MSc Applied Bioinformatics
alexey.larionov@cranfield.ac.uk

Working with Conda environments

Practical 1, part B

Contents

It is a good practice in bioinformatics to create a virtual environment for each project. This facilitates reproducible research and allows avoiding conflicts between dependencies. Conda is one of the most popular tools for creating and managing Python virtual environments.

Our task during this practical is to print "`I` 💔 `Conda`" using ***emoji*** package. For this we will create a Conda environment, add ***emoji*** package into this environment, and then write a script that will import and use this ***emoji*** package.

First, let's create a separate project folder for this task. Close the previous "Hello_Word" project folder. Make a new project folder, call it "Hello_Conda" (or give it any other name: just don't use spaces in the folder names!).

Identifying available *base* Conda environment

Press F1 to open ***Command Palette***, then type `Python: Select Interpreter`, select an available ***base*** Conda environment as illustrated below:

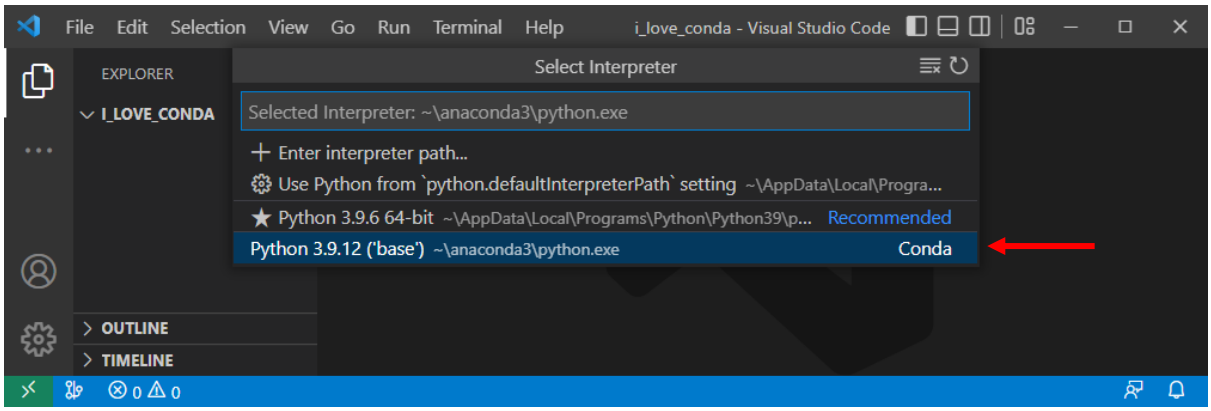

Figure 1: Selecting Conda base environment

Open a new terminal:

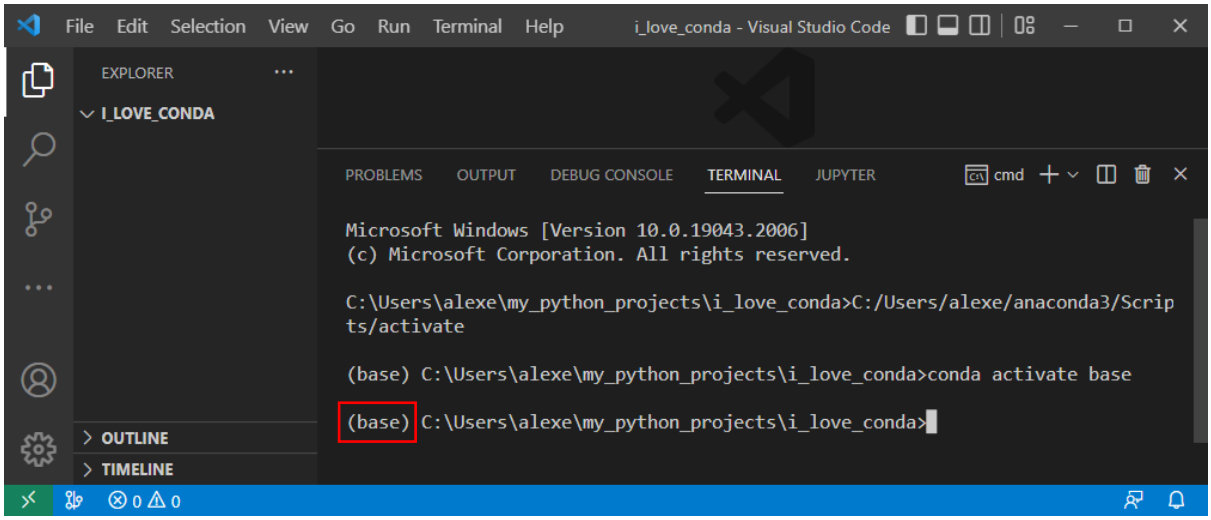

Figure 2: CMD terminal with Conda base environment

Note `(base)` at the start of command prompt: it confirms that Conda base environment has been activated. This means that you can proceed with creating a new Conda environment.

Creating a new Conda environment

Create a new Conda environment for your project (later you will install **emoji** package in it):

```
conda create -n emoji_env
```

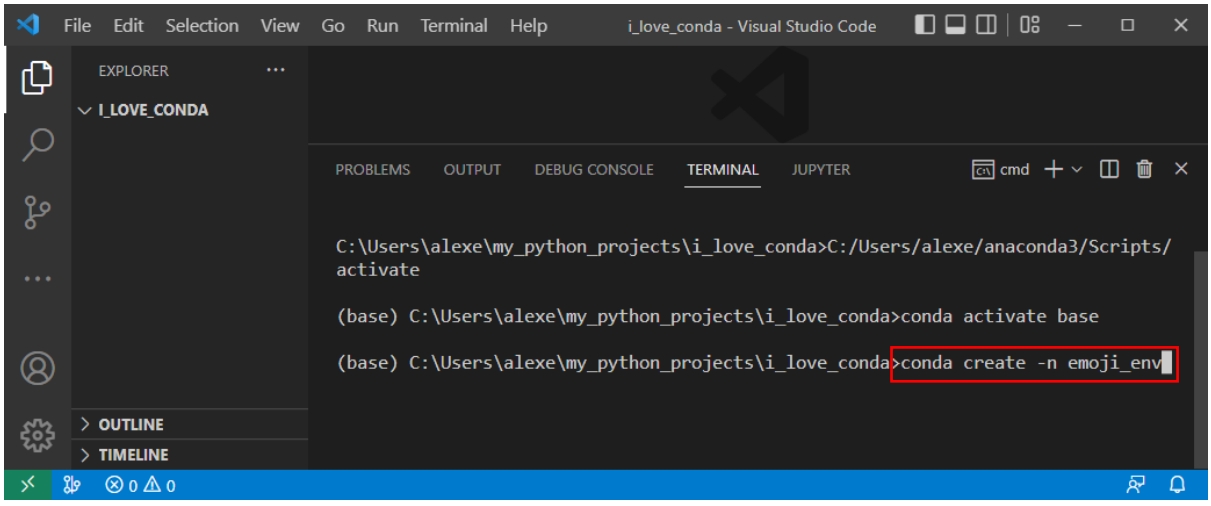

Figure 3: Creating a Conda environment

In the example above, we call the new environment `emoji_env`. You may call the new environment in any way you like (of course, avoiding spaces and special characters).

Conda will collect necessary information and ask you whether to proceed; type “y” (yes):

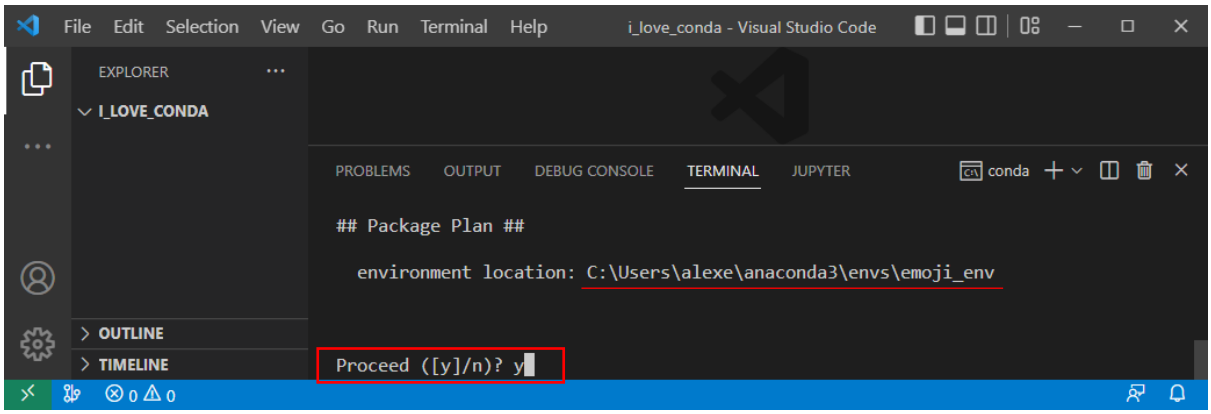

Figure 4: Interactive messages in terminal during creation of Conda environment

Note that one of the messages in Figure 4 shows the location of the new environment. You can see that Conda keeps its environments separate from the projects. In this example Conda keeps environments in `C:\Users\alexe\anaconda3\envs` folder. It will be a different folder on your laptop or on the Lab PC

After the new environment has been created, it is not activated automatically: you can see on Figure 5 that the active environment in terminal is still (base).

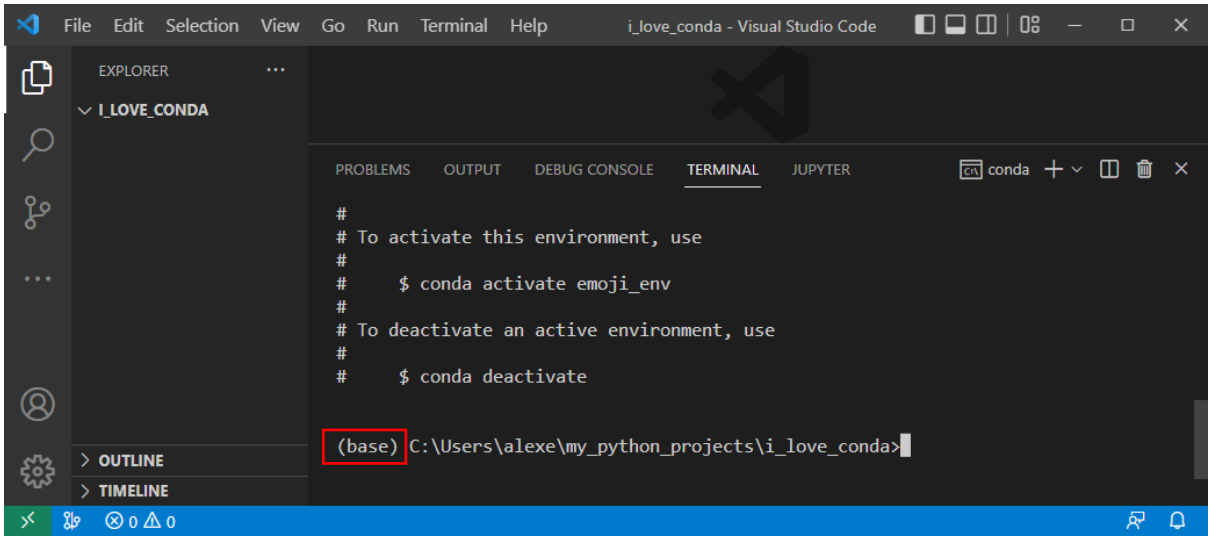

Figure 5: Messages in terminal after making a Conda environment

So, you need to activate the Conda environment after creating it.

To activate Conda environment in terminal, use command `conda activate …` :

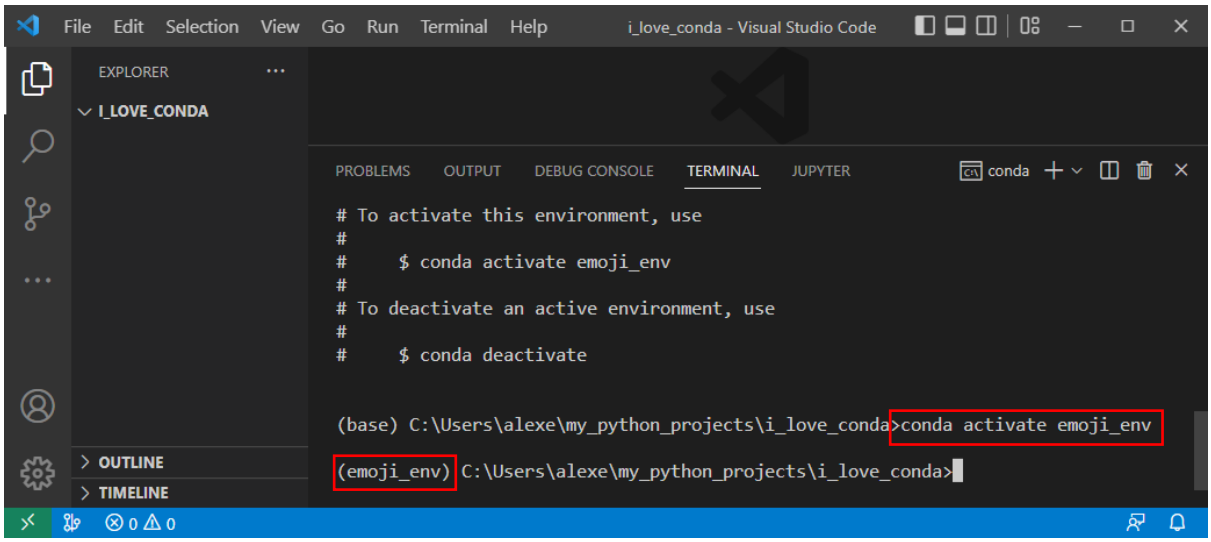

Figure 6: Activating a new Conda environment

Adding a package into the new Conda environment

Finally, when the new Conda environment is active, you may install **emoji** package in it:

```
conda install -c conda-forge emoji
```

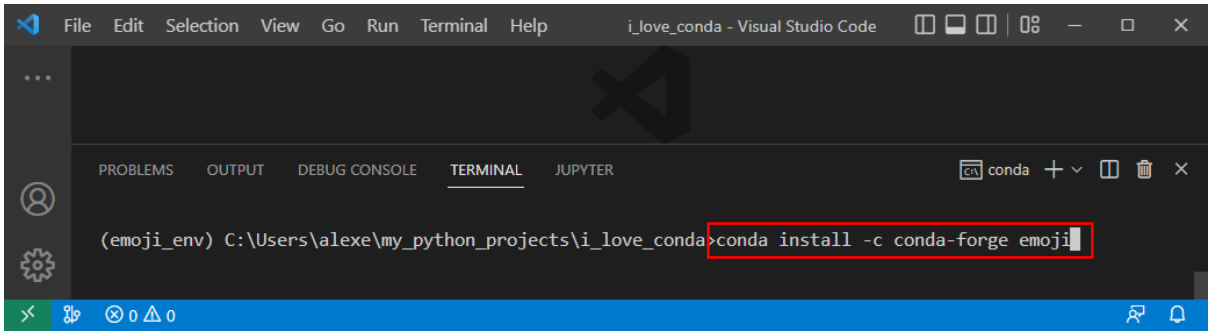

Figure 7: Installing a package from ***conda-forge*** channel

After the successful installation of **emoji** your terminal may look like this:

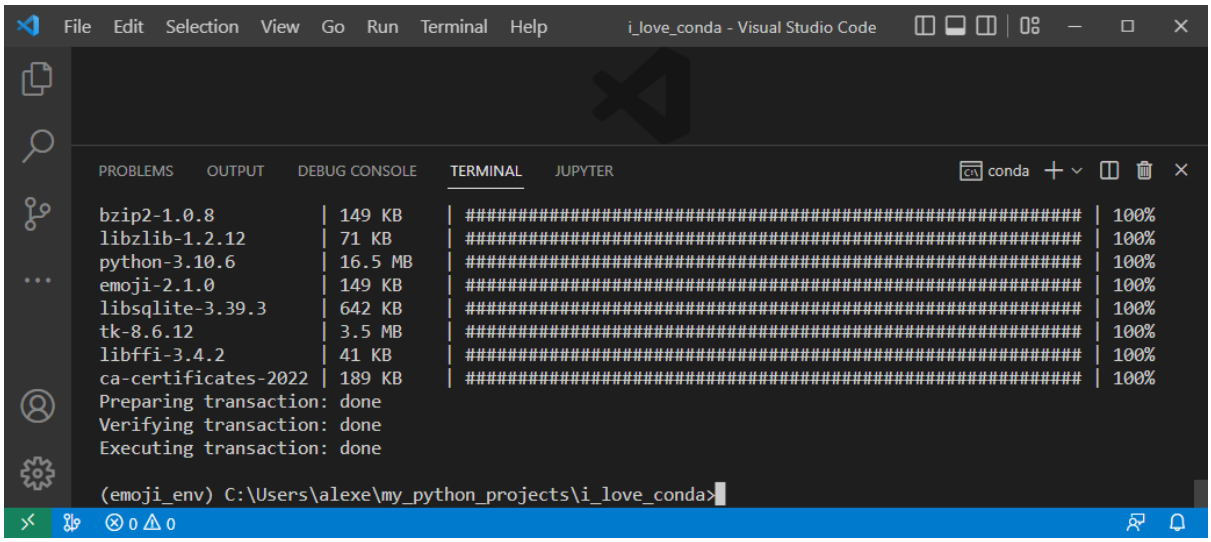

Figure 8: Successfully installed Conda package

You may note that in addition to **emoji** Conda has installed many other packages to the environment. It's OK: they are needed for **emoji** to work. Later we will install even more packages to this environment to use it in Jupyter etc.

In the installation command we explicitly asked to install emoji from the Conda repository called **conda-forge** (https://conda-forge.org). In the Conda word the repositories are called "channels". Hence the option looked like `-c conda-forge` . There are many Conda channels. The most popular bioinformatics channel is **bioconda** (https://bioconda.github.io).

Writing and executing the script that uses emojis ♡

Now, when we have a Conda environment with **emoji** package in it, we may close the terminal (hint: use trash-bin icon, NOT the cross symbol) and write the script, which uses this package. Create a new file in your project folder, call it ***i_love_conda.py***, and add the following code to the file:

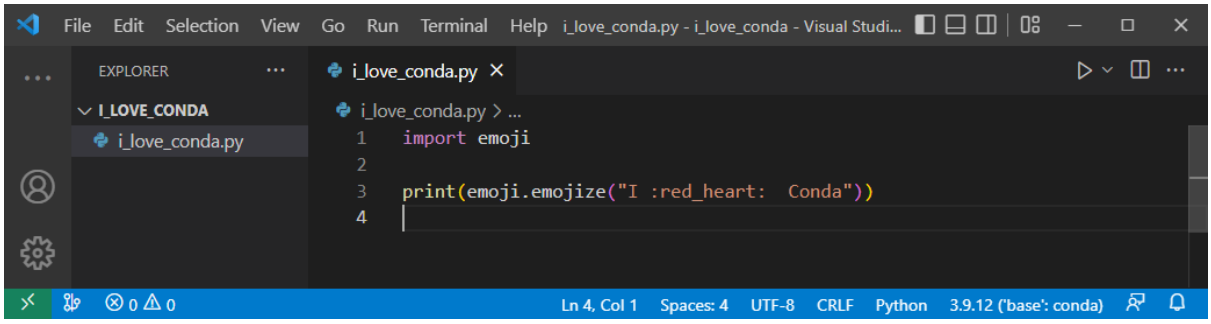

Figure 9: I love Conda script

Save the script (File > Save or Ctr+S).

Now, it's VERY important to select the correct Python interpreter to run the script. Note that the Python interpreter that you used in the terminal (e.g. when you created the new Conda environment in the terminal) may be different from the interpreter selected by VS Code for execution of your script.

For instance, Figure 10 shows a system that is going to use a wrong interpreter (indicated by the red arrow) to run our script. Because the additional package required for the script (**emoji)** was installed into a different environment, you may get the Module Not Found error when you try running the script with this wrong interpreter.

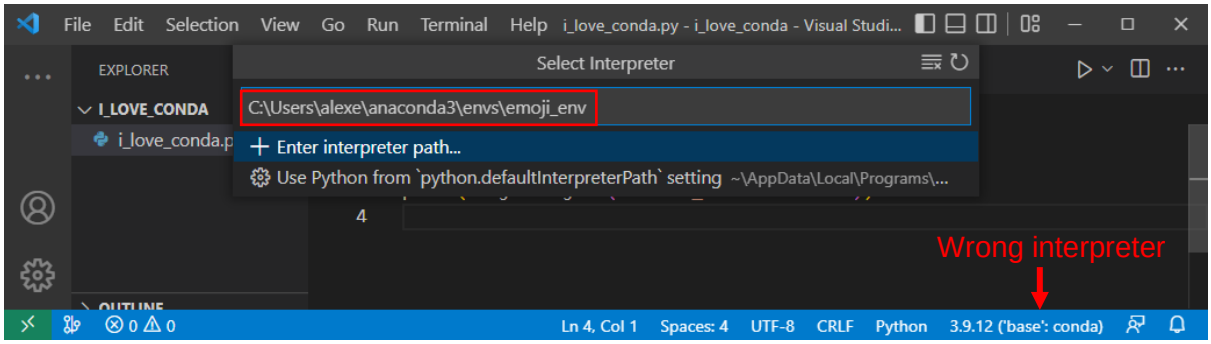

Figure 10: Selecting the required Conda interpreter

If you have the wrong interpreter selected by VS Code, you may click on it, and VS Code will allow you to select a different one (Figure 10). In this exercise we should select the `emoji_env` Conda environment, which we have just created.

In most cases, you should see the `emoji_env` in the environments list presented by VS Code. However, if VS Code still is unaware about the new Conda environment, which you just created, you may esnter the path to the interpreter e.g (see Figure 4):

```
C:\Users\alexe\anaconda3\envs\emoji_env
```

Note that if you work in Linux, the Python interpreter locations could be very different from what is illustrated above for Windows!

Now you should be able to execute the script:

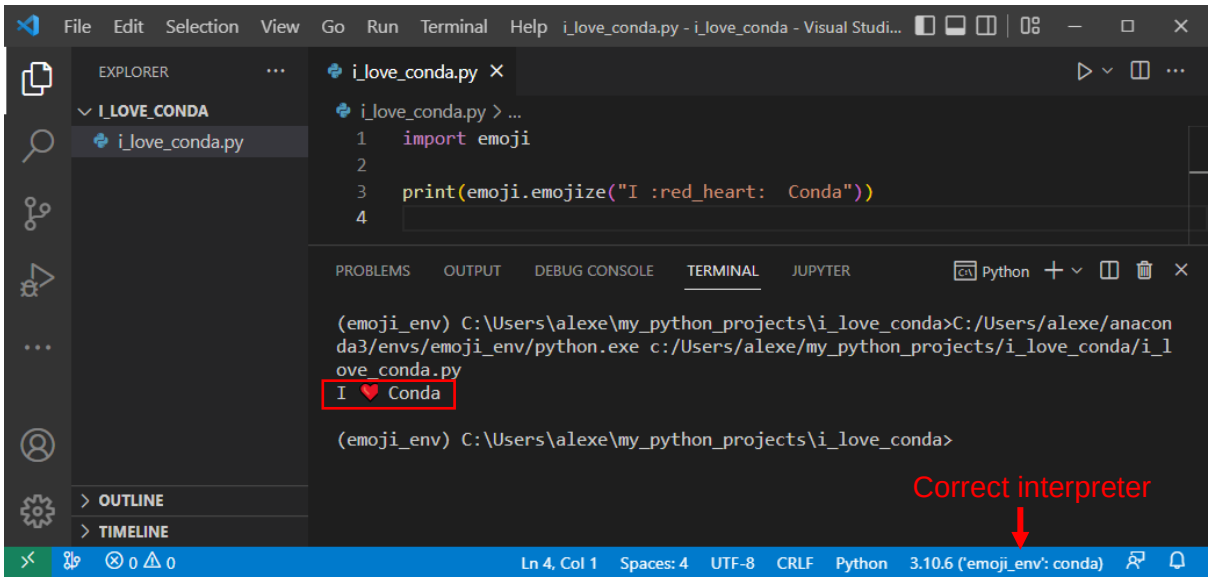

Figure 12: Successfully executed script

Hurray! You have successfully completed the task: you created a Conda environment, installed an additional package in it, and executed a script, which uses this additional package.

By the way, you may execute code line-by-line by using Sift+Enter.

Try to add more print statements using different emojies (e.g. **:thumbs_up:** or **:grinning_face:**) and run the code line-by-line.

Well done! You have finished the 2nd part of Practical 1