



Data Management & Visualization Exercises

Alexis Bogroff

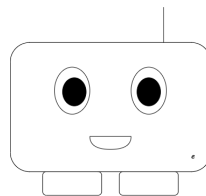
Data Management - Part I

Core concepts of Series and DataFrames to train on:

- create a Series / DataFrame
- delete a Series / DataFrame
- add an element
- delete an element
- retrieve an element
- modify an element
- same operations with multiple elements

Introduction

In this series of exercises we are going to follow the adventures of a young robot endowed with capacity to feel emotions named "Eoki". Eoki can learn fast, but is also a sensible HR (Human-Robot), so please be gentle!



Eoki loves animals, but its favorites are pandas! You are going to help Eoki find where its pandas friends are located, ensure they are in a good shape and in a safe environment with enough healthy food.

1.1 - Import & Explore

Eoki heard in the news that it was possible to transport animals on large distances by communicating through an API (Application Programming Interface) with a cloud platform specialized in autonomous drones. Yes, Eoki too has started to learn Python since it enables infinite possibilities in its world!

Eoki would like to protect the remaining pandas from Earth by importing the pandas in its country, the last country without frequent wars on Earth. You can help Eoki by running a single line of Python to import pandas to its country.

Q1: Import the pandas package. For convinience, we would like to use an alias for this package, let's use 'pd' like the whole Python community

In [2]:

```
## ----- YOUR CODE HERE ----- ##
```

It seems that a lot of pandas have been imported! Let's check that they properly landed

Q2: Verify by any mean that the package has been correctly imported

Hint :If the import failed, it should have throw an error, or the methods should not be working.

```
In [3]: ## ----- YOUR CODE HERE ----- ##
```

Even though it is not yet possible for Eoki to see them, some information were transmitted by the cloud platform about each pandas:

- age
- size
- weight

Q3: Create a pandas Series for each variable (age, size, weight):

- age: 2, 5, 1, 12, 4
- size: 100, 160, 60, 170, 180
- weight: 12, 120, 70, 130, 130

```
In [4]: ## ----- YOUR CODE HERE ----- ##
```

Eoki would like to know these pandas before it can see them. Thus, Eoki is diving into the data that have been transfered in order to mentally visualize these nice pandas. Also, it would like to give them names. I told you, Eoki has feelings :)

Q4: Rename the index of each series with the following names (in that order):

- foo
- bar
- lazy
- greedy
- kwargs

```
In [5]: ## ----- YOUR CODE HERE ----- ##
```

Woow, these are strange names. Eoki is a robot in the end, please don't judge it.

Let's try to understand why Eoki chose those names:

- it seems for the first and second that Eoki lacked some inspiration. Indeed, in the Python community, tutorials and reference documentations usually use *foo* and *bar* as variable names when explaining a code snippet (little portion of code).
- lazy often refers to programs that do the minimum of work.
- greedy are the opposite of lazy programs, they do all the possible work, a kind of overzealous worker.
- kwargs, well, let's talk about that directly with your tutors

Q5: Display the name, age, size and weight of the first pandas in the series

Expected output: The first pandas 'foo' is 2 old, 20 tall and weights 40

```
In [6]: ## ----- YOUR CODE HERE ----- ##
```

According to [some articles](#) this pandas seems in the norm. No doubt Eoki named it *foo* because it is standard!

Q6: Display the name, age, size and weight of the second pandas in the series

Expected: The second pandas 'bar' is 5 year(s) old, 160 tall and weights 120

```
In [7]: ## ----- YOUR CODE HERE ----- ##
```

Q7: Display the name, age, size and weight of the second pandas in the series

Expected: The third pandas 'lazy' is 1 year(s) old, 60 tall and weights 70

```
In [8]: ## ----- YOUR CODE HERE ----- ##
```

Wooooow! This is not a job for lazy robots, doing it over again for each pandas is quite boring, nor is it beautiful. Isn't everything about beauty in the end?

Q8: Display the name, age, size and weight for all the pandas! Try to use as few code words as possible

Expected:

The pandas n°0 called 'foo' is 2 year(s) old, 20 tall and weights 40

The pandas n°1 called 'bar' is 5 year(s) old, 160 tall and weights 120

The pandas n°2 called 'lazy' is 1 year(s) old, 60 tall and weights 70

The pandas n°3 called 'greedy' is 12 year(s) old, 170 tall and weights 130

The pandas n°4 called 'kwargs' is 4 year(s) old, 180 tall and weights 130

```
In [12]: ## ----- YOUR CODE HERE ----- ##
```

Great! That's quicker to implement and to modify, it is scalable to million of pandas. We Data Scientists and coders like these principles! (it is also a curse, beware not to over-optimize every bit of code 🐱🐼🐶)

Still, don't you feel that should be even simpler? (am I in the danger zone of over-optimization? Not this time)

Q9: Create a pandas DataFrame that contains the series aforementioned. The index of this DataFrame should be the name of the pandas. You can name this variable 'saved_pandas'

```
In [11]: ## ----- YOUR CODE HERE ----- ##
```

Now that you have a better view of your data, can you tell why 'lazy' might be a lazy pandas?

Q10: Analyze your data and provide a plausible explanation

----- YOUR ANSWER HERE -----

Eoki still needs you to know better this group of newly arrived pandas. Can you help it?

Q11: Compute and display the average age, size and weight

```
In [13]: ## ----- YOUR CODE HERE ----- ##
```

Q12: Compute and display the maximum age, size and weight

```
In [14]: ## ----- YOUR CODE HERE ----- ##
```

Q13: Compute and display: the minimum, maximum, mean of these factors in a single line of code

Hint: you can find help in the [pandas API reference](#) (even it can be scary at first). Look for 'Computations / descriptive stats'.

```
In [15]: ## ----- YOUR CODE HERE ----- ##
```

Q14: Compute and display the standard deviation of these factors

```
In [16]: ## ----- YOUR CODE HERE ----- ##
```

Let's produce a cleaner output. Do we really need a precision of 6 digits (4.324350)? Don't think so. Let's reduce that to a precision of 2 digits (4.32)

Q15: Compute and display the same standard deviation with a precision of 2

```
In [17]: ## ----- YOUR CODE HERE ----- ##
```

Hummmm nice! Eoki is very grateful for your help.

Oh no! It seems however that Eoki is disturbed by something. A pandas seems in a very bad shape. Eoki is too disturbed to even give you the name of this pandas. Which pandas is it, have you seen it while you were producing this descriptive analysis? What is the problem?

Q16: Find out what could make Eoki so sad

Hint: is a pandas about to die? How could one know about that without a life spent studying pandas?

----- YOUR ANSWER HERE -----

If you didn't see it, it is quite normal. That is another curse of Data Analysts / Scientists, well, mostly at the beginning. You were so eager to help, you deep dived into the code, and forgot about the big picture. It is important to frequently take a step back and think about the coherence of our code outputs.