

Quick Assignment Review

Chapter 2 and 3 - Ruby Basics



Task #3 Equations

1..puts $4 * (30 + 30 + 30 + 30 + 28) + 1$

2..puts $(4 * 28) + 1 + (4 * 30) * 4$

3..puts $(4 * 30) * 4 + (28 * 3) + 29$

4..puts $(365.25 * 4) - ((31.0 * 7) * 4)$

Variables

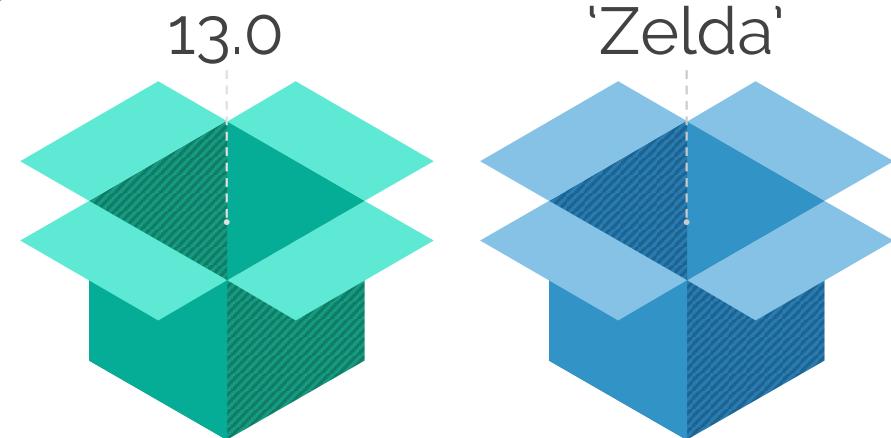
Chapter 4 - Ruby Basics

Variables

Containers to hold values or pointers to objects in memory

Variables are used to store intermediate values so a program can reuse them or store results from number or string arithmetic.

NOTE: variable names can be pretty much any sequence of letters and numbers, BUT MUST START with a lowercase letter



Variable Examples

```
my_variable = 13  
puts my_variable
```

```
my_variable = 13 + 10  
puts my_variable
```

```
my_variable = 80  
my_variable = my_variable + 80  
puts my_variable
```

Mixing it Up

Chapter 4 - Ruby Basics



Converting

1.puts '13.0'.to_f = 13.0

2.puts '13.0'.to_i = 13

3.puts '3 is the magic number'.to_i = 3

4.puts 'Today is 420 day'.to_f = 0.0



User Input

The `gets` command allows you to capture data from the user, and your data is only as good as the user that enters it. In the future we will have to protect the program from bad users.

Variable Examples

```
my_value = gets  
puts my_value  
  
my_new_value = gets.chomp  
pus my_new_value
```

The chomp method cleans up any carriage return characters at the end of the captured string



Commenting

Ruby Basics



Variable Examples

The pound sign is used to tell the computer, ignore the rest of this line

```
# Here is a comment  
  
# I needed to divide robs shots / number of goals to figure out how effect he is  
# He usually isn't very effective  
puts robs_shots / robs_goals
```

If we name our variables correctly and write logical code, we shouldn't need too many comments throughout our code. Clean code reads without comments.

Comments should be used for information not inherent in the code.



Any Questions ???

