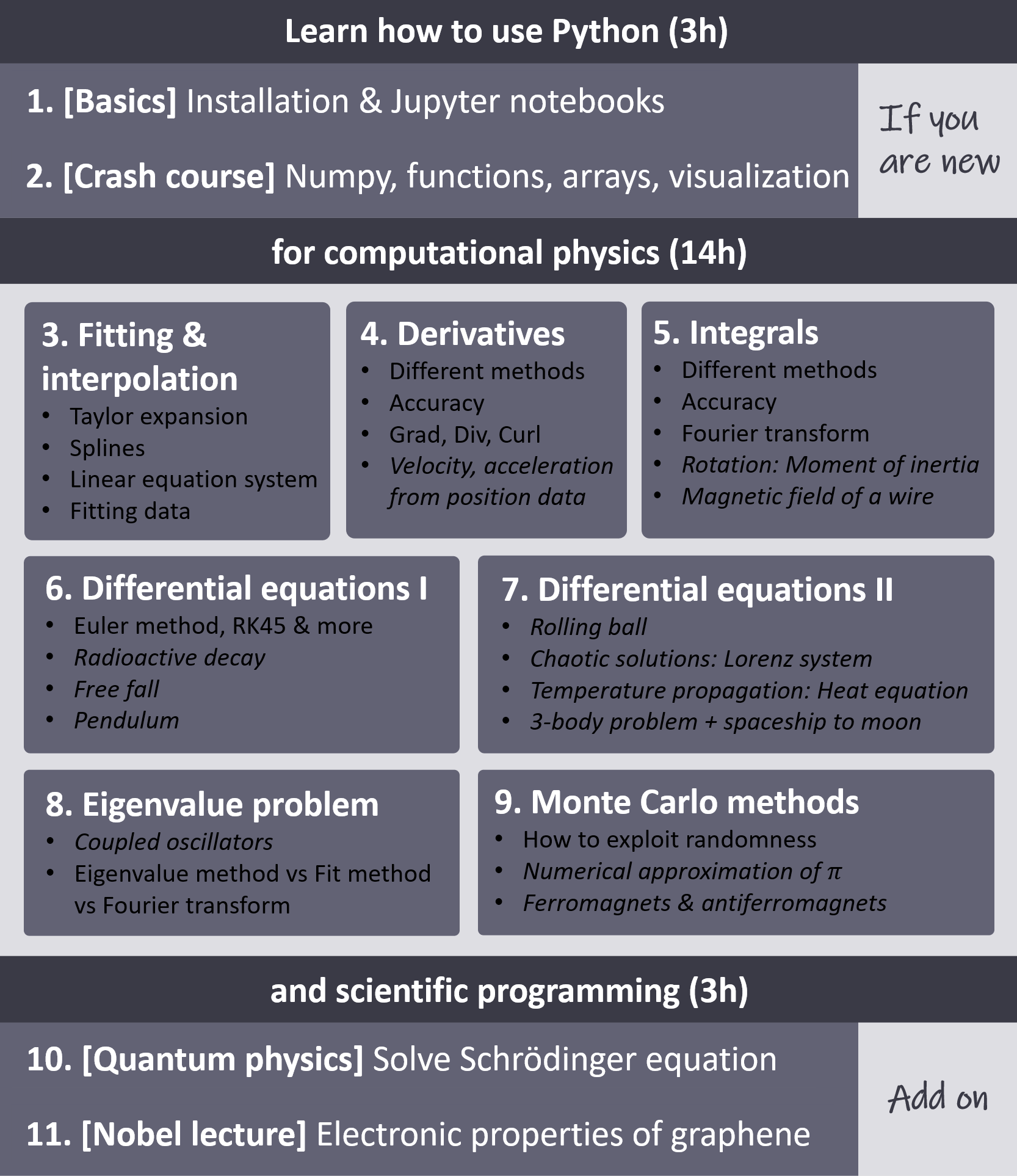
Computational Physics: Scientific Programming with Python

Sección 1: Python installation via Anaconda & Alternatives

1. Hello & Welcome!

2. Structure & Overview of this course



3. Installing Python via Anaconda for free

4. Jupyter notebook - Our tool of choice

5. Style your notebook

Cuestionario 1: Test your knowledge about the basics: Python in Jupyter notebooks

6. HOW TO use this course

So before we get started with the actual content of this course, let me give you a hint on how I want you to use the course. So I will often provide you a template file where I've basically prepared already the heading and some background information, like some equations for the numerical part. And then you watch my videos and see me code the problem and I want you really to code along with me and the end of the section. I will always provide you the notes book that I have prepared, so you could just download this and open this. But I think if you would do this, then you wouldn't learn so much about Python because programming is really learning by doing. And this is why I want you to work with a template file and to program really along with me. And I absolutely encourage you that when you have your own idea or when you see some potential for improvement in my course, please just post a video and try to do this. It really helps enormously your progress in the Python language if you just try out new things and if you really realize your ideas. OK, so now we can continue. And as I says, please always download the template files and work with them. And then in the end, you can compare your notebook with the notebook that I provide, which I have programmed.

7. LET'S GET STARTED with scientific programming!

LET'S GET STARTED with scientific programming!

Now everything is installed, so let's get started. Please note:

The next few lectures of this section are optional. I will:

show you ways to style our notebook explain to you typical errors that may emerge in jupyter notebook introduce to you alternative development environments

If you want to get right into the topic, feel free to skip these videos for now.

The following section is a crash course on python. It is optional as well and serves the purpose to get all the students on a similar level.

This means, the real course starts in section 3. See you soon :-)

8. (FAQ) Typical problems & errors

**Frequently asked questions** (a growing collection of my hints and students' remarks)

* *I have multiple hard drives. How can I change directory e.g. from C: to D:?*

Open a terminal and navigate to the desired directory. Type jupyter notebook and jupyter will open in the current directory.  
Example Windows: Open Anaconda, launch CMD.exe Prompt, run D:, run jupyter notebook

* *The figures in jupyter notebook are too small. I cannot see details. How can I fix this?*

Run the following cell in your notebook: plt.rcParams['figure.figsize'] = [40, 15]  
(The numbers describe the standard width and height of figures. You may have to adapt them according to your screen resolution and browser zoom to find the optimal values.)

Alternative: Export the image with high resolution.  
In the same cell of your plot write at the end: plt.savefig('name.png', dpi = 300)

* *While working on my notebook, I get the pop-up message:  
  “Notebook validation failed: Non-unique cell id 'waiting-opening' detected.”  
  I cannot even save my notebook anymore.*

It is unknown what exactly causes this issue but it sometimes occurs after copying cells in the notebook causing non-unique IDs. A quick-and-dirty fix is to select the whole notebook [Ctrl+A], cut (click the scissors icon) and paste right after (paste icon).  
(Your whole notebook will disappear in the process but gets restored with unique IDs.)

9. (optional) Style sheets for your notebook

One last thing that I want to show you is how you can really use a style sheet for your Jupyter notebooks because to be honest, I think opening this Jupyter notebook in this browser looks a bit ugly. So we have this a white background, which is quite bright and I don't really like it so much. But in the following of the course, I will, of course, continue just with the standard Jupyter notebook in the browser so that everyone can follow along and has the same visuals as I have. But if you don't like this style of the Jupyter notebook, I want to show you something. So I open up a file that I have prepared previously, which is called notebook, underscore style. So it doesn't really matter. You can just open up a new notebook by clicking here. New notebook. And here we are. And what I wrote there is two command pip space install space Jupiter themes, and this installs something called Jupiter themes. So you write this and then run it using Shift Enter, and then it is still running. Now it gives you the output, which basically says for me that it's already satisfied because of course, I have already installed it. Then you go here and add a new cell that clicking on the plus and you type pip install minus minus upgrade Jupiter themes, which updates the Jupiter themes. Maybe this is not necessary, but it's better to do it. So once again, it's tells me that all of this has been done already, and now we can use the actual commands of Jupiter themes by writing exclamation mark JT for Jupiter themes minus L. This is just a command which gives you all of the available themes that we can choose. And, for example, a theme that I like quite a lot is called Monarch. So I write exclamation mark j t minus T monarchy shift enter. And here's something that could happen to you as well. We have an error that we do not have the permission and to solve this error. What we have to do is we have to restart Anaconda and run it as an administrator. So I save this, I save this and then I just basically, yeah, we can just close it and maybe even close. The Anaconda Navigator quit. Yes. And now run it as an administrator. So let's wait again a few seconds until Anaconda started. So I'm just doing this because I want to show you that whenever you want to really install something and change something, you must run Anaconda as an administrator. However, if you're just writing some code and this is not necessary. OK, so here we are again. Let's sorry. That's run Jupyter Notebook. And you can see now it finally or it works already. So you see the style looks different, so it has worked. So let's open up again our notebook style and go here and you see how the style has changed. And this is because we have run JT one. Okay, and you see, this time it doesn't give us an error message. You can, of course, also choose another option. Like, I don't know, let's just test some of these or let's use the first one, just crash and to make it work. You click on refresh. And then you see, once again, a different style. So if you don't like all of these or you just want to have the same style as I have, you just right acclamation. Mark J. T minus are to restore the default and then you just reload and everything is back to normal. All right. So this was just something that I wanted to show you to prove to you that you can really change the visuals of this Jupyter notebook and also make it look more beautiful. But now we really go back to the initial style. And so in this way, it is most accessible for everybody. So here we can also reload this, OK? Here we are. And another thing that I want to show you is that, for example, you can click on running and then you see all of the files that are running. So currently we only have the notebook style file running. But for example, if I open up the old file, our first notebook and I run the whole kernel restart and run all. Then. We have now two notebooks running, and this is even true if I close this one and refresh. Still, both of them are running, which means I can now go here packed files and reopen them. And so they are still open. And if you want to close the file, so or better to say if you want to close the kernel, you can click here on shut down kernel or you just click on closed to whole browser and then everything is closed once you close or so Anaconda Navigator quit. Yes, and everything is closed. So this was a bit more information about Jupyter Notebook and in the next lectures, I want to explain to you a bit more about Pi Charm and some other text editor that you can use to write Python Code and also Jupyter notebooks in a more comfortable way.

10. (optional) Alternative development environments: For large projects - PyCharm

11. (optional) Alternative development environments: Allrounder - Visual Studio Code

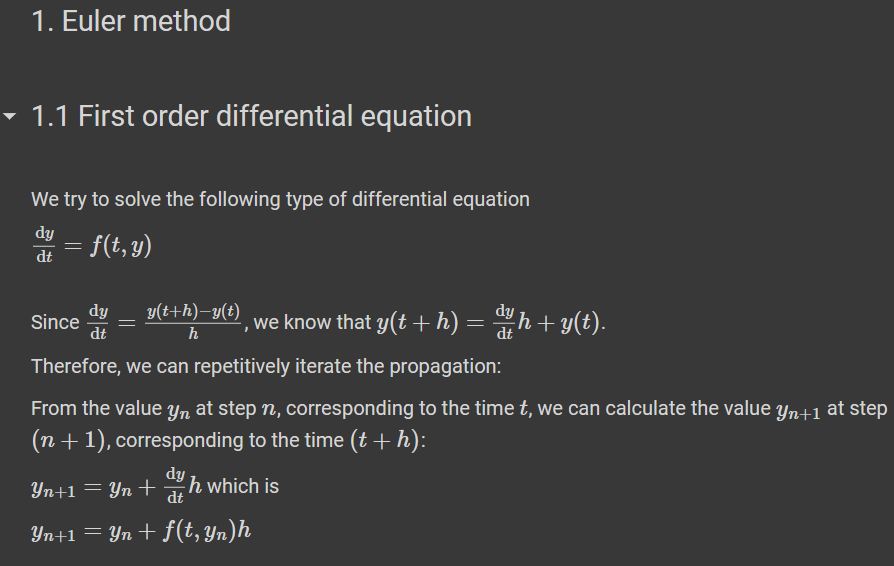
12. (optional) Environments & Updates

Seccion 6: Differential equations 1

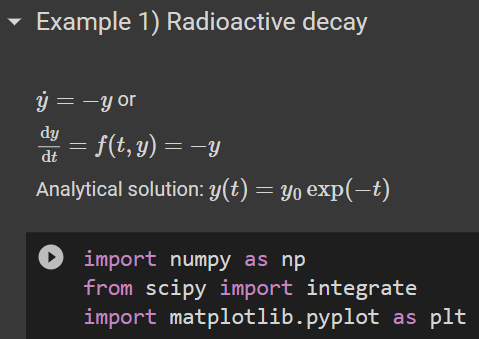
87. Introduction

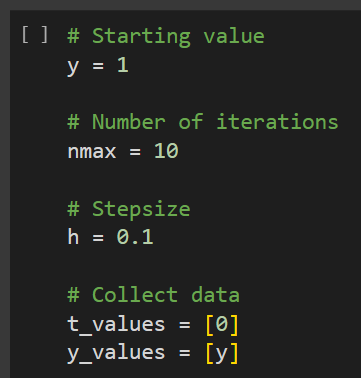
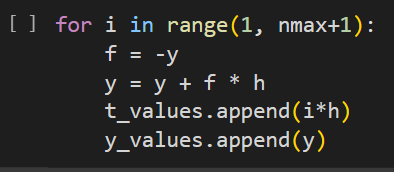
88. Template file

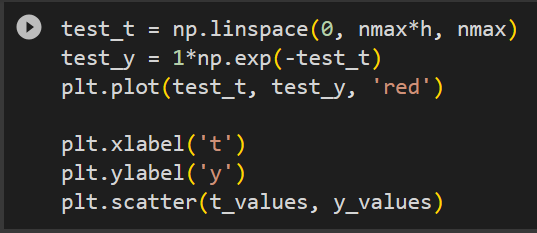
89. Background: Euler method



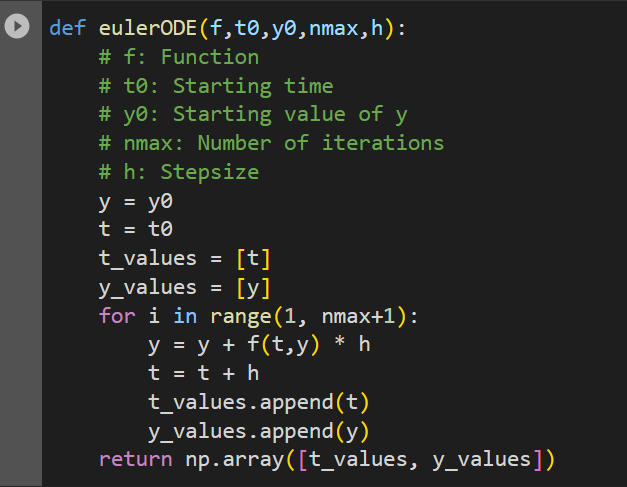
90. Example 1: Radioactive decay

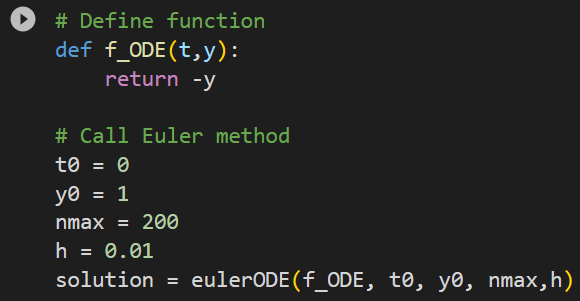


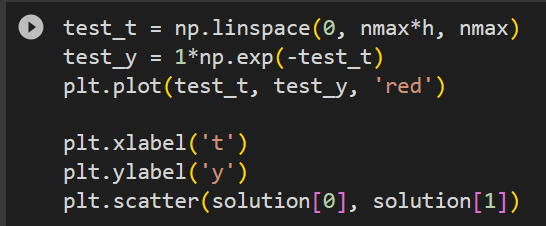
 



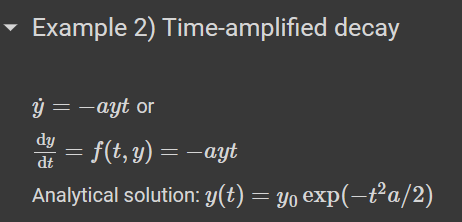
91. Defining a general function for the Euler method

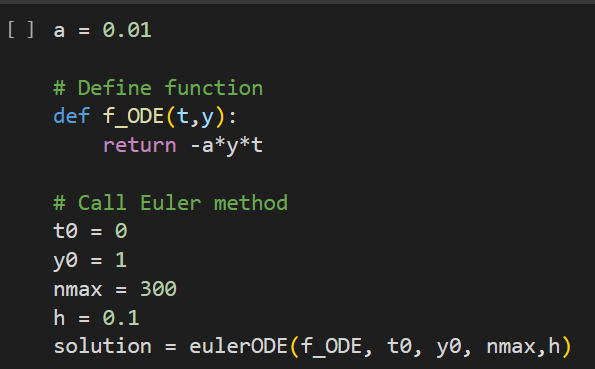


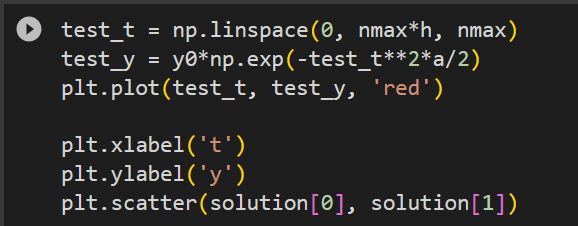




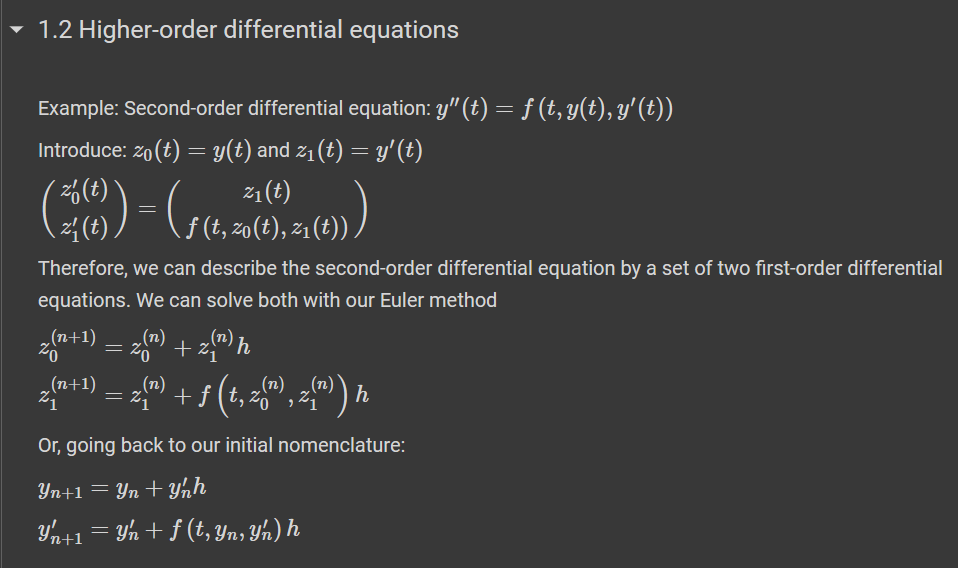
92. Example 2: Time-amplified radioactive decay

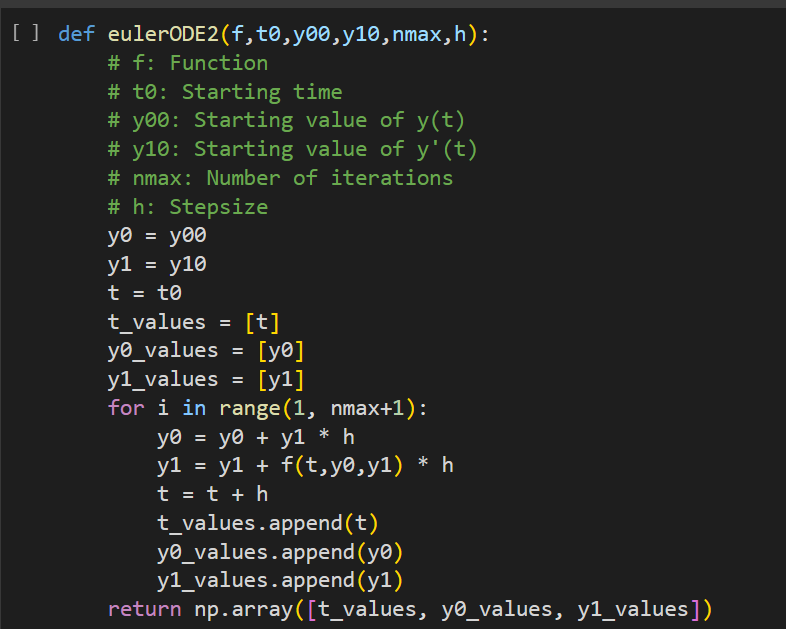




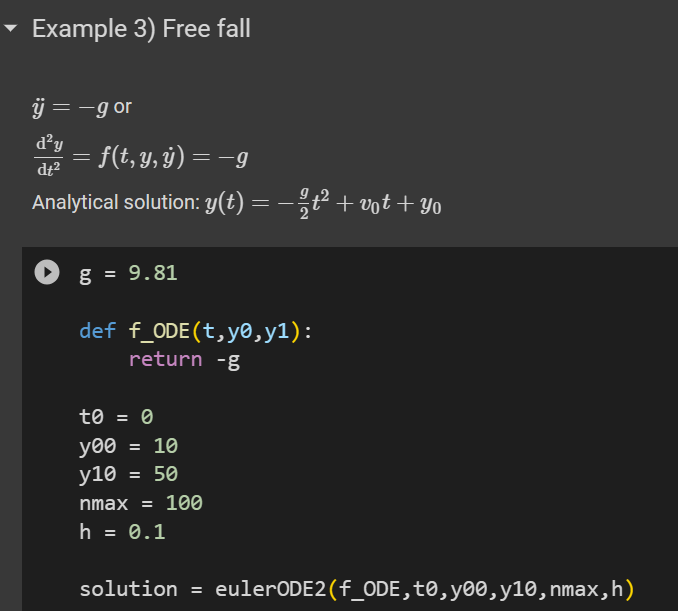


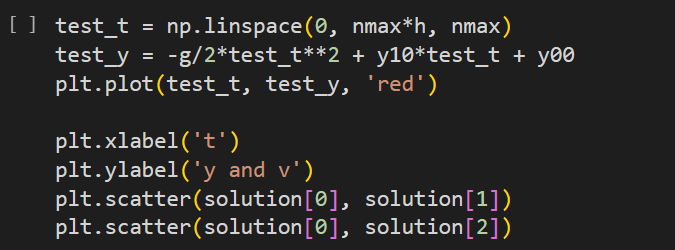
93. Higher-order differential equations





94. Example 3: Free fall

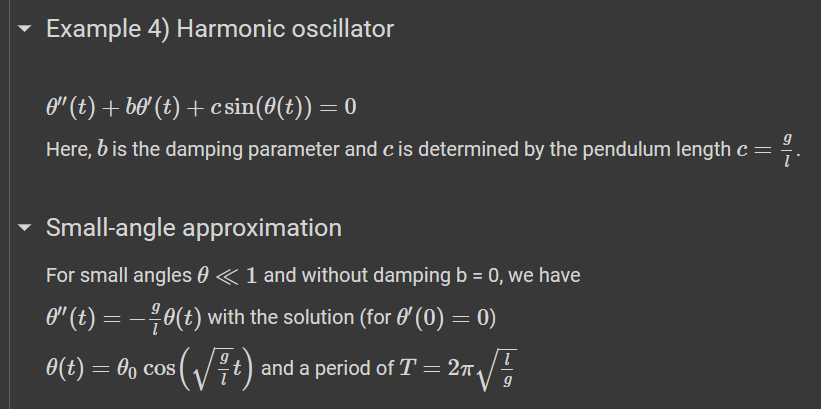


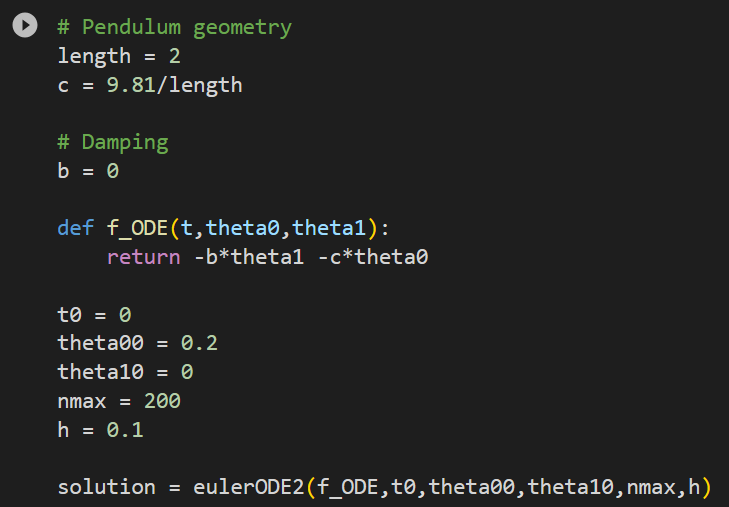


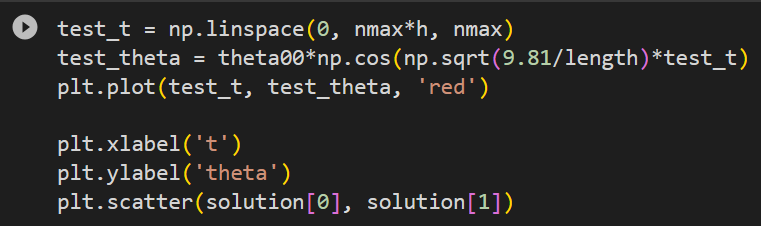
Detener

95. Example 4: Pendulum

Hay que revisar la ODE

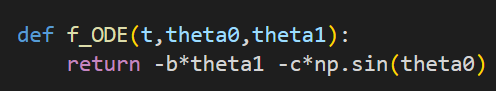






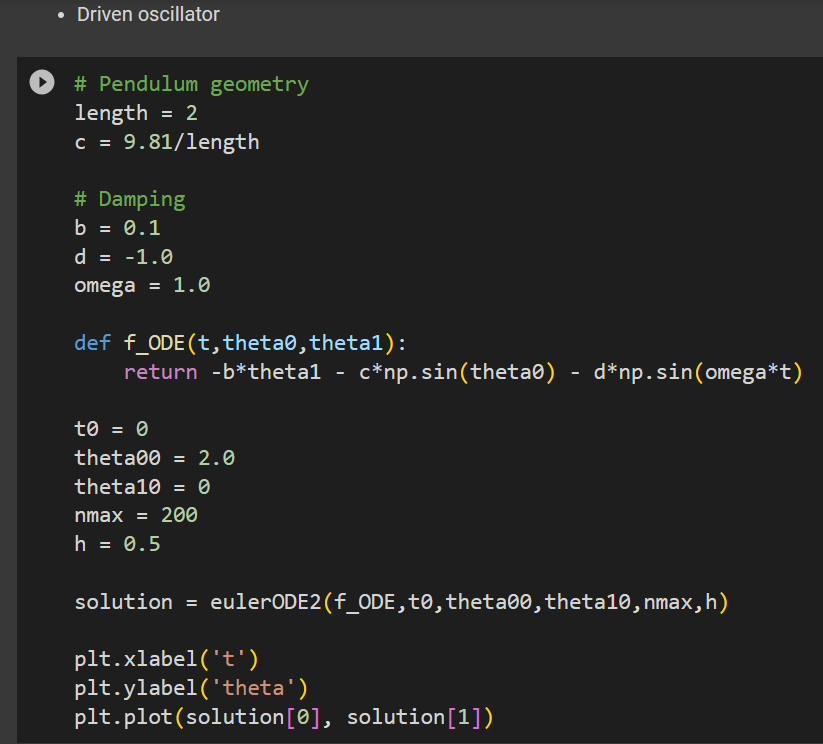
96. Accurate solution of the pendulum

Aqui hacemos lo mismo pero modificamos la ecuación diferencial

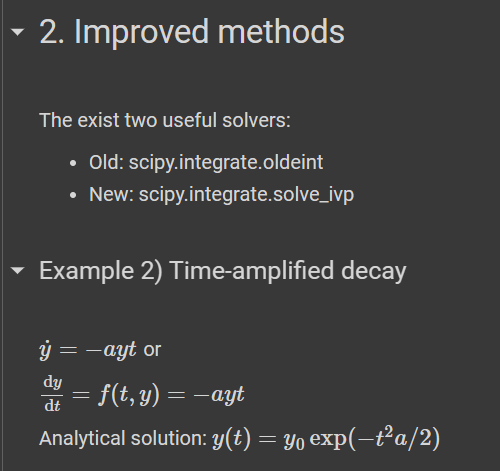


97. Adding damping and driving forces

Aquí aplicamos damping y fuerzas externas



98. Improvement: Use the SciPy function solve\_ivp



<https://docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.solve_ivp.html#scipy.integrate.solve>

So here we first have to provide our function like in our own solution.

Then we need the time span and this is here given by such a list of two values starting value and end

value. Then we have to provide the starting value for the value of y.

And here we use one and we have to give it also as a list. The reason for this is that once we try to solve higher order differential equations, then we would

have to provide here two or more values, as we have seen for the free fall, where we have to provide

the starting position and the starting velocity.

So here, from a logical point of view, it wouldn't really be necessary to use this as a list, but

that's the syntax.

So we have to provide a list.