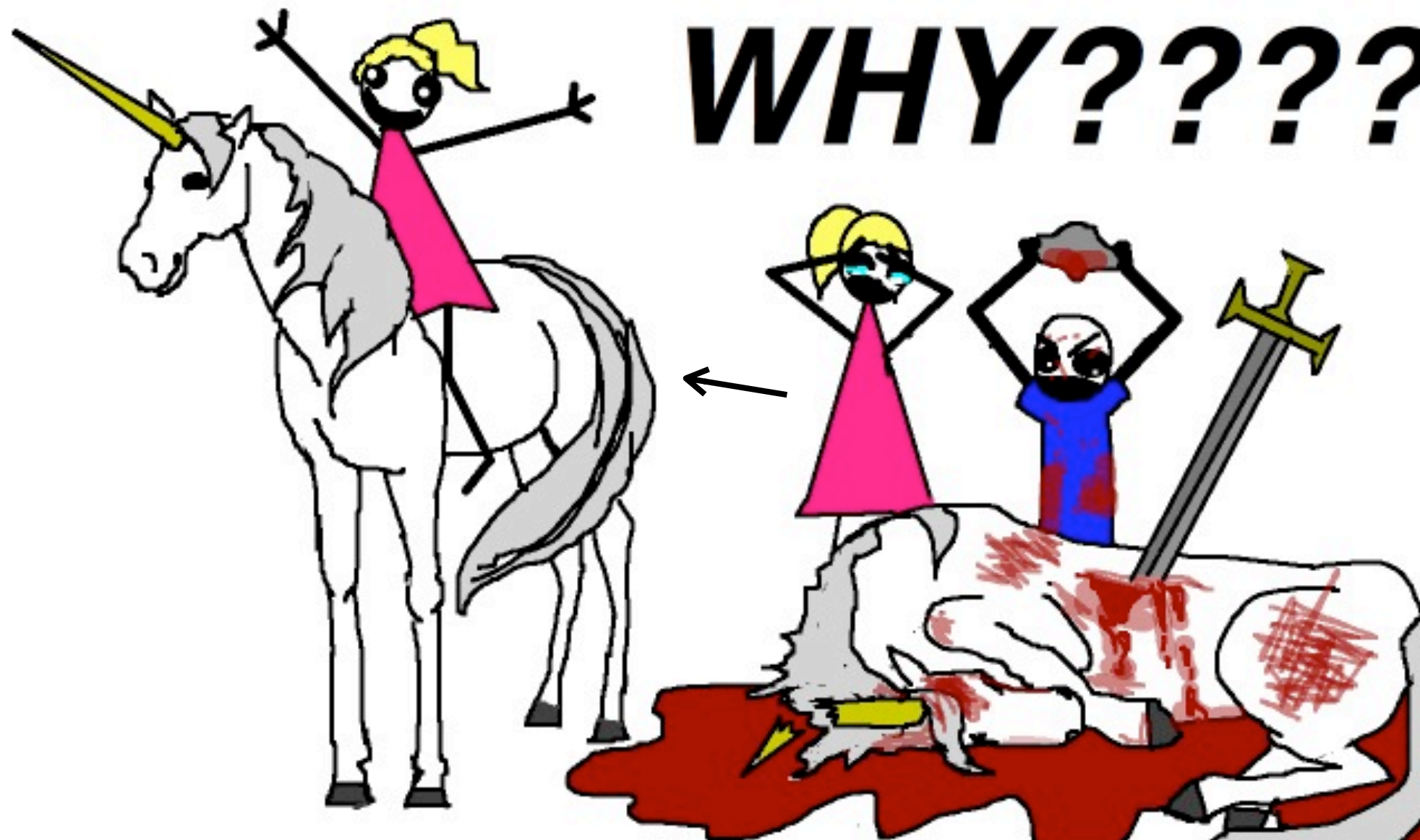


Illustration from **There's a Lesson in Here Somewhere, I Think** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/2010/01/theres-lesson-in-here-somewhere-i-think.html>



**refactoring**  
**is not**  
**rehacktoring**

# A Story

*Inspired by true events*



Illustration from **The Scariest Story** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/2011/02/scariest-story.html>

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^\[a-z\]]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```

There's Blood everywhere.



Illustration from **The Scariest Story** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/2011/02/scariest-story.html>



It's your blood.



Illustration from **The Scariest Story** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/2011/02/scariest-story.html>

And you cry, and your  
tears are also blood.



Illustration from **The Scariest Story** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/2011/02/scariest-story.html>

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^\[a-z\]]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```

```
module XYZService
```

```
def self.xyz_filename(target)
```

```
  # File format:
```

```
  # [day of month zero-padded][three-letter prefix] \
```

```
  # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
  # _[8 random chars]_[10 first chars of title].jpg
```

```
  filename = "#{target.publish_on.strftime("%d")}"
```

```
  filename << "#{target.xyz_category_prefix}"
```

```
  filename << "#{target.kind.gsub("_", "")}"
```

```
  filename << "_%03d" % (target.age || 0) if target.personal?
```

```
  filename << "_#{target.id.to_s}"
```

```
  filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
  truncated_title = target.title.gsub(/^[^a-z\ ]+/i, '').downcase
```

```
  length = truncated_title.length
```

```
  truncate_to = length > 9 ? 9 : length
```

```
  filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
  filename << ".jpg"
```

```
  return filename
```

```
end
```

```
end
```

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^a-z\ ]+/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```



```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^a-z\ ]+/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,9]}"
```

```
    truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,9]}"
```

```
    truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^\[a-z\]]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```





# Tests

*(the first middle)*

```
describe XYZService do  
end
```

```
describe XYZService do
```

```
  it "works" do
```

```
  end
```

```
end
```

```
describe XYZService do

  subject { XYZService.xyz_filename(stub) }

  it "works" do
    end
end
```

```
describe XYZService do
```

```
  subject { XYZService.xyz_filename(stub) }
```

```
  it "works" do
```

```
  end
```

```
end
```



```
describe XYZService do

  subject { XYZService.xyz_filename(stub) }

  it "works" do
    end
end
```

```
describe XYZService do

  subject { XYZService.xyz_filename(stub) }

  it "works" do
    subject.should eq("something")
  end
end
```

```
subject.should eq  
  "something"
```

```
subject.should eq  
  "whatever"
```

```
subject.should eq  
  "freddy"
```

```
subject.should eq  
  "something"
```

## Failures:

1) XYZService works

```
Failure/Error:  
subject.should eq("something")
```

```
Stub received unexpected message :publish_on with (no args)
```

```
# ./xyz_service.rb:6:in `xyz_filename'
```

## Failures:

1) XYZService works

```
Failure/Error:  
subject.should eq("something")
```

```
Stub received unexpected message :publish_on with (no args)
```

```
# ./xyz_service.rb:6:in `xyz_filename'
```



Stub received unexpected message

: publish\_on

with (no args)

## Failures:

1) XYZService works

```
Failure/Error:  
subject.should eq("something")
```

```
Stub received unexpected message :publish_on with (no args)
```

```
# ./xyz_service.rb:6:in `xyz_filename'
```

```
# ./xyz_service.rb:6
```

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^a-z\ ]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```

```
target.publish_on.  
    strftime( "%d" )
```

```
target.publish_on.  
    strftime( "%d" )
```

```
target.publish_on.  
    strftime( "%d" )
```

```
describe XYZService do
```

```
  subject { XYZService.xyz_filename(stub) }
```

```
  it "works" do
```

```
  end
```

```
end
```



```
describe XYZService do
```

```
  subject { XYZService.xyz_filename(target) }
```

```
  it "works" do
```

```
  end
```

```
end
```

```
describe XYZService do
```

```
  let(:target) do
```

```
    stub(:target)
```

```
  end
```

```
  subject { XYZService.xyz_filename(target) }
```

```
  it "works" do
```

```
    subject.should eq("something")
```

```
  end
```

```
end
```

```
describe XYZService do
```

```
  let(:target) do
```

```
    messages = {  
      }
```

```
    stub(:target, messages)
```

```
  end
```

```
  subject { XYZService.xyz_filename(target) }
```

```
  it "works" do
```

```
    subject.should eq("something")
```

```
  end
```

```
end
```

```
describe XYZService do
```

```
  let(:target) do
```

```
    messages = {
```

```
      :publish_on => ?
```

```
    }
```

```
    stub(:target, messages)
```

```
  end
```

```
  subject { XYZService.xyz_filename(target) }
```

```
  it "works" do
```

```
    subject.should eq("something")
```

```
  end
```

```
end
```

```
describe XYZService do
```

```
  let(:target) do
```

```
    messages = {
```

```
      :publish_on => Date.new(2012, 3, 14)
```

```
    }
```

```
    stub(:target, messages)
```

```
  end
```

```
  subject { XYZService.xyz_filename(target) }
```

```
  it "works" do
```

```
    subject.should eq("something")
```

```
  end
```

```
end
```

Stub received unexpected message

:xyz\_category\_prefix

with (no args)

...

:xyz\_category\_prefix => ?,

...

...

:xyz\_category\_prefix => 'abc',

...



Stub received unexpected message

:kind

with (no args)

...

:kind => ?

...

...

:kind => 'unicorn'

...

Stub received unexpected message

:personal?

with (no args)

...

:personal? => ?

...

...

`:personal? => false`

...

Stub received unexpected message

:id

with (no args)

...

:id => ?

...



...

:id => 1337

...

Stub received unexpected message

:title

with (no args)

...

:title => ?

...

...

:title => 'magic & superglue'

...

## Failures:

### 1) XYZService works

Failure/Error:

subject.should eq("something")

expected: "something"

got: "14abcunicorn\_1337\_3f4894ca\_magicsuper.jpg"

(compared using ==)

# ./xyz\_file\_spec.rb:18:in `block (2 levels) in  
<top (required)>

Failures:

1) XYZService works

Failure/Error:  
subject.should eq("something")

expected: "something"

got: "1/subunit\_04\_3\_894\_main\_screen.jpg"

(compare using ==)

# ./xyz\_file\_spec.rb:18:in `block (2 levels) in  
<top (required)>

**#WIN**

```
let(:target) do
  messages = {
    :publish_on => Date.new(2012, 3, 14),
    :xyz_category_prefix => 'abc',
    :kind => 'unicorn',
    :personal? => false,
    :id => 1337,
    :title => 'magic & superglue'
  }
  stub(:target, messages)
end
```

## Failures:

### 1) XYZService works

Failure/Error:

subject.should eq("something")

expected: "something"

got: "14abcunicorn\_1337\_3f4894ca\_magicsuper.jpg"

(compared using ==)

# ./xyz\_file\_spec.rb:18:in `block (2 levels) in  
<top (required)>



## Failures:

1) XYZService works

Failure/Error:

subject.should eq("something")

expected: "something"

got: "14abcunicorn\_1337\_3f4894ca\_magicsuper.jpg"

(compared using ==)

# ./xyz\_file\_spec.rb:18:in `block (2 levels) in  
<top (required)>

14abcunicorn\_1337\_3f4894ca\_magicsuper.jpg

14abcunicorn  
\_1337\_3f4894ca\_  
magicsuper.jpg

subject.should eq

"14abcunicorn\_1337\_3f4894ca\_magicsuper.jpg"

expected

"14abcunicorn\_1337\_3f4894ca\_magicsuper.jpg"

got

"14abcunicorn\_1337\_59a2b50d\_magicsuper.jpg"

expected

"14abcunicorn\_1337\_3f4894ca\_magicsuper.jpg"

got

"14abcunicorn\_1337\_59a2b50d\_magicsuper.jpg"

subject.should eq

"14abcunicorn\_1337\_**3f4894ca**\_magicsuper.jpg"

subject.should match

/14abcunicorn\_1337\_**[0-9a-f]{8}**\_magicsuper.jpg/



•

Finished in 0.00073 seconds

1 example, 0 failures

•

Finished in 0.0073 seconds

1 example, 1 failures

# #WIN

```
def self.xyz_filename(target)
  # File format:
  # [day of month zero-padded][three-letter prefix] \
  # _[kind]_[age_if_kind_personal]_[target.id] \
  # _[8 random chars]_[10 first chars of title].jpg
  filename = "#{target.publish_on.strftime("%d")}"
  filename << "#{target.xyz_category_prefix}"
  filename << "#{target.kind.gsub("_", "")}"
  filename << "_%03d" % (target.age || 0) if target.personal?
  filename << "_#{target.id.to_s}"
  filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
  truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
  length = truncated_title.length
  truncate_to = length > 9 ? 9 : length
  filename << "_#{truncated_title[0..(truncate_to)]}"
  filename << ".jpg"
  return filename
end
```

```
def self.xyz_filename(target)
  # File format:
  # [day of month zero-padded][three-letter prefix] \
  # _[kind]_[age_if_kind_personal]_[target.id] \
  # _[8 random chars]_[10 first chars of title].jpg
  filename = "#{target.publish_on.strftime("%d")}"
  filename << "#{target.xyz_category_prefix}"
  filename << "#{target.kind.gsub("_", "")}"
  filename << "_%03d" % (target.age || 0) if target.personal?
  filename << "_#{target.id.to_s}"
  filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
  truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
  length = truncated_title.length
  truncate_to = length > 9 ? 9 : length
  filename << "_#{truncated_title[0..(truncate_to)]}"
  filename << ".jpg"
  return filename
end
```

[Approximately  
23 Similar  
scenes have  
been omitted  
for brevity]

Illustration from **The Scariest Story** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/2011/02/scariest-story.html>

```
def self.xyz_filename(target)
  # File format:
  # [day of month zero-padded][three-letter prefix] \
  # _[kind]_[age_if_kind_personal]_[target.id] \
  # _[8 random chars]_[10 first chars of title].jpg
  filename = "#{target.publish_on.strftime("%d")}"
  filename << "#{target.xyz_category_prefix}"
  filename << "#{target.kind.gsub("_", "")}"
  filename << "_%03d" % (target.age || 0) if target.personal?
  filename << "_#{target.id.to_s}"
  filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
  truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
  length = truncated_title.length
  truncate_to = length > 9 ? 9 : length
  filename << "_#{truncated_title[0..(truncate_to)]}"
  filename << ".jpg"
  return filename
end
```

```
target.publish_on.  
    strftime( "%d" )
```

```
:publish_on => Date.new(2012, 3, 14)
```



```
:publish_on => Date.new(2012, 2, 7)
```

```
def self.xyz_filename(target)
  # File format:
  # [day of month zero-padded][three-letter prefix] \
  # _[kind]_[age_if_kind_personal]_[target.id] \
  # _[8 random chars]_[10 first chars of title].jpg
  filename = "#{target.publish_on.strftime("%d")}"
  filename << "#{target.xyz_category_prefix}"
  filename << "#{target.kind.gsub("_", "")}"
  filename << "_%03d" % (target.age || 0) if target.personal?
  filename << "_#{target.id.to_s}"
  filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
  truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
  length = truncated_title.length
  truncate_to = length > 9 ? 9 : length
  filename << "_#{truncated_title[0..(truncate_to)]}"
  filename << ".jpg"
  return filename
end
```

```
target.kind.gsub( "_", "" )
```

:kind => 'unicorn'

:kind => 'magic\_unicorn'

```
def self.xyz_filename(target)
  # File format:
  # [day of month zero-padded][three-letter prefix] \
  # _[kind]_[age_if_kind_personal]_[target.id] \
  # _[8 random chars]_[10 first chars of title].jpg
  filename = "#{target.publish_on.strftime("%d")}"
  filename << "#{target.xyz_category_prefix}"
  filename << "#{target.kind.gsub("_", "")}"
  filename << "_%03d" % (target.age || 0) if target.personal?
  filename << "_#{target.id.to_s}"
  filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
  truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
  length = truncated_title.length
  truncate_to = length > 9 ? 9 : length
  filename << "_#{truncated_title[0..(truncate_to)]}"
  filename << ".jpg"
  return filename
end
```

```
target.title.  
gsub( /^[^a-z\\]/i, '' ).  
  lowercase
```

```
:title => 'magic & superglue'
```



```
:title => 'I <3 Sparkles!!1!'
```

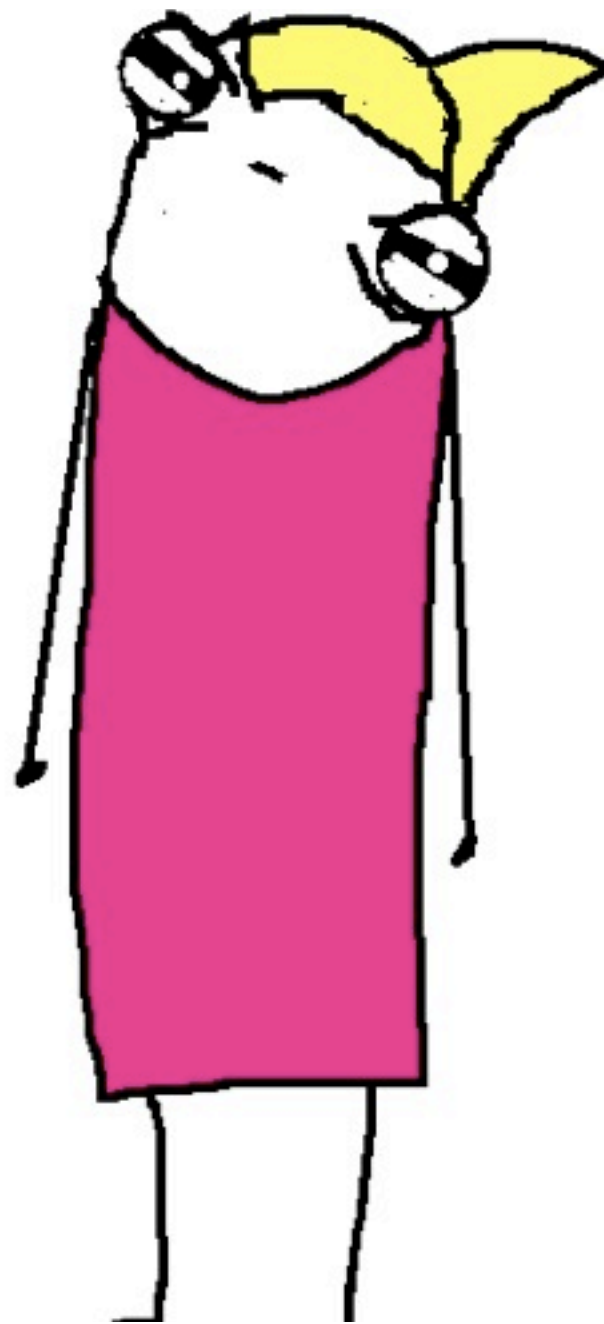


Illustration from **Bycicle** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.no/2010/07/bicycle.html>

```
target.title.  
gsub( /^[^a-z\\]/i, '' ).  
  lowercase
```

```
target.title.  
gsub( /^\[a-z\]/i, ' ' ).  
downcase
```

```
it "works" { ... }
```

```
it "leaves square brackets???"
```

```
it "leaves square brackets???" do
  target.stub(:title => 'i[sparkle]s')
end
```

expected

"07abcmagicunicorn\_1337\_f3b6d325\_**i[sparkle]**.jpg"

to match

/07abcmagicunicorn\_1337\_[0-9a-z]{8}\_**isparkles**.jpg/

```
it "works" { ... }
```

```
it "leaves square brackets" { ... }
```



```
def self.xyz_filename(target)
  # File format:
  # [day of month zero-padded][three-letter prefix] \
  # _[kind]_[age_if_kind_personal]_[target.id] \
  # _[8 random chars]_[10 first chars of title].jpg
  filename = "#{target.publish_on.strftime("%d")}"
  filename << "#{target.xyz_category_prefix}"
  filename << "#{target.kind.gsub("_", "")}"
  filename << "_%03d" % (target.age || 0) if target.personal?
  filename << "_#{target.id.to_s}"
  filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
  truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
  length = truncated_title.length
  truncate_to = length > 9 ? 9 : length
  filename << "_#{truncated_title[0..(truncate_to)]}"
  filename << ".jpg"
  return filename
end
```

```
"_%03d" % (target.age || 0)  
if target.personal?
```

```
"_%03d" % (target.age || 0)  
if target.personal?
```

```
it "works" { ... }  
it "leaves square brackets" { ... }  
it "personalizes"
```

```
it "personalizes" do
  target.stub(:personal? => true)
end
```

Stub received unexpected message

:age

with (no args)

```
it "personalizes" do
  target.stub(:personal? => true)
  target.stub(:age? => 42)
end
```

```
"_%03d" % (target.age || 0)  
if target.personal?
```



```
"_%03d" % (target.age || 0)  
if target.personal?
```

```
it "works" { ... }  
it "leaves square brackets" { ... }  
it "personalizes" { ... }  
it "handles nil age"
```

```
it "handles nil age" do
  target.stub(:personal? => true)
  target.stub(:age? => nil)
end
```

```
def self.xyz_filename(target)
  # File format:
  # [day of month zero-padded][three-letter prefix] \
  # _[kind]_[age_if_kind_personal]_[target.id] \
  # _[8 random chars]_[10 first chars of title].jpg
  filename = "#{target.publish_on.strftime("%d")}"
  filename << "#{target.xyz_category_prefix}"
  filename << "#{target.kind.gsub("_", "")}"
  filename << "_%03d" % (target.age || 0) if target.personal?
  filename << "_#{target.id.to_s}"
  filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
  truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
  length = truncated_title.length
  truncate_to = length > 9 ? 9 : length
  filename << "_#{truncated_title[0..(truncate_to)]}"
  filename << ".jpg"
  return filename
end
```

length > 9 ? 9 : length

```
:title => 'I <3 Sparkles!!1!'
```

```
:title => ' I <3 SPARKLY Sparkles!!1! '
```

```
it "works" { ... }  
it "leaves square brackets" { ... }  
it "personalizes" { ... }  
it "handles nil age" { ... }  
it "handles short titles"
```



```
it "handles short titles" do
  target.stub(:title => 'O HAI')
end
```

.....

Finished in 0.00386 seconds

5 examples, 0 failures

.....

Finished in 0.00386 seconds  
5 examples, 0 failures

#WIN



Illustration from **This is Why I'll Never be an Adult** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/2010/06/this-is-why-ill-never-be-adult.html>



# Refactoring

*(the second middle)*

# Refactoring

*replace method  
with method object*

```
class XYZFile  
end
```



```
class XYZFile
```

```
    def initialize(target)
```

```
    end
```

```
end
```

```
class XYZFile
```

```
    attr_reader :target
```

```
    def initialize(target)
```

```
        @target = target
```

```
    end
```

```
end
```

```
class XYZFile
```

```
  attr_reader :target
```

```
  def initialize(target)
```

```
    @target = target
```

```
  end
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^\[a-z\]]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
class XYZFile
```

```
...
```

```
def self.xyz_filename(target)
```

```
...
```

```
end
```

```
end
```

```
class XYZFile
```

```
...
```

```
def xyz_filename(target)
```

```
...
```

```
end
```

```
end
```

```
class XYZFile
```

```
...
```

```
def name(target)
```

```
...
```

```
end
```

```
end
```

```
class XYZFile
```

```
  ...
```

```
  def name
```

```
    ...
```

```
  end
```

```
end
```

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^\[a-z\]]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```



```
module XYZService  
  
  def self.xyz_filename(target)  
  
  end  
  
end
```

```
require_relative './xyz_file'
```

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
  end
```

```
end
```

```
require_relative './xyz_file'
```

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    XYZFile.new
```

```
  end
```

```
end
```

```
require_relative './xyz_file'
```

```
module XYZService
```

```
  def self.xyz_filename(target)  
    XYZFile.new(target)  
  end
```

```
end
```

```
require_relative './xyz_file'
```

```
module XYZService
```

```
  def self.xyz_filename(target)  
    XYZFile.new(target).name  
  end
```

```
end
```

.....

Finished in 0.00398 seconds

5 examples, 0 failures

.....

Finished in 0.00398 seconds  
5 examples, 0 failure

# #WIN

```
class XYZFile
```

```
...
```

```
def name
```

```
  # File format:
```

```
  # [day of month zero-padded][three-letter prefix] \
```

```
  # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
  # _[8 random chars]_[10 first chars of title].jpg
```

```
  filename = "#{target.publish_on.strftime("%d")}"
```

```
  filename << "#{target.xyz_category_prefix}"
```

```
  filename << "#{target.kind.gsub("_", "")}"
```

```
  filename << "_%03d" % (target.age || 0) if target.personal?
```

```
  filename << "_#{target.id.to_s}"
```

```
  filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
  truncated_title = target.title.gsub(/^[^\[a-z\]]/i, '').downcase
```

```
  length = truncated_title.length
```

```
  truncate_to = length > 9 ? 9 : length
```

```
  filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
  filename << ".jpg"
```

```
  return filename
```

```
end
```

```
end
```



```
class XYZFile
  ...
  def name
    # File format:
    # [day of month zero-padded][three-letter prefix] \
    # _[kind]_[age_if_kind_personal]_[target.id] \
    # _[8 random chars]_[10 first chars of title].jpg
    filename = "#{target.publish_on.strftime("%d")}"
    filename << "#{target.xyz_category_prefix}"
    filename << "#{target.kind.gsub("_", "")}"
    filename << "_%03d" % (target.age || 0) if target.personal?
    filename << "_#{target.id.to_s}"
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
    truncated_title = target.title.gsub(/^[^a-z\ ]+/i, '').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title[0..(truncate_to)]}"
    filename << ".jpg"
    return filename
  end
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
filename = "#{target.publish_on.strftime("%d")}"
```

```
filename << "#{target.xyz_category_prefix}"
```

```
filename << "#{target.kind.gsub("_", "")}"
```

```
filename << "_%03d" % (target.age || 0) if target.personal?
```

```
filename << "_#{target.id.to_s}"
```

```
filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
truncated_title = target.title.gsub(/^[^a-z\ ]+/i, '').downcase
```

```
length = truncated_title.length
```

```
truncate_to = length > 9 ? 9 : length
```

```
filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
filename << ".jpg"
```

```
return filename
```

```
end
```

```
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
filename = "#{target.publish_on.strftime("%d")}"
```

```
filename << "#{target.xyz_category_prefix}"
```

```
filename << "#{target.kind.gsub("_", "")}"
```

```
filename << "_%03d" % (target.age || 0) if target.personal?
```

```
filename << "_#{target.id.to_s}"
```

```
filename << " #{Digest::SHA1.hexdigest(rand(10000).to_s)[0..8]}"
```

```
truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
```

```
length = truncated_title.length
```

```
truncate_to = length > 9 ? 9 : length
```

```
filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
filename << ".jpg"
```

```
return filename
```

```
end
```

```
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
...
```

```
truncated_title = target.title.gsub(/^[^a-z\ ]+/i, '').downcase
```

```
length = truncated_title.length
```

```
truncate_to = length > 9 ? 9 : length
```

```
filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
...
```

```
end
```

```
end
```

```
class XYZFile
  ...
  def name
    ...
    truncated_title = target.title.gsub(/^[^a-z\ ]+/i, '').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title[0..(truncate_to)]}"
    ...
  end
end
```

```
class XYZFile
  ...
  def name
    ...
    truncated_title = target.title.gsub(/^[^a-z\ ]/i, '').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title[0..(truncate_to)]}"
    ...
  end

  def truncated_title
  end

end
```

```
class XYZFile
  ...
  def name
    ...
    truncated_title = target.title.gsub(/^[^a-z\ ]/i, '').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title[0..(truncate_to)]}"
    ...
  end

  def truncated_title
    truncated_title = target.title.gsub(/^[^a-z\ ]/i, '').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title[0..(truncate_to)]}"
  end
end

end
```

```
class XYZFile
  ...
  def name...

  def truncated_title
    truncated_title = target.title.gsub(/^[^a-z\]/i, ' ').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title[0..(truncate_to)]}"
  end
end

end
```



```
class XYZFile
  ...
  def name...

  def truncated_title
    truncated_title = target.title.gsub(/^[^a-z\ ]+/i, ' ').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title[0..(truncate_to)]}"
  end
end
```

```
class XYZFile
  ...
  def name...

  def truncated_title
    truncated_title = target.title.gsub(/^[^a-z\ ]+/i, ' ').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title[0..(truncate_to)]}"
  end
end

end
```

```
class XYZFile
  ...
  def name...

  def truncated_title
    truncated_title = target.title.gsub(/[^\[a-z\]]/i, ' ').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    truncated_title[0..(truncate_to)]
  end
end

end
```

```
class XYZFile
  ...
  def name...

  def truncated_title
    truncated_title = target.title.gsub(/^[^a-z\ ]+/i, ' ').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    truncated_title[0..(truncate_to)]
  end
end
```

```
class XYZFile
  ...
  def name
    ...
    truncated_title = target.title.gsub(/^[^a-z\ ]+/i, '').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title[0..(truncate_to)]}"
    ...
  end

  def truncated_title...

end
```

```
class XYZFile
  ...
  def name
    ...
    truncated_title = target.title.gsub(/[^\[a-z\]]/i, ' ').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title}"
    ...
  end

  def truncated_title...

end
```

```
class XYZFile
  ...
  def name
    ...
    truncated_title = target.title.gsub(/^[^a-z\ ]/i, '').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    filename << "_#{truncated_title}"
    ...
  end

  def truncated_title...
end
```

```
class XYZFile
  ...
  def name
    ...
    filename << "_#{truncated_title}"
    ...
  end

  def truncated_title...

end
```



```
class XYZFile
```

```
...
```

```
def name
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    filename << "_#{truncated_title}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
end
```

```
def truncated_title...
```

```
end
```

```
class XYZFile
```

```
...
```

```
def truncated_title
```

```
    truncated_title = target.title.gsub(/^[^a-z\ ]+/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    truncated_title[0..(truncate_to)]
```

```
end
```

```
end
```

```
class XYZFile
  ...
  def truncated_title
    truncated_title = target.title.gsub(/[^\[a-z\]]/i, '').downcase
    length = truncated_title.length
    truncate_to = length > 9 ? 9 : length
    truncated_title[0..(truncate_to)]
  end
end

end
```

```
target.title.  
gsub( /^[^a-z\\]/i, '' ).  
  lowercase
```

```
target.title.  
gsub( /^\[a-z\] / i, ' ' ).  
downcase
```

```
target.title.  
gsub( /^[^a-z\\]/i, ' ' ).
```

**downcase**

```
target.title.downcase  
gsub(/^[^a-z\\]/, '')
```

```
truncated_title[0..(truncate_to)]
```



```
truncated_title[0..(truncate_to)]
```

```
truncated_title[0..truncate_to]
```

```
truncate_to = length > 9 ? 9 : length  
truncated_title[0..truncate_to]
```

~~truncate\_to = length > 9 ? 9 : length~~

truncated\_title[0..9]

```
length = truncated_title.length  
truncate_to = length > 9 ? 9 : length  
truncated_title[0..9]
```

```
length = truncated_title.length  
truncate_to = length > 9 ? 9 : length  
truncated_title[0..9]
```

```
class XYZFile
  ...
  def truncated_title
    truncated_title = target.title.downcase.gsub(/^[^a-z\\]/, '')
    truncated_title[0..truncate_to]
  end
end

end
```

```
class XYZFile
  ...
  def truncated_title
    truncated_title = target.title.downcase.gsub(/^[^a-z\\]/, '')
    truncated_title[0..truncate_to]
  end
end

end
```



```
class XYZFile
```

```
...
```

```
def truncated_title
```

```
  target.title.downcase.
```

```
  gsub( /[^\[a-z\]]/, '')[0..9]
```

```
end
```

```
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    filename << "_#{truncated_title}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
end
```

```
def truncated_title...
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = "#{target.publish_on.strftime("%d")}"
    filename << "#{target.xyz_category_prefix}"
    filename << "#{target.kind.gsub("_", "")}"
    filename << "_%03d" % (target.age || 0) if target.personal?
    filename << "_#{target.id.to_s}"
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    return filename
  end

  def truncated_title...

end
```

```
class XYZFile
  ...
  def name
    ...
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
    ...
  end

  def truncated_title...

end
```

```
class XYZFile
  ...
  def name
    ...
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
    ...
  end

  def noise
  end

  def truncated_title...

end
```

```
class XYZFile
  ...
  def name
    ...
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
    ...
  end

  def noise
    Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]
  end

  def truncated_title...

end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
...
```

```
filename << "_#{noise}"
```

```
...
```

```
end
```

```
def noise
```

```
  Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]
```

```
end
```

```
def truncated_title...
```

```
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = "#{target.publish_on.strftime("%d")}"
```

```
  filename << "#{target.xyz_category_prefix}"
```

```
  filename << "#{target.kind.gsub("_", "")}"
```

```
  filename << "_%03d" % (target.age || 0) if target.personal?
```

```
  filename << "_#{target.id.to_s}"
```

```
  filename << "_#{noise}"
```

```
  filename << "_#{truncated_title}"
```

```
  filename << ".jpg"
```

```
  return filename
```

```
end
```

```
def noise...
```

```
def truncated_title...
```

```
end
```



```
class XYZFile
  ...
  def name
    filename = "#{target.publish_on.strftime("%d")}"
    filename << "#{target.xyz_category_prefix}"
    filename << "#{target.kind.gsub("_", "")}"
    filename << "_%03d" % (target.age || 0) if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    return filename
  end

  def noise...

  def truncated_title...

end
```

```
class XYZFile
  ...
  def name
    filename = "#{target.publish_on.strftime("%d")}"
    filename << "#{target.xyz_category_prefix}"
    filename << "#{target.kind.gsub("_", "")}"
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id.to_s}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    return filename
  end

  def age
    "%03d" % (target.age || 0)
  end

  def noise...

  def truncated_title...

end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = "#{target.publish_on.strftime("%d")}"
```

```
  filename << "#{target.xyz_category_prefix}"
```

```
  filename << "#{target.kind.gsub("_", "")}"
```

```
  filename << "_#{age}" if target.personal?
```

```
  filename << "_#{target.id.to_s}"
```

```
  filename << "_#{noise}"
```

```
  filename << "_#{truncated_title}"
```

```
  filename << ".jpg"
```

```
  return filename
```

```
end
```

```
def age...
```

```
def noise...
```

```
def truncated_title...
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = "#{target.publish_on.strftime("%d")}"
    filename << "#{target.xyz_category_prefix}"
    filename << "#{target.kind.gsub("_", "")}"
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id.to_s}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    return filename
  end

  def age...

  def noise...

  def truncated_title...

end
```

```
class XYZFile
  ...
  def name
    filename = "#{publication_day}"
    filename << "#{target.xyz_category_prefix}"
    filename << "#{target.kind.gsub("_", "")}"
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id.to_s}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    return filename
  end

  def publication_day
    target.publish_on.strftime("%d")
  end

  def age...

  def noise...

  def truncated_title...

end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = "#{publication_day}"
```

```
  filename << "#{target.xyz_category_prefix}"
```

```
  filename << "#{target.kind.gsub("_", "")}"
```

```
  filename << "_#{age}" if target.personal?
```

```
  filename << "_#{target.id.to_s}"
```

```
  filename << "_#{noise}"
```

```
  filename << "_#{truncated_title}"
```

```
  filename << ".jpg"
```

```
  return filename
```

```
end
```

```
def publication_day...
```

```
def age...
```

```
def noise...
```

```
def truncated_title...
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = "#{publication_day}"
    filename << "#{target.xyz_category_prefix}"
    filename << "#{target.kind.gsub("_", "")}"
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id.to_s}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    return filename
  end

  def publication_day...

  def age...

  def noise...

  def truncated_title...

end
```

```
class XYZFile
  ...
  def name
    filename = "#{publication_day}"
    filename << "#{category}"
    filename << "#{target.kind.gsub("_", "")}"
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id.to_s}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    return filename
  end

  def category
    target.xyz_category_prefix
  end

  def publication_day...

  def age...

  def noise...

  def truncated_title...
```



```
class XYZFile
  ...
  def name
    filename = "#{publication_day}"
    filename << "#{category}"
    filename << "#{target.kind.gsub("_", "")}"
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id.to_s}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    return filename
  end

  def publication_day...

  def category...

  def age...

  def noise...

  def truncated_title...

end
```

```
class XYZFile
  ...
  def name
    filename = "#{publication_day}"
    filename << "#{category}"
    filename << "#{kind}"
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id.to_s}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    return filename
  end

  def kind
    target.kind.gsub( "_", "" )
  end

  def publication_day...

  def category...

  def age...

  def noise...
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = "#{publication_day}"
```

```
  filename << "#{category}"
```

```
  filename << "#{kind}"
```

```
  filename << "_#{age}" if target.personal?
```

```
  filename << "_#{target.id.to_s}"
```

```
  filename << "_#{noise}"
```

```
  filename << "_#{truncated_title}"
```

```
  filename << ".jpg"
```

```
  return filename
```

```
end
```

```
def publication_day...
```

```
def category...
```

```
def kind...
```

```
def age...
```

```
def noise...
```

```
def truncated_title...
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = "#{publication_day}"
    filename << "#{category}"
    filename << "#{kind}"
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id.to_s}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    return filename
  end

  def publication_day...
  def category...
  def kind...
  def age...
  def noise...
  def truncated_title...

end
```

```
filename = "#{publication_day}"
```

```
filename = "#{publication_day}"
```

```
filename = publication_day
```

```
filename << "#{category}"
```



```
filename << category
```

```
filename << "#{kind}"
```

```
filename << kind
```

```
filename << "_#{target.id.to_s}"
```

```
filename << "_#{target.id}"
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = publication_day
```

```
  filename << category
```

```
  filename << kind
```

```
  filename << "_#{age}" if target.personal?
```

```
  filename << "_#{target.id}"
```

```
  filename << "_#{noise}"
```

```
  filename << "_#{truncated_title}"
```

```
  filename << ".jpg"
```

```
  filename
```

```
end
```

```
...
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = publication_day
    filename << category
    filename << kind
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    filename
  end
  ...
end
```

```
class XYZFile
  ...
  def name
    filename = publication_day
    filename << category
    filename << kind
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    filename
  end
  ...
end
```



```
class XYZFile
  ...
  def name
    filename = ""
    filename << publication_day
    filename << category
    filename << kind
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    filename
  end
  ...
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = ""
```

```
  filename << publication_day
```

```
  filename << category
```

```
  filename << kind
```

```
  filename << "_#{age}" if target.personal?
```

```
  filename << "_#{target.id}"
```

```
  filename << "_#{noise}"
```

```
  filename << "_#{truncated_title}"
```

```
  filename << ".jpg"
```

```
  filename
```

```
end
```

```
...
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = ""
    filename << publication_day
    filename << category
    filename << kind
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    filename
  end
  ...
end
```

```
class XYZFile
  ...
  def name
    filename = ""
    filename << publication_day
    filename << category
    filename << kind
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    filename << ".jpg"
    filename
  end
  ...
end
```

```
class XYZFile
  ...
  def name
    filename = ""
    filename << publication_day
    filename << category
    filename << kind
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    "#{filename}.jpg"
  end
  ...
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = ""
```

```
  filename << publication_day
```

```
  filename << category
```

```
  filename << kind
```

```
  filename << "_#{age}" if target.personal?
```

```
  filename << "_#{target.id}"
```

```
  filename << "_#{noise}"
```

```
  filename << "_#{truncated_title}"
```

```
  "#{filename}.jpg"
```

```
end
```

```
...
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = ""
    filename << publication_day
    filename << category
    filename << kind
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    "#{filename}.jpg"
  end
  ...
end
```

```
class XYZFile
  ...
  def name
    filename = ""
    filename << publication_day
    filename << category
    filename << kind
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    "#{filename}.jpg"
  end
  ...
end
```



```
class XYZFile
  ...
  def name
    ...
    filename << publication_day
    filename << category
    filename << kind
    ...
  end
  ...
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
...
```

```
filename << publication_day
```

```
filename << category
```

```
filename << kind
```

```
...
```

```
end
```

```
def prefix
```

```
  [publication_day, category, kind].join
```

```
end
```

```
...
```

```
end
```

```
class XYZFile
```

```
  ...
```

```
  def name
```

```
    ...
```

```
    filename << prefix
```

```
    ...
```

```
  end
```

```
  def prefix
```

```
    [publication_day, category, kind].join
```

```
  end
```

```
  ...
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = ""
    filename << prefix
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    "#{filename}.jpg"
  end
  ...
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = ""
```

```
  filename << prefix
```

```
  filename << "_#{age}" if target.personal?
```

```
  filename << "_#{target.id}"
```

```
  filename << "_#{noise}"
```

```
  filename << "_#{truncated_title}"
```

```
  "#{filename}.jpg"
```

```
end
```

```
...
```

```
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = []
```

```
  filename << prefix
```

```
  filename << "_#{age}" if target.personal?
```

```
  filename << "_#{target.id}"
```

```
  filename << "_#{noise}"
```

```
  filename << "_#{truncated_title}"
```

```
  "#{filename}.jpg"
```

```
end
```

```
...
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = []
    filename << prefix
    filename << "_#{age}" if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    "#{filename}.jpg"
  end
  ...
end
```

```
class XYZFile
  ...
  def name
    filename = ""
    filename << prefix
    filename << age if target.personal?
    filename << "_#{target.id}"
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    "#{filename}.jpg"
  end
  ...
end
```



```
class XYZFile
  ...
  def name
    filename = []
    filename << prefix
    filename << age if target.personal?
    filename << target.id
    filename << "_#{noise}"
    filename << "_#{truncated_title}"
    "#{filename}.jpg"
  end
  ...
end
```

```
class XYZFile
  ...
  def name
    filename = []
    filename << prefix
    filename << age if target.personal?
    filename << target.id
    filename << noise
    filename << "_#{truncated_title}"
    "#{filename}.jpg"
  end
  ...
end
```

```
class XYZFile
  ...
  def name
    filename = []
    filename << prefix
    filename << age if target.personal?
    filename << target.id
    filename << noise
    filename << truncated_title
    "#{filename}.jpg"
  end
  ...
end
```

```
class XYZFile
  ...
  def name
    filename = []
    filename << prefix
    filename << age if target.personal?
    filename << target.id
    filename << noise
    filename << truncated_title
    "#{filename.join("_")}.jpg"
  end
  ...
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = []
```

```
  filename << prefix
```

```
  filename << age if target.personal?
```

```
  filename << target.id
```

```
  filename << noise
```

```
  filename << truncated_title
```

```
  "#{filename.join("_")}.jpg"
```

```
end
```

```
def prefix...
```

```
def publication_day...
```

```
def category...
```

```
def kind...
```

```
def age...
```

```
def noise...
```

```
def truncated_title...
```

```
...
```

```
end
```



Illustration from **About** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.com/p/about.html>

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^\[a-z\]]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = []
    filename << prefix
    filename << age if target.personal?
    filename << target.id
    filename << noise
    filename << truncated_title
    "#{filename.join("_")}.jpg"
  end

  def prefix...
  def publication_day...
  def category...
  def kind...
  def age...
  def noise...
  def truncated_title...
  ...
end
```



```
class XYZFile
```

```
...
```

```
def prefix
```

```
    [publication_day, category, kind].join
```

```
end
```

```
def publication_day
```

```
    target.publish_on.strftime("%d")
```

```
end
```

```
def category
```

```
    target.xyz_category_prefix
```

```
end
```

```
def kind
```

```
    target.kind.gsub("_", "")
```

```
end
```

```
def age
```

```
    "%03d" % (target.age || 0)
```

```
end
```

```
def noise
```

```
    Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]
```

```
end
```

```
def truncated_title
```

```
    target.title.downcase.gsub(/^[^a-z\\]/, '')[0..9]
```

```
end
```

```
end
```

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^\[a-z\]]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```

```
class XYZFile
```

```
...
```

```
def name
```

```
  filename = []
```

```
  filename << prefix
```

```
  filename << age if target.personal?
```

```
  filename << target.id
```

```
  filename << noise
```

```
  filename << truncated_title
```

```
  "#{filename.join("_")}.jpg"
```

```
end
```

```
def prefix...
```

```
def publication_day...
```

```
def category...
```

```
def kind...
```

```
def age...
```

```
def noise...
```

```
def truncated_title...
```

```
...
```

```
end
```

```
module XYZService
```

```
  def self.xyz_filename(target)
```

```
    # File format:
```

```
    # [day of month zero-padded][three-letter prefix] \
```

```
    # _[kind]_[age_if_kind_personal]_[target.id] \
```

```
    # _[8 random chars]_[10 first chars of title].jpg
```

```
    filename = "#{target.publish_on.strftime("%d")}"
```

```
    filename << "#{target.xyz_category_prefix}"
```

```
    filename << "#{target.kind.gsub("_", "")}"
```

```
    filename << "_%03d" % (target.age || 0) if target.personal?
```

```
    filename << "_#{target.id.to_s}"
```

```
    filename << "_#{Digest::SHA1.hexdigest(rand(10000).to_s)[0,8]}"
```

```
    truncated_title = target.title.gsub(/^[^\[a-z\]]/i, '').downcase
```

```
    length = truncated_title.length
```

```
    truncate_to = length > 9 ? 9 : length
```

```
    filename << "_#{truncated_title[0..(truncate_to)]}"
```

```
    filename << ".jpg"
```

```
    return filename
```

```
  end
```

```
end
```

```
class XYZFile
  ...
  def name
    filename = []
    filename << prefix
    filename << age if target.personal?
    filename << target.id
    filename << noise
    filename << truncated_title
    "#{filename.join("_")}.jpg"
  end

  def prefix...
  def publication_day...
  def category...
  def kind...
  def age...
  def noise...
  def truncated_title...
  ...
end
```



# Codejunk

*(the ending)*

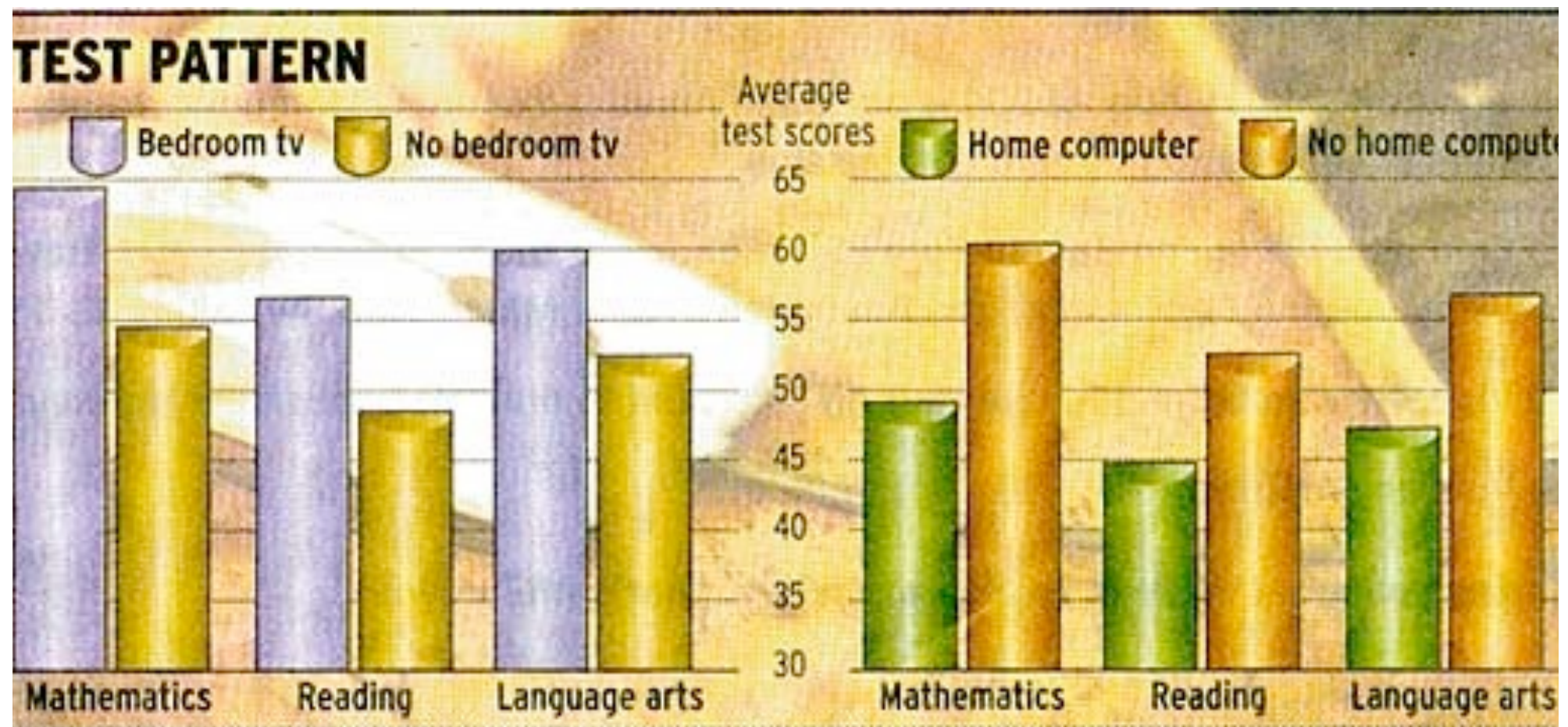


Illustration from **Chartjunk** by **Viveka Weiley**  
<http://chartjunk.karmanaut.com/?p=8>



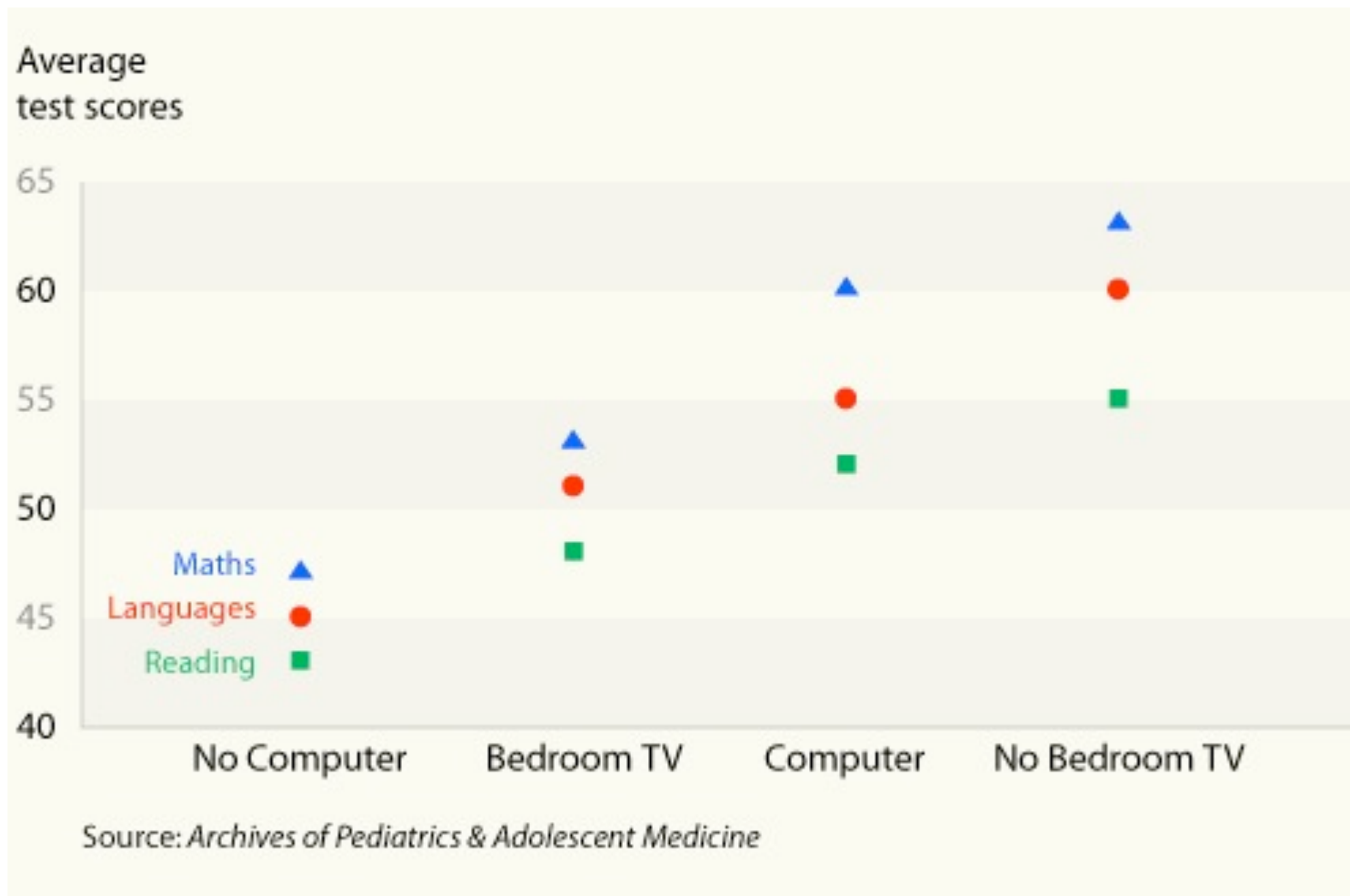


Illustration from **Chartjunk** by **Viveka Weiley**  
<http://chartjunk.karmanaut.com/test-pattern-retuned/>

# Top 10

*(in no particular order)*

# #10

*Lousy Comments*

```
module Codejunk
```

```
  # States the obvious
```

```
  def state_the_obvious
```

```
    "whatever"
```

```
  end
```

```
end
```

```
module Codejunk
```

```
  # Takes modulus 100
```

```
  def say_it_again(number)
```

```
    number % 100
```

```
  end
```

```
end
```

```
module Codejunk
```

```
  # Subtracts
```

```
  def incorrect_comment
```

```
    1 + 1
```

```
  end
```

```
end
```

```
module Codejunk
```

```
  # add
```

```
  def fuzzy_comment
```

```
    1 + 1
```

```
  end
```

```
end
```

```
module Codejunk
```

```
  # ad too numbers
```

```
  def bad_comment
```

```
    1 + 2
```

```
  end
```

```
end
```



# #9

*Trailing Whitespace*

```
module Codejunk.....
```

```
..
```

```
def a_method.
```

```
  'Use your words'.....
```

```
end...
```

```
..
```

```
.....
```

```
end
```

# #8

*Commented-Out Code*

```
module Codejunk
```

```
    def tweaked_logic(wday)
```

```
        # ((7 - wday) % 7) * 1.day
```

```
        (7 - wday) % 7
```

```
    end
```

```
end
```

```
module Codejunk
```

```
  def tweaked_logic(wday)
```

```
    if false
```

```
      ((7 - wday) % 7) * 1.day
```

```
    end
```

```
      (7 - wday) % 7
```

```
  end
```

```
end
```

#7

*Needless Parentheses*

```
module Codejunk
```

```
    def spurious_parentheses ( )
```

```
        junk ( )
```

```
    end
```

```
end
```

```
module Codejunk
```

```
  def more_spurious_parentheses  
    (0..(one_method_call)).to_a  
  end
```

```
  def one_method_call  
    (10 - rand(10))  
  end
```

```
end
```



# #6

*Powerless Code*

```
module Codejunk
```

```
  def spurious_arguments
```

```
    [1, 2, 3].join('')
```

```
  end
```

```
end
```

```
module Codejunk
```

```
  def spurious_arguments
```

```
    [1, 2, 3].join
```

```
  end
```

```
end
```

```
class ObjectWrapper  
  
  attr_accessor :object  
  def initialize(object)  
    @object = object  
  end  
  
end
```

```
module Codejunk
```

```
  def extract(response)
```

```
    if (200..299).include?(response.code.to_i)
```

```
      response.body if response
```

```
    end
```

```
  end
```

```
end
```

```
module Codejunk
```

```
  def extract(response)
```

```
    if (200..299).include?(response.code.to_i)
```

```
      response.body if response
```

```
    end
```

```
  end
```

```
end
```

```
module Codejunk
```

```
  def extract(response)
```

```
    if (200..299).include?(response.code.to_i)
```

```
      response.body if response
```

```
    end
```

```
  end
```

```
end
```

# #5

*Unnecessary Requires*



```
require 'active_support/all'
```

```
# Nothing uses active support
```

```
module Codejunk
```

```
  . . .
```

```
end
```

# #4

*The booleanest boolean*

```
module Codejunk
```

```
  def truth(code)
```

```
    code == 13 ? true : false
```

```
  end
```

```
end
```

```
module Codejunk
```

```
  def lie!
```

```
    !true
```

```
  end
```

```
end
```

# #3

*Too Much Hard Work*

```
module Codejunk
```

```
  def spurious_string_interpolation
```

```
    "#{thing}"
```

```
  end
```

```
  def thing
```

```
    "a string"
```

```
  end
```

```
end
```

```
module Codejunk
```

```
  def spurious_stringification  
    "I am #{age.to_s} years old"  
  end
```

```
  def age  
    rand(25) + 15  
  end
```

```
end
```

```
module Codejunk
```

```
  def spurious_transformations
```

```
    [1, 2, 3].map(&:to_s).join
```

```
  end
```

```
end
```



```
module Codejunk
```

```
  def spurious_complexity
```

```
    s = rand(2) == 0 ? "abcde" : "abc"
```

```
    cutoff = s.length > 4 ? 4 : s.length
```

```
    s[0..cutoff]
```

```
  end
```

```
end
```

```
module Codejunk
```

```
  def spurious_hard_work
```

```
    s = "I <3 Magic!"
```

```
    s.gsub( /^[^a-z]/i, ' ' ).downcase
```

```
  end
```

```
end
```

# #2

## *Duplicated Tests*

```
module Codejunk
  def add(n1, n2)
    n1 + n2
  end
end

describe Codejunk do
  subject { stub.extend(Codejunk) }

  it "adds" do
    subject.add(1, 1).should eq(2)
  end

  it "adds twin primes" do
    subject.add(29, 31).should eq(60)
  end

  it "adds big numbers" do
    subject.add(9238498, 2039238).should eq(11277736)
  end

  it "adds negative numbers" do
    subject.add(100, -390).should eq(-290)
  end
end
```

```
describe Codejunk do
```

```
  subject { stub.extend(Codejunk) }
```

```
  it "gets rid of spaces" do
```

```
    subject.sanitize(" o m g ").should eq('omg')
```

```
  end
```

```
  it "gets rid of funky characters" do
```

```
    subject.sanitize("omg^%#={?_").should eq('omg')
```

```
  end
```

```
  it "gets rid of numbers" do
```

```
    subject.sanitize("omg123").should eq('omg')
```

```
  end
```

```
  it "downcases everything" do
```

```
    subject.sanitize("OMG").should eq('omg')
```

```
  end
```

```
end
```

# #1

*Combine All The Codejunk*

```
module Codejunk
```

```
  def junk
```

```
    # Subtracts
```

```
    s = rand(2) == 0 ? "abcde" : "abc"
```

```
    cutoff = s.length > 4 ? 4 : s.length
```

```
    s = s[0..cutoff]
```

```
    s << "#{spurious_string_interpolation()}"
```

```
    s << "I am #{age.to_s} years old, and "
```

```
    s << "I <3 Magic!".gsub(/^[a-z]/i, '').downcase
```

```
    s << "#{(0..(one_method_call)).map(&:to_s).join(' ')}"
```

```
    return s
```

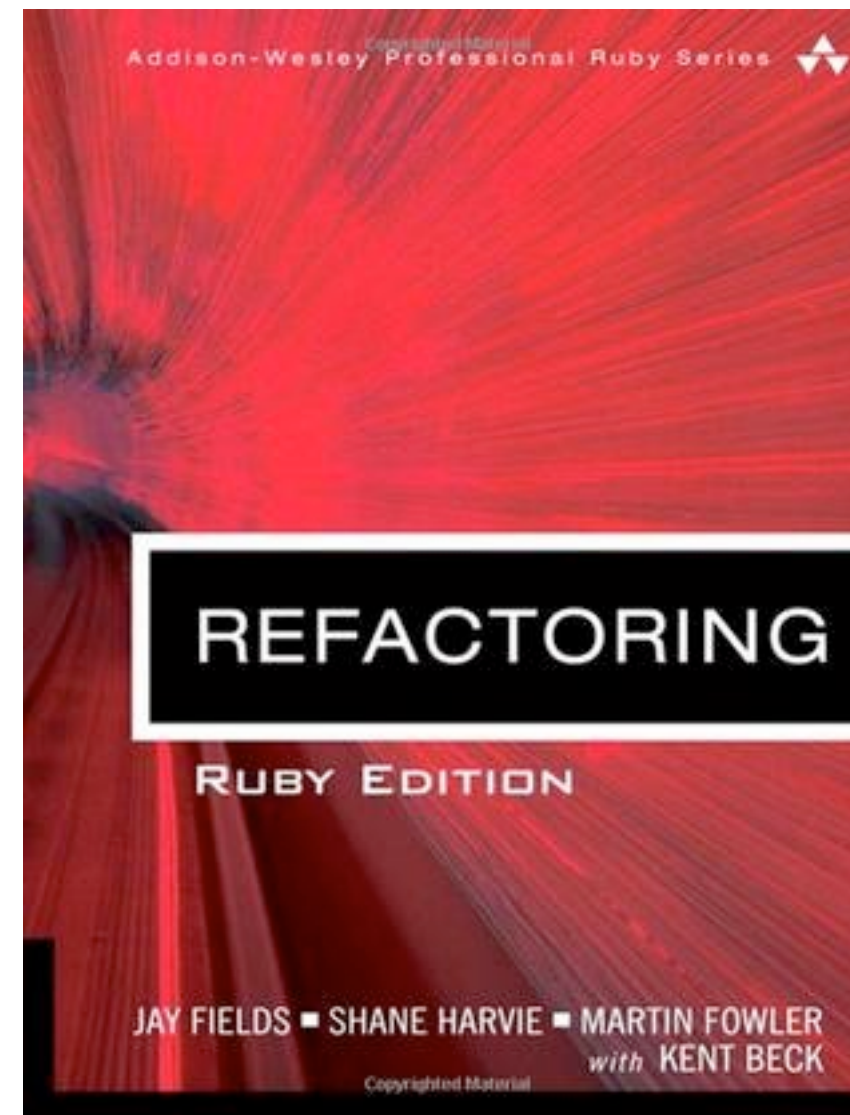
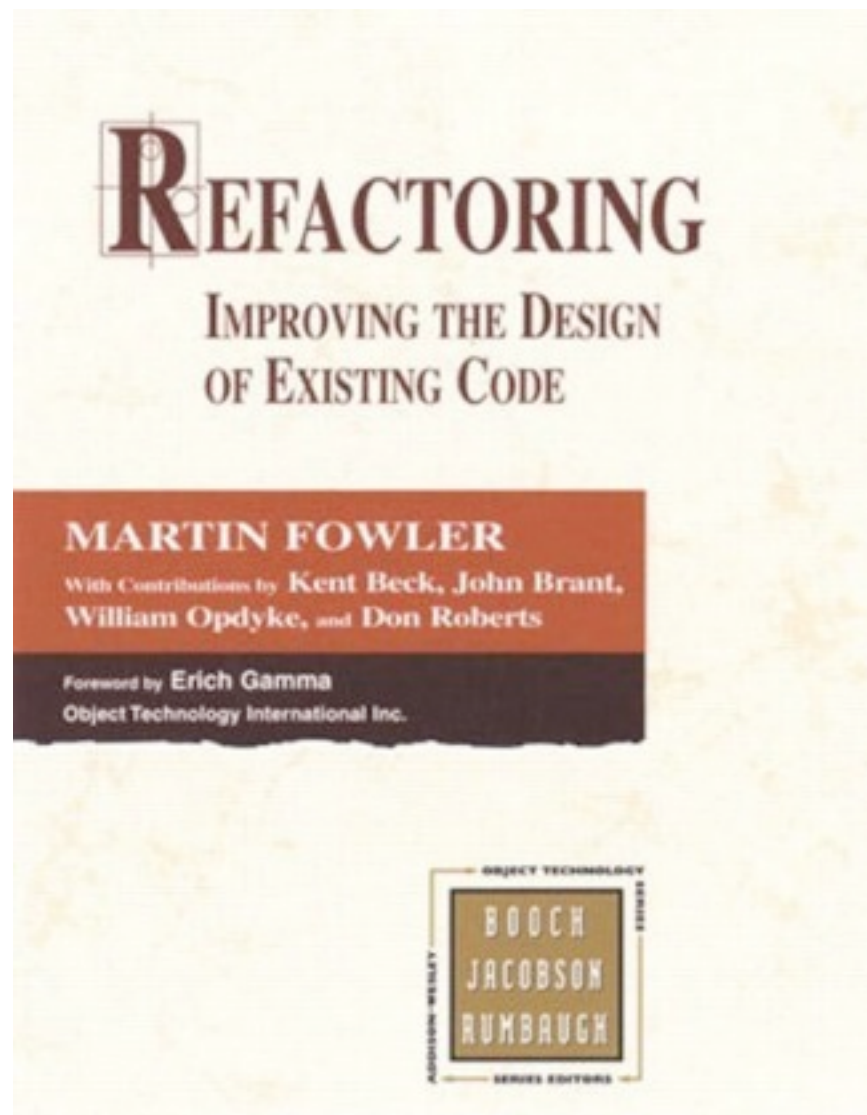
```
  end
```

```
end
```



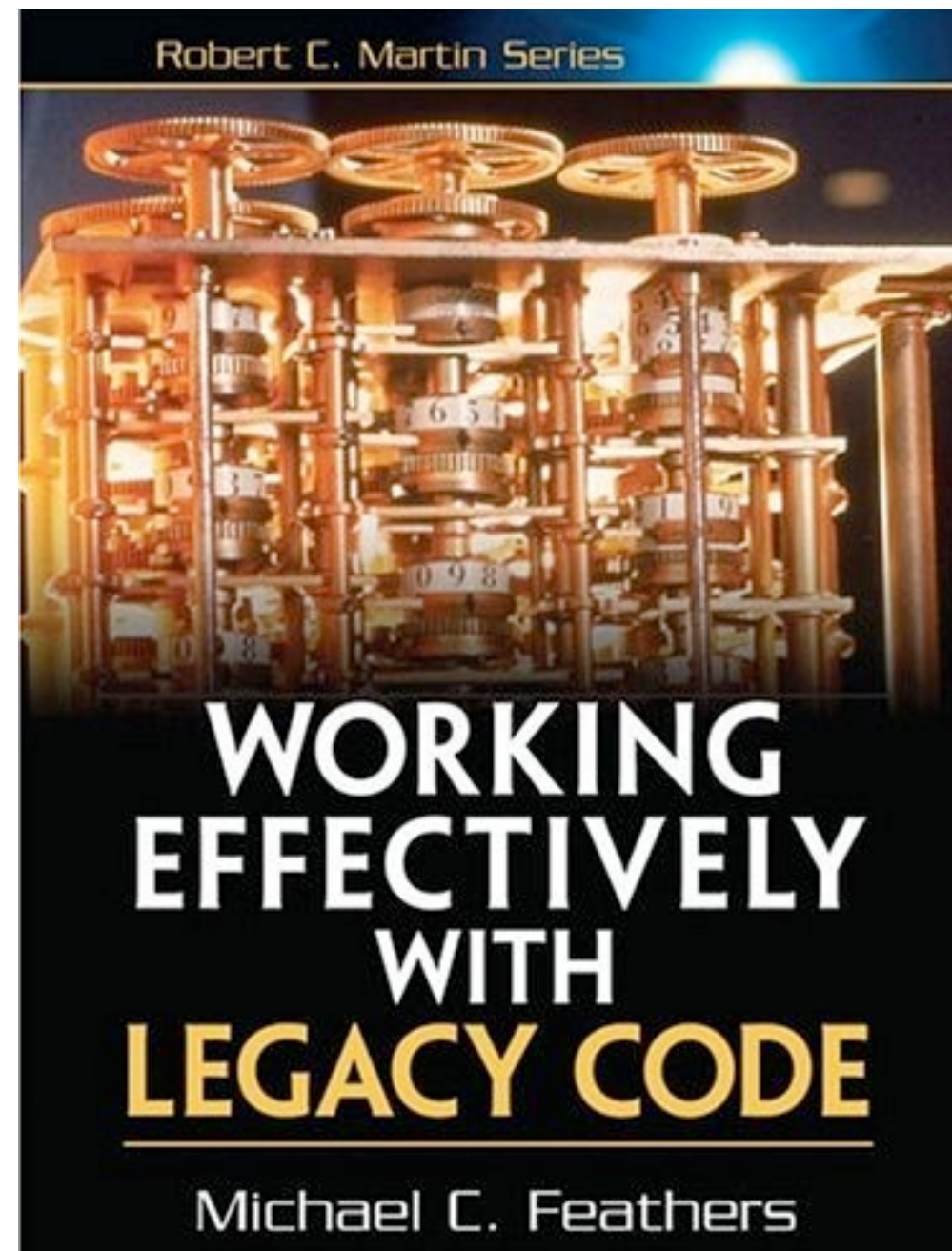


***github.com/kytrinyx  
/therapeutic-refactoring***

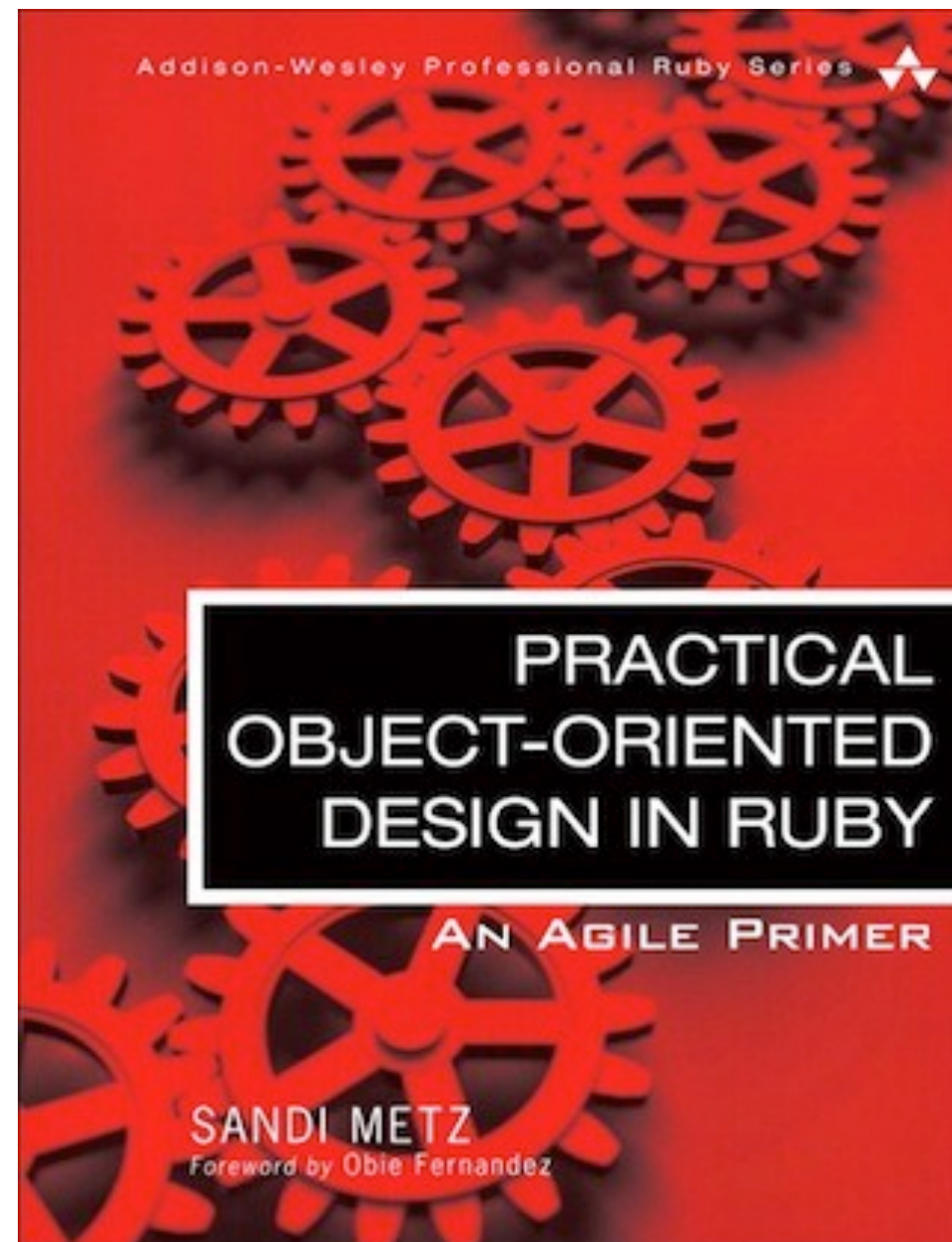


**Refactoring** by **Martin Fowler** *et al.*  
<http://www.goodreads.com/book/show/44936.Refactoring>

**Refactoring, Ruby Edition** by **Jay Fields** *et al.*  
<http://www.goodreads.com/book/show/11560939-refactoring>



**Working Effectively With Legacy Code** by **Michael Feathers**  
[http://www.goodreads.com/book/show/44919.Working\\_Effectively\\_with\\_Legacy\\_Code](http://www.goodreads.com/book/show/44919.Working_Effectively_with_Legacy_Code)



**Practical Object-Oriented Design in Ruby** by **Sandi Metz**  
<http://www.goodreads.com/book/show/13507787-practical-object-oriented-design-in-ruby>

# Therapy

*(the moral)*



I 

refactoring



# Science!

## *Working Memory*

\* Beilock & DeCaro, Department of Psychology, University of Chicago  
<http://www.ncbi.nlm.nih.gov/pubmed/17983308>



# Science!

*Problem-Solving Strategies*

# Science!

*Worry & Panic*

# Refactoring

*makes you smarter!*

# Exobrain

# Exobrain

*counteracts panic*

# Refactoring

*makes you happier!*

# Fast Tests

# Fast Tests

*are awesome*



# Happiness

# Happiness

*leads to good design*



Illustration from **God of Cake** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.no/2010/10/god-of-cake.html>

# exobrain



Illustration from **God of Cake** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.no/2010/10/god-of-cake.html>

# prevents panic



Illustration from **God of Cake** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.no/2010/10/god-of-cake.html>

# happiness



Illustration from **God of Cake** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.no/2010/10/god-of-cake.html>

# decoupled code



Illustration from **God of Cake** by **Allie Brosh**  
<http://hyperboleandahalf.blogspot.no/2010/10/god-of-cake.html>

# KTHXBYE

**Katrina Owen**

@kytrinyx

\_[kytrinyx.com](mailto:kytrinyx@kytrinyx.com)

[github.com/kytrinyx/therapeutic-refactoring](https://github.com/kytrinyx/therapeutic-refactoring)





**JumpstartLab**

**Katrina Owen**

@kytrinyx

\_[kytrinyx.com](mailto:kytrinyx@kytrinyx.com)

[github.com/kytrinyx/therapeutic-refactoring](https://github.com/kytrinyx/therapeutic-refactoring)