

DETAILSEITE ARTIKEL

BILD

BLOG
EINTRAG

KOMMENTAR

...

STARTSEITE

BLOG
EINTRAG

BLOG
EINTRAG

BLOG
EINTRAG

NAVI

NEUES KOMMENTAR ANLEGEN

AUTOR

TITEL

TEXT

OK

NEUEN ARTIKEL ANLEGEN

TITEL

TEXT

OK

SUCHE

LOGIN

EINEN ANTRAG
ANSEHEN

NEUEN EINTRAG
ANSEHEN

ZURÜCK

KOMMENTIEREN

ZURÜCK

Standard Assertions

assert	boolean
assert_equal	expected, actual
assert_raise	*args
assert_raises	*args, &block
assert_instance_of	klass, object
assert_nil	object
assert_kind_of	klass, object
assert_respond_to	object, method
assert_match	pattern, string
assert_same	expected, actual
assert_operator	object1, operator, object2
assert_nothing_raised	*args
assert_not_same	expected, actual
assert_not_equal	expected, actual
assert_not_nil	object
assert_no_match	regexp, string
assert_throws	expected_symbol, &proc
assert_nothing_thrown	&proc
assert_in_delta	expected_float, actual_float, delta
assert_send	send_array

Rails Assertions

assert_response	type
assert_redirected_to	options = {}
assert_template	expected
assert_recognizes	expected_options, path, extras={}
assert_generates	expected_path, options, defaults={}, extras = {}
assert_routing	path, options, defaults={}, extras={}
assert_tag	*opts
assert_no_tag	*opts
assert_dom_equal	expected, actual
assert_dom_not_equal	expected, actual
assert_valid	record

Test::Rails Assertions

assert_assigned	ivar, value = NOTHING
• deny_assigned	
assert_content_type	type, message = nil
assert_flash	key, content
assert_image	src
assert_error_on	field, type
assert_field	form_action, type, model, column, value = nil
assert_input	form_action, type, name, value = nil
assert_label	form_action, name, include_f = true
assert_links_to	href, content = nil
• deny_links_to	
assert_multipart_form	form_action
assert_post_form	form_action
assert_select	form_action, model, column, options
assert_submit	form_action, value
assert_tag_in_form	form_action, options
• deny	
assert_empty	obj
• deny_empty	
assert_includes	obj, item, message = nil
• deny_includes	

Most also take a message argument as the last parameter. The message will be shown if the test fails.



Methods

Strings

capitalize!
center
chomp!
chop!
concat
count
crypt
delete!
downcase!
dump
each
each_byte
empty?
gsub!
hash
hex
include?
index
intern
length
ljust, rjust
next!
oct
replace
reverse!
rindex
scan
slice!
split
squeeze!
strip!
sub!
sum
swapcase!
tr!
tr_s!
unpack
upcase!
upto

Regex

escape
last_match
new
quote
casefold?
kcode
match
source

Time

asctime
ctime
day
gmt?
gmtime
hour
isdst
localtime
mday
min
mon
month
sec
strftime
tv_sec
tv_usec
usec
utc
utc?
wday
yday
year
zone

DEFAULT DIRECTORY STRUCTURE

```
rails_root
├── app
│   ├── apis
│   ├── controllers
│   │   └── application.rb
│   ├── helpers
│   │   └── application_helper.rb
│   ├── models
│   ├── views
│   └── layouts
├── components
├── config
│   ├── environments
│   │   ├── development.rb
│   │   ├── production.rb
│   │   └── test.rb
│   ├── database.yml
│   ├── environment.rb
│   └── routes.rb
├── db
├── doc
├── lib
├── log
│   ├── development.log
│   ├── production.log
│   ├── server.log
│   └── test.log
├── public
│   ├── images
│   ├── javascripts
│   │   ├── controls.js
│   │   ├── dragdrop.js
│   │   ├── effects.js
│   │   └── prototype.js
│   ├── stylesheets
│   ├── .htaccess
│   ├── 404.html
│   ├── 500.html
│   ├── dispatch.cgi
│   ├── dispatch.fcgi
│   ├── dispatch.rb
│   ├── favicon.ico
│   └── index.html
├── script
├── test
│   ├── fixtures
│   ├── functional
│   ├── mocks
│   │   └── development
│   └── test
├── unit
│   └── test_helper.rb
└── vendor
```

METHODS NOTE

! - Denotes where a trailing ! may be used. A colourless ! denotes that the ! is compulsory.

PRE-DEFINED VARIABLES

\$!	Exception information
\$&	String of last match
\$`	String left of last match
\$'	String right of last match
\$+	Last group of last match
\$N	Nth group of last match
\$=	Case insensitive flag
\$/	Input record separator
\$\	Output record separator
\$_	Output field separator
\$.	Current line number of last file read
\$>	Default output for print
\$_	Last input line of string
\$0	Name of script
\$*	Command line arguments
\$stderr	Standard error output
\$stdin	Standard input
\$stdout	Standard output
\$-a	True if -a is set.
\$-d	Status of -d switch
\$-l	True if -l is set
\$-p	True if -p is set
\$-v	Verbose Flag

RESERVED WORDS

=begin	elsif	rescue
=end	end	retry
BEGIN	ensure	return
END	false	self
alias	for	super
and	if	then
begin	in	true
break	module	undef
case	next	unless
class	nil	until
def	not	when
defined?	or	while
do	redo	yield
else		

REGULAR EXPRESSIONS SYNTAX

^	Start of string
\$	End of string
.	Any single character
(a b)	a or b
(...)	Group section
[abc]	Item in range (a or b or c)
[^abc]	Not in range (not a or b or c)
a?	Zero or one of a
a*	Zero or more of a
a+	One or more of a
a{3}	Exactly 3 of a
a{3,}	3 or more of a
a{3,6}	Between 3 and 6 of a
!(pattern)	"Not" prefix. Apply rule when URL does not match pattern.

Methods

Arrays

assoc
at
clear
collect!
compact!
concat
delete
delete_at
delete_if
each
each_index
empty?
eq!?
fill
first
flatten!
include?
index
indexes
join
last
length
nitems
pack
pop
push
rassoc
reject!
replace
reverse!
reverse_each
rindex
shift
slice!
sort!
uniq!
unshift

Validation

condition_block?
create!
evaluate_condition
validate
validate_on_create
validate_on_update
validates_acceptance_of
validates_associated
validates_confirmation_of
validates_each
validates_exclusion_of
validates_format_of
validates_inclusion_of
validates_length_of
validates_numericality_of
validates_presence_of
validates_size_of
validates_uniqueness_of

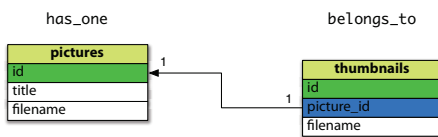
Enumerable Mixin

collect
each_with_index
entries
find
find_all
grep
include?
max
min
reject
sort

Available free from
ILoveJackDaniels.com

Ruby on Rails Logo
used with permission.

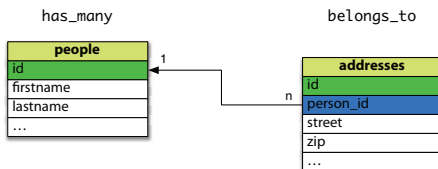
has_one



```
class Picture < ActiveRecord::Base
  has_one :thumbnail
end

class Thumbnail < ActiveRecord::Base
  belongs_to :picture
end
```

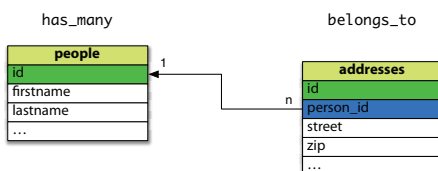
belongs_to



```
class Picture < ActiveRecord::Base
  has_one :thumbnail
end

class Thumbnail < ActiveRecord::Base
  belongs_to :picture
end
```

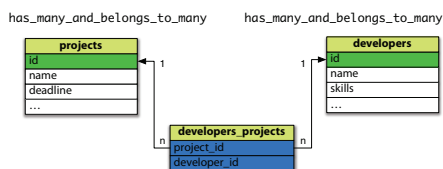
has_many



```
class Person < ActiveRecord::Base
  has_many :addresses
end

class Address < ActiveRecord::Base
  belongs_to :person
end
```

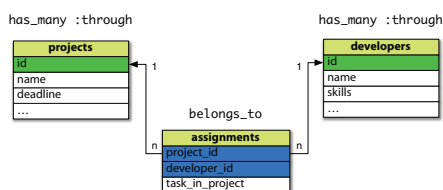
has_many_and_belongs_to



```
class Project < ActiveRecord::Base
  has_and_belongs_to_many :developers
end

class Developer < ActiveRecord::Base
  has_and_belongs_to_many :projects
end
```

has_many :through



```
class Project < ActiveRecord::Base
  has_many :developers, :through => assignments
end

class Assignment < ActiveRecord::Base
  belongs_to :developer
  belongs_to :project
end

class Developer < ActiveRecord::Base
  has_many :projects, :through => assignments
end
```


ERROR: undefined
OFFENDING COMMAND: eexec

STACK:

/quit
-dictionary-
-mark-