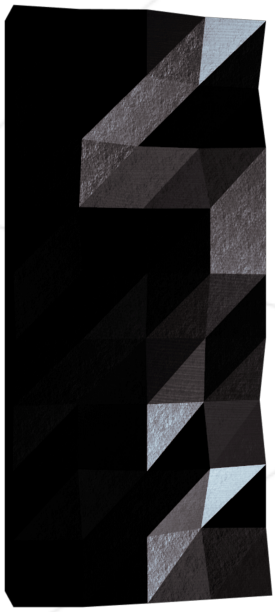

Introduction to Programming - 42

Day 01

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

Summary: This document is the subject of the day 01 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

Here is what `Wikipédia` has to say about the otter:

`Otters are animals that live near and around water. They are a part of the animal family Mustelid. A group of otters is called a "romp", because they play together and are energetic. They live in nests called holts. There are many different kinds of otters. Some live near rivers, some in the sea (Sea Otters). Otters live on every continent except for Australia and Antarctica.`

`Otters are carnivores; they must eat a lot of meat to live, so they spend 3 to 5 hours every day fishing and hunting. They can die of hunger more quickly than most animals. They eat fish, crayfish, crabs, and frogs. They will eat any small animals and birds they can catch. Much of their food is found in water. They dive in rivers, lakes and streams until they can find a suitable animal to eat, which they then chase. Once they catch it, they bring it to the top of the water, where they eat it. If the animal has a hard shell, otters can use a rock as a tool to break open the shell.`

`Otters are playful and energetic. They can be seen sliding down hills and slopes, as well as chasing other otters for fun. Some kinds of otters live in groups, while others are almost always alone. Because they spent so much time in cold water, they have to groom their fur often to stop themselves from freezing.`

`Otters talk to other animals and otters with whistles, growls, chuckles and screams, as well as chirps, squeals, and some kinds of otters might even purr. They also leave their scent on plants to mark their territory.`

`Sea otters (Enhydra lutris) are classified as marine mammals. They live along the Pacific coast of North America. Their historic range included shallow waters of the Bering Strait. It also included Kamchatka, and as far south as Japan. They have a rich fur for which humans hunted them almost to extinction. They frequently carry a rock in a pouch under their forearm. They use this to smash open shells. This makes them one of the few animals that use tools. Although once near extinction, they have begun to spread again, from remnant populations in California and Alaska.`

It's a cute otter.

Chapter III

Exercise 00 : 42

	Exercise 00
How to display 42	
Turn-in directory : $ex00/$	
Files to turn in : `42.rb`	
Allowed functions : `All`	
Notes : `n/a`	

- Create a script `42.rb` that, when executed, displays "42" followed by a newline.

```
?> ./42.rb | cat -e
42$
?>
```

Does it sound simple? That's normal.

Ruby is a language meant to read like plain English - easier for us!

Chapter IV

Exercise 01 : name

	Exercise 01
Display a name	
Turn-in directory : $ex01/$	
Files to turn in : `name.rb`	
Allowed functions : `All`	
Notes : `n/a`	

- Create a script `name.rb` in which you define a first_name variable and a last_name variable, initialized with your first and last names respectively, and then display them followed by a newline.

```
?> ./name.rb | cat -e
Arthur Dent$
?>
```

Chapter V

Exercise 02 : name++

	Exercise 02
Display a name, better	
Turn-in directory : $ex02/$	
Files to turn in : `name.rb`	
Allowed functions : `All`	
Notes : `n/a`	

- Modify the script name.rb: still do the same thing, but this time your first and last name should be concatenated and assigned to a third variable.

```
?> ./name.rb | cat -e
Ford Prefect$
?>
```

`Google ruby strings.`

Chapter VI

Exercise 03 : what's your name

	Exercise 03
What's your name?	
Turn-in directory : $ex03/$	
Files to turn in : `whatsyourname.rb`	
Allowed functions : `All`	
Notes : `n/a`	

- Create a script whatsyourname.rb first asks the user to enter their first name, then their last name, and finally displays both.

```
?> ./whatsyourname.rb
Hey, what's your first name ? : Arthur
And your last name ? : Dent
Well, pleased to meet you Arthur Dent.
?>
```

`Google gets, chomp.`

Chapter VII

Exercise 04 : disp_first_param

	Exercise 04
Display a parameter	
Turn-in directory : *ex*04/	
Files to turn in : `disp_first_param.rb`	
Allowed functions : `All`	
Notes : `n/a`	

- Create a script `disp_first_param.rb` which, when executed, displays the first string passed as a parameter, followed by a newline. If there are no parameters, display `none` followed by a newline.

```
?> ./disp_first_param.rb | cat -e
none$
?> ./disp_first_param.rb "Beeblebrox" "Improbability" "Slartibartfast" | cat -e
Beeblebrox$
?>
```

`Google ARGV, array, condition if.`

Chapter VIII

Exercise 05 : age

	Exercise 05
Receive and modify a number	
Turn-in directory : $ex05/$	
Files to turn in : `age.rb`	
Allowed functions : `All`	
Notes : `n/a`	

- Create a script `age.rb` that asks the user to enter their age, and then displays how old the user will be in 10 years, 20 years, and 30 years.

```
?> ./age.rb
Please tell me your age : 15
You are currently 15 years old.
In 10 years, you'll be 25 years old.
In 20 years, you'll be 35 years old.
In 30 years, you'll be 45 years old.
?>
```

`Google string to_i.`

Chapter IX

Exercise 06 : UPCASE_IT

	Exercise 06
Show in all caps	
Turn-in directory : *ex*06/	
Files to turn in : `upcase_it.rb`	
Allowed functions : `All`	
Notes : `n/a`	

- Create a script `upcase_it.rb` which takes a character string as a parameter. When executed the script displays the string in all caps followed by a newline. If the number of parameters is different from 1, display `none` followed by a newline.

```
?> ./upcase_it.rb | cat -e
none$
?> ./upcase_it.rb "don't panic" | cat -e
DON'T PANIC$
?> ./upcase_it.rb 'tHiS iS sO eAsY! - rUbY iS bAe' | cat -e
THIS IS SO EASY! - RUBY IS BAE$
?>
```

Google upcase.

Chapter X

Exercise 07 : downcase_it

	Exercise 07
Show in lowercase	
Turn-in directory : *ex*07/	
Files to turn in : `downcase_it.rb`	
Allowed functions : `All`	
Notes : `n/a`	

- Create a script `downcase_it.rb` which takes a character string as a parameter. When executed, the script displays the string in lowercase followed by a newline. If the number of parameters is different from 1, display `none` followed by a newline.

```
?> ./downcase_it.rb | cat -e
none$
?> ./downcase_it.rb "TRILLIAN" | cat -e
trillian$
?> ./downcase_it.rb 'tHiS iS sO eAsY! - rUbY iS bAe' | cat -e
this is so easy! - ruby is bae$
?>
```

Try without asking for hints.

Chapter XI

Exercise 08 : scan_it

	Exercise 08
Scan the text	
Turn-in directory : *ex*08/	
Files to turn in : `scan_it.rb`	
Allowed functions : `All`	
Notes : `n/a`	

- Create a script `scan_it.rb` that takes two parameters. The first is a keyword to look for in a string. The second is the string to search. When executed, the program displays the number of occurrences of the keyword in the string. If the number of parameters is different from 2 or the first string does not appear in the second string, then display `none` followed by a newline.

```
?> ./scan_it.rb | cat -e
none$
?> ./scan_it.rb "the" | cat -e
none$
?> ./scan_it.rb "the" "these exercises in day01 are not the hardest ones we'll see \=P" | cat -
 e
2$
?>
```

Try "string.scan(string)".