

5 Ways to set Attributes in ActiveRecord

Posted on [June 28, 2011](#) by [David](#)

Rails 3 allows the developer to change ActiveRecord attributes in various ways. Each one does it slightly differently with sometimes unique side-effects. It's important you understand which method to use, so here's a short list (*and a neat cheat table at the end!*)

`user.name = "Rob"`

This regular assignment is the most common and easiest to use. It is the default write accessor generated by Rails. The **name** attribute will be marked as dirty and the change will not be sent to the database yet.

You can undo the change by calling **reload!** or save the change to the database by calling **save**.

`user.write_attribute(:name, "Rob")`

This is the method that is called by the default accessor above. A synonym for this function is **user[:name] = "Rob"**. It also has a **read_attribute** counterpart.

Just like above, this method does not yet change the attribute in the database. Use this method anywhere you need to bypass the default write accessor above, for example when you want to write a custom **attribute=** writer.

`user.update_attribute(:name, "Rob")`

This method will change the attribute in the model and pass it straight to the database, without running any validations.

Two gotchas:

- Any *other* changed attributes are also saved to the database.
- Validations are skipped so you could end up with invalid data.

Because of that last quirk it's a good practice to use **update_attributes** instead.

`user.attributes = {:name => "Rob"}`

This method will set all the attributes you pass it, except those who are protected from mass assignment if you're using **attr_protected** or **attr_accessible**. The changes are not saved to the database.

You can override the mass assignment protection by passing false:

```
user.send(:attributes=, {:name => "Rob"}, false)
```

`user.update_attributes(:name => "Rob")`

This method changes the attributes of the model, checks the validations, and updates the record in the database if it validates. Since it uses the above **attributes=** method, attributes protected from mass assignment are not changed.

Note that just like **update_attribute** this method also saves other changed attributes to the database.

Handy Cheat Sheet Table

Method	Uses Default Accessor	Mass Assignment Protection	Saved to Database	Validations
attribute=	Yes	No	No	n/a
write_attribute	No	No	No	n/a
update_attribute	Yes	No	Yes	No
attributes=	Yes	Yes ((Mass Assignment Protection for attributes= is overridable.))	No	n/a
update_attributes	Yes	Yes	Yes	Yes

If you want to understand more about these methods I suggest you check out their source code. Each time it's only a couple of lines and it will really broaden your understanding of how Rails works!

This entry was posted in [Ruby on Rails](#). Bookmark the [permalink](#).

Comments (42)

Login

Sort by: [Date](#) [Rating](#) [Last Activity](#)



[Today download](#) · 91 weeks ago

-12

Excellent post. I want to thank you for this informative read, I really appreciate sharing this great post. Keep up your work. Thanks for this very useful info you have provided us.

Reply



[Today Download](#) · 81 weeks ago

-15

Thank you for making this awesome and wonderful works of yours!!keep it up!

Reply



Martin Streicer · 74 weeks ago

+2

Recent changes in Rails 3 altered the behavior of update_attribute to run validations. Use update_column to save to the database without validations.

Reply [1 reply](#) · active 43 weeks ago

 Eric · 71 weeks ago -3

Thanks for answering my question!!!

Reply [1 reply](#) · active 1 week ago

 [dennis](#) · 57 weeks ago -2


Enjoyed the post! Thanks for the tips!

Reply

 tom · 54 weeks ago -2

this is great!

Reply

 Mido · 45 weeks ago -2

Enjoyed this post. Very helpful. Thanks!

Reply

 [go here](#) · 45 weeks ago -5

To set active records in attributes is not that a big task. But there are certain methods that you need to follow. The five ways that you have shown here is perfect but I strictly recommend that there are other methods too.

Reply

 [maven-repository](#) · 43 weeks ago -1

Thanks for this handy overview

Reply

 [stretch](#) · 42 weeks ago -3

she's terrible and loud, don't like her all all on the show!

Reply

 [Emily](#) · 42 weeks ago 0

The article is written in a very good manner.

Reply

 [Judy](#) · 42 weeks ago -7

The article is written in a very good manner. It help me in my college project <http://www.feload.com>

Reply

 [Jasmine](#) · 42 weeks ago -5

Hey, just wanted to know if this site is in the UK? <http://www.buyraiderz.com/Product/GOLD>

Reply

 [Aimee](#) · 42 weeks ago -4

Very interesting! Really very nice blog you :) <http://www.weddgown.com>

Reply

 [Christina](#) · 42 weeks ago -7

The Photo Expo sounds like a great time! : D <http://www.feload.com>

Reply

 <http://garson.info> · 38 weeks ago -3

Thank you for publishing 5 Ways to set Attributes in ActiveRecord | David Verhasselt, I just actually had been looking for anything very similar and was happy to discover the tips through this particular content.

Reply



[derek feng](#) · 29 weeks ago

0

excellent post. very basic stuff that needs to be understood.

Reply



Nolan · 27 weeks ago

0

Dig it!

Reply



[Kama](#) · 23 weeks ago

-1

This is my first visit to your site. Today I have got a new site. I am really happy to be here. I am stay tuned here for your next blog...

Reply



[Laura](#) · 23 weeks ago

0

I enjoyed your entries on Toxic Words - such great thoughts and a wonderful reminder to watch the words I use - to be positive and kind and use

words to build up rather than tear down. :)

Reply



[Madge](#) · 23 weeks ago

-1

Your website is really cool and this is a great inspiring article.

Reply



[Mabel](#) · 23 weeks ago

-2

Thanks for your marvelous posting! I actually enjoyed reading it, you will be a great author. I will ensure that I bookmark your blog and will

come back in the foreseeable future. I want to encourage that you continue your great job, have a nice weekend!

Reply



[Donald Saelinger Law](#) · 22 weeks ago

0

The worth of these tips will make me always remember this post, thanks for sharing it.

Reply



[folgen hier](#) · 20 weeks ago

-2

This was such a well structured tutorial, very good insight into the application for getting these amazing results.

Reply



david · 18 weeks ago

-2

Enjoyed the post! Thanks for the tips!

[fundraising options on long island](#)

Reply

Post a new comment

Enter text right here!

Comment as a Guest, or login:

Name

Displayed next to your comments.

Email

Not displayed publicly.

Website (optional)

If you have a website, link to it here.

Subscribe to

None ▼

Submit Comment

David Verhasselt

Proudly powered by WordPress.