

Learning Ruby

Ruby Arrays and Lists

1

Enumerated Type – Ruby doesn't have them

But there are cool workarounds

2

Implementation of Arrays

- Ruby's Arrays are backed by actual C arrays. When you first declare a basic array (`[]` or `Array.new`) an instance of the `RArray` struct is created, and it points to a C array which is allocated with a size of 16.
- When we need more space– we increase the size by half again of the needed size and copy

3

```
a = []  
100_000.times do  
  |i| a << 'shenanigans!' ← adds to end of array  
end
```

Only reallocates 22 times!!!

You can waste time and space, but using

`Array.new(n)` solves the problem.

Reallocations are really pretty fast as mem copy

4

Implementation

- Push/pop are $O(1)$ operations (as are appending and that is fast)
- unshift/shift are $O(n)$ operations (as moving the whole array)

5

Learn Ruby Conventions for capitalization and naming

- ClassNames
- method_names and variable_names
- methods_asking_a_question?
- changing_parameter_methods!
- @instance_variables (in a class)
- \$global_variables
- SOME_CONSTANTS or OtherConstants

6