



Log in (/login?redirect=https%3A%2F%2Fwww.codecademy.com%2Fen%2Fforum_questions%2F534dbd327c82ca9c53000440)

Sign up (/register?redirect=https%3A%2F%2Fwww.codecademy.com%2Fen%2Fforum_questions%2F534dbd327c82ca9c53000440)

This forum is now read-only. Please use our new forums at discuss.codecademy.com (/discourse_redirect/ruby-beginner-en-Qn7Qw).

Course Forum (/forums/ruby-beginner-en-Qn7Qw) / Section 3 (/forums/ruby-beginner-en-Qn7Qw/2)

/ Exercise 3.5 (/forums/ruby-beginner-en-Qn7Qw/2/exercises/4) / How to include multiple values on a key? And th...

Hashes and Symbols Forum

View Course (/courses/ruby-beginner-en-Qn7Qw/0/1) » View Exercise (/courses/ruby-beginner-en-Qn7Qw/2/5)

498 points



Submitted by

Brad (/4f8ed09c70698a00030322ee)

almost 3 years ago

How to include multiple values on a key? And then iterate?

The lesson has conveyed that keys can have multiple values.

1) How would multiple values be created, for instance, in this exercise? (e.g. If I wanted to add a description of the movie using a string along with the numerical rating, how would I be able to do that?)

2) And then once multiple values are assigned to a key, how can you iterate over those values? For instance, how could I put out just the movie description or just the rating?

For reference, here's what I would like to be able to write in order to do this. (Granted, I know it doesn't work at all...)

```
movie_ratings = {
  memento: 3, "ok",
  primer: 3.5, "decent",
  the_matrix: 3, "ok",
  truman_show: 4, "good",
  red_dawn: 1.5, "meh",
  skyfall: 4, "good",
  alex_cross: 2, "not great",
  uhf: 1, "awful",
  lion_king: 3.5, "decent"
}

movie_ratings.each_value {|value1, value2| puts value2}
```

5 votes

permalink

The lesson has conveyed that keys can have multiple values.

If the exercise text suggests that anywhere, it's certainly incorrect. Each key can only have one value. But the same value can occur more than once inside a Hash, while each key can occur only once. So `{:a => 1, :b => 1}` is a valid Ruby Hash, but `{:a => 1, :a => 2}` isn't (well, you can actually **type** that latter Hash in, but then the value for the key `:a` will be `2`, that is, the first value gets overwritten).

That being said, **any kind of object** can be the value of a key in a Ruby Hash. So for storing lists of things, you can store Arrays:

```
holiday_expenses = {
  food: [23.99, 4.35, 8.95],
  travel: [134.95, 12.45],
  accommodation: [413.17]
}
```

In this fashion you could store the rating as the first member of your Array, the verbal description as the second member:

```
movie_ratings = {
  memento: [3, "ok"],
  # ...
}
```

Then to access them you'd have to say

```
puts movie_ratings[:memento][1]    #=> ok
```

You can also store Hashes in a Hash:

```
movies = {  
  skyfall: {  
    year: 2012,  
    oscars: 2,  
    starring: ["Daniel Craig", "Judi Dench"]  
  }  
}
```

and then access the single bits of information by name:

```
movies[:skyfall][:starring].first    #=> "Daniel Craig"
```

Or you can define a custom class to represent a movie, then create instances of that class, and store those instances in a Hash. The possibilities are limitless.

3114 points



Submitted by

Alex J (/4f182941e22479000100d360)

almost 3 years ago

1 Comment



gorchov (/52d764fa282ae3809e00049e) almost 3 years ago

nice answer :)