

Ruby master - Feature #17155

Add a Diggable mixin

09/05/2020 02:05 PM - janfri (Jan Friedrich)

Status:	Open
Priority:	Normal
Assignee:	
Target version:	

Description

Ruby 2.3 introduced the `#dig` method as a general way to access elements in deep structures. Initially it was implemented only for Array and Hash. In the meanwhile also the classes Struct, OpenStruct and Set have a `#dig` method. There is also a [proposal](#) to implement `#dig` for Thread.

In my development I have own classes which have a `#[]` method implemented and are also candidates for a `#dig` method. But implementing `#dig` isn't so trivial as it seems: you have to handle some corner cases. This is the intention for my proposal: Add a Diggable mixin to Ruby (core or stdlib). A simple implementation of it could be something like this:

```
module Diggable

  def dig *args
    a = args.shift
    o = self[a]
    return o if args.empty? || o.nil?
    o.send :dig, *args
  end

end
```

So you could simply include Diggable to any class which implements `#[]`, and you would get a `#dig` method for free. Like Comparable or Enumerable for which you have to implement `#<=>` respectively `#each` and get other methods implemented in this mixins.

History

#1 - 09/05/2020 02:08 PM - janfri (Jan Friedrich)

- Description updated