Introduction to Programming - 42

Day 03

Kai kai@42.us.org
Gaetan gaetan@42.us.org

Summary: This document is the subject of the day 03 of the introduction to programming piscine.

Contents

Chapter I

Guidelines

- Corrections will take place in the last hour of the day. Each person will correct another person accoring to the peer-corrections model.
- Questions? Ask the neighbor on your right. Next, ask the neighbor on your left.
- Read the examples carefully. The exercises might require things that are not specified in the subject...
- Your reference manual is called Google / "Read the Manual!" / the Internet / ...

Chapter II

Preamble

Chapter III

Exercise 00 : My First Method

- Create a script `my_first_method.rb` which includes a method. The method takes a string as an argument. The method must return an uppercase version of the string, if and only if the character string is longer than 10 characters. If the string is 10 characters or less, the method returns nil.

- Your program will call this method and display the return value on the command line. If there are no parameters, display `none` followed by a newline.

```
?> ./my_first_method.rb | cat -e
none$
?> ./my_first_method.rb "eheh" | cat -e
?> ./my_first_method.rb "eheh" | cat -e
HELLO WORLD$
I'M HAPPY TO BE HERE$
?>
```

`Google ruby methods.`

Chapter IV

Exercise 01 : Greetings for All

- Create a script `greetings_for_all.rb` that contains a greetings method. The method takes a parameter name and displays a welcome message with that name. If the method is called without arguments, its default setting will be "noble stranger". If the method is called with an arguent that is not a string, an error message should be displayed instead of the welcome message.

- So the following script:

```
?> cat greetings_for_all.rb
# your method definition here

greetings lucie
greetings
greetings 22
?>
```

will have the result:

```
?> ./greetings_for_all.rb | cat -e
Hello, lucie.$
Hello, noble stranger.$
Error! That doesn't sound like a name.$
?>
```

Google is_a.

Chapter V

Exercise 02 : Help your Professor

- Create a script `help_your_professor.rb` which contains a method average_mark. The method will use a hash, associating the first name of the students with their grade, to calculate the average score of the class on that test.

- So the following script:

```
?> cat help_your_professor.rb
# your method definition here$

class_csci101 = {
    "margot" => 18,
    "june" => 8,
    "colin" => 15,
    "lewis" => 9
}
class_csci102 = {
    "quentin" => 17,
    "julie" => 15,
    "mark" => 8,
    "stephanie" => 13
}
puts "Average mark for the CSCI 101 class: #{average_mark class_csci101}."
puts "Average mark for the CSCI 102 class: #{average_mark class_csci102}."
?>
```

has the result:

```
?> /.help_your_professor.rb | cat -e
Average mark for the CSCI 101 class: 12.$
Average mark for the CSCI 102 class: 13.$
```

`Google ruby hashes`

Chapter VI

Exercise 03 : Family Affairs

- Create a script `family_affairs.rb`. It will contain a find_the_gingers method which takes in as a parameter a hash containing the first names of family members as key and their hair colors as attribute. This method will use the select metho to collect the first names of the redheads in a new array, which it will return.

- So a script like this:

```
?> ./family_affairs.rb | cat -e
# your method definition here

Dupont_family = {
    "matthew" => :red,
    "mary" => :blonde,
    "virginia" => :brown,
    "gaetan" => :red,
    "fred" => :red,
}

p find_the_gingers Dupont_family
?>
```

would have this result:

```
?> ./family_affairs.rb | cat -e
["matthew", "gaetan", "fred"]$
?>
```

`Google ruby hashes, select, to_a`

Chapter VII

Exercise 04 : Persons of Interest

- Create a script `persons_of_interest.rb`. It will contain a method `great_births` that takes a hash representing people from history, each entry itself being a hash with keys "name" and "year_of_birth". Display them in order sorted by birth dates.

- A script like this:

```
?> cat persons_of_interest.rb
# your method definition here

women_in_science = {
    :ada => { :name => "Ada Lovelace", :year_of_birth => "1815" },
    :cecilia => { :name => "Cecila Payne", :year_of_birth => "1900" },
    :lise => { :name => "Lise Meitner", :year_of_birth => "1878" },
    :grace => { :name => "Grace Hopper", :year_of_birth => "1906" }
}

great_births women_in_science
```

  has output like this:

```
?> ./persons_of_interest.rb | cat -e
Ada Lovelace is a great person born in 1815.$
Lise Meitner is a great person born in 1878.$
Cecila Payne is a great person born in 1900.$
Grace Hopper is a great person born in 1906.$
?>
```

`Google ruby hashes, sort_by.`