PH36X **Guidelines on how to present your code**

For most activities in the PH36X series we end the class with presentations. Doing a good presentation is not trivial, and quite frankly what is "good" is a matter of opinion and there are therefore multiple ways of doing a "good job". In addition, a code that can be presented well is most likely well written and could be easily (or less difficulty) understood by someone who hasn't coded it. Finally, especially starting from PH366, one of the learning objectives of the class is to use the codes to foster a deeper understanding of the physics involved. How do we do that?

This set of guidelines is supposed to be a living body of suggestions that can be edited, updated, commented upon to grow into a comprehensive guide.

1. Make sure your code is divided in jupyter blocks that are also logical blocks. For example:
    a. A block for importing libraries
    b. A block for defining and assigning values to parameters, both physical and computational (e.g., the energy of a system, the number of x values used to tabulate a function, the time step *dt*)
    c. A block for defining and assigning values to constants (e.g., gravity constant, Planck constant)
    d. A block for each python function
    e. A block (or more) for calculations
    f. A block (or more) for plotting
2. When presenting, address every block, even if shortly
3. For function and calculation blocks, describe both the semantic and the syntax. Also address any difficulties and/or roadblocks you faced, if relevant
4. Always run your code in front of the audience, do not just show the output of a run you carried out beforehand
5. If your code is not quite working as expected, describe the strange behavior, and demonstrate it. Repeat for a few different choices of the parameters.
6. If your code produces a warning, address it
7. If your code produces an error and stops executing, address it and discuss what you did to attempt troubleshooting
8. If your code works as expected (or at least you believe so), discuss how you can use it to better understand the science or the coding. For example:
    a. Demonstrate how the behavior of the physical system changes by changing physical parameters
    b. Demonstrate how the solution changes by increasing/decreasing resolution
    c. Change the initial conditions of your system
    d. Show how execution time changes by using different algorithms
    e. etcetera