

# Writing faster Python

*Tuesday 31 May 2016 17:00 (30 minutes)*

Did you know that Python preallocates integers from -5 to 257? Reusing them 1000 times, instead of allocating memory for a bigger integer, can save you a couple of milliseconds of code's execution time. If you want to learn more about this kind of optimizations then, ...well, probably this presentation is not for you :) Instead of going into such small details, I will talk about more "sane" ideas for writing faster code. After a very brief overview of how to optimize Python code (rule 1: don't do this; rule 2: don't do this yet; rule 3: ok, but what if I really want to do this?), I will show simple and fast ways of measuring the execution time and finally, discuss examples of how some code structures could be improved. You will see:

- What is the fastest way of removing duplicates from a list
- How much faster your code is when you reuse the built-in functions instead of trying to reinvent the wheel
- What is faster than the good ol' for loop
- If the lookup is faster in a list or a set (and when it makes sense to use each)
- How the "It's better to beg for forgiveness than to ask for permission" rule works in practice

I will NOT go into details of "serious" optimization, like using different Python implementation or rewriting critical code in C, etc.

## Talk Length

30 minutes

**Primary author:** WITOWSKI, Sebastian (CERN)

**Presenter:** WITOWSKI, Sebastian (CERN)