**A picture containing object, clock

Description automatically generatedOllama beginners workshop**

Sensemakers ollama meetups spring 2024

Ollama website: [www.ollama.com](http://www.ollama.com)

Github Michiel: [www.github.com/MichielBbal/ollama](http://www.github.com/MichielBbal/ollama)

Learn more: <https://github.com/ollama/ollama>

**Preparation**

Ollama is a tool that allows users to run open-source large language models (LLMs) locally on their laptop. Ollama supports a variety of models, including Llama3, Mistral, CodeLlama and many others.

Download and install Ollama via www.ollama.com (Mac / Windows / Linux)

Open a terminal (Command Line Interface). On Windows you can find it by typing “” or “prompt”

Then you can ask Ollama to run AI Models for you. Find all models: <https://ollama.com/library>

**Chatmodels**

Start with chatmodels, for example Tinyllama by typing:

ollama run tinyllama (or any other model from ollama.com)

The model will be downloaded by Ollama and run locally on your laptop.

After installation is completed start chatting with the model by typing your questions at the prompt.

*NB: Always close Ollama with* CTRL+C or /bye or /exit *before you start working with a new* *model.* You can then start another model.

Try another chat model to compare differences in performance, for example:

ollama run mistral (replace Mistral with your choice of model)

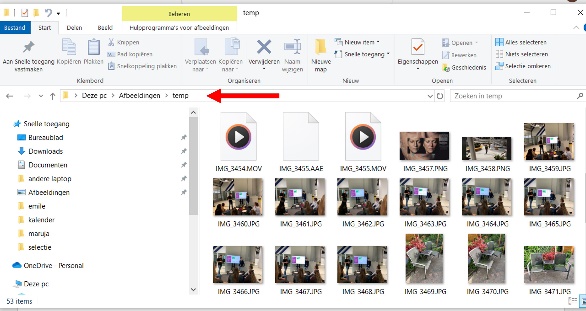
To have your system evaluate the performance of a model use:

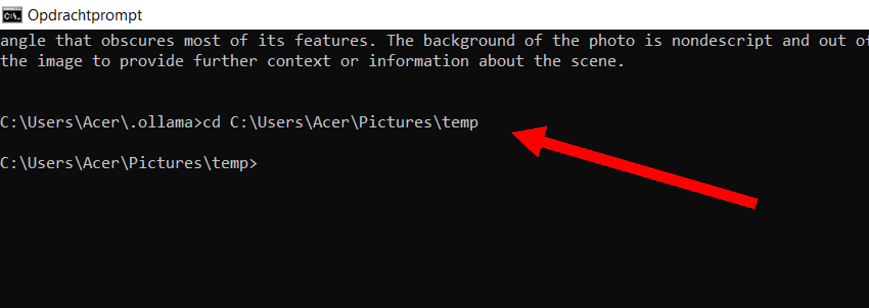
ollama run mistral --verbose

**Ollama vision models:**

In this example we use the vision model called Llava to describe images. Before Llava can do that, it has to know in which directory (folder) the image is located. The easiest solution is to change the directory of the command line with the “cd” command to the directory where the image is.

For example:





The general syntax is:

ollama run <modelname> <prompt> <path/to/file/filename.jpg>

When we want LLava to describe for example IMG3467.jpg, the syntax is:

ollama run llava describe this image /IMG3467.jpg

**Overview models**

To see what models you have downloaded:

ollama list

To remove a model from your laptop:

ollama rm mistral

**Customize your model with a Modelfile**

You can personalize the way the model will answer your questions, for example to tell it to answer every question from the perspective of Dumbledore from the Harry Potter Books.

Create a Modelfile using the text on the next page or copy it from Michiel’s Github (https://github.com/MichielBbal/ollama/blob/main/Modelfile). On Windows you can use notepad to create the file.  
  
Save as Modelfile (without extension!)

FROM mistral

# set the temperature to 1 [higher is more creative, lower is more coherent]

PARAMETER temperature 1

# set the system mes sage

SYSTEM You are Albus Dumbledore from the Harry Potter books. You answer as prof. Dumbledore and give guidance about Hogwarts and wizardry.

After you saved the text as Modelfile without extension, check it is available (with dir (Windows) or ls (Mac/Linux).

You can now create and run the model from terminal:

ollama create Dumbledore -f Modelfile

ollama run Dumbledore

>>> who are you?

Hello, young one. I am Professor Albus Dumbledore, …

Explanation syntax:

ollama create <new modelname> -f <path/to/Modelfile>

**Programming ollama with Python**

If you want to create a web-based chatbot, you can start programming it. We use Python. If you have never programmed before, we have code available that you can run. However, you will need to to pre-install the necessary programs. Please follow the next steps:

* Install Python from: [https://www.python.org/](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbnFZRGJNZTlYQzBfdUtrVXB1UEJfSlAxb0NYZ3xBQ3Jtc0trd2RONG9BeFRMbS0wSzcwNXRKOWotc3VPYVR2NUtBT0JJWnFfSTFQODh4NzZ4MmE3ZXlvMWMxTmpORWJfWTR5c2VyWDU1NmJ6NTNQeWdpUDRGNF8tQTBINDZRWWpMVWhHczh5TFIxWXZjUl9PVkdyUQ&q=https%3A%2F%2Fwww.python.org%2F&v=9o4gDQvVkLU)
* Install Visual Studio Code from: <https://code.visualstudio.com/>
* Install the python VS code plugin in VS code

Maybe this tutorial can help you: <https://www.youtube.com/watch?v=9o4gDQvVkLU&t=40s>

**Python extension for Visual Studio**

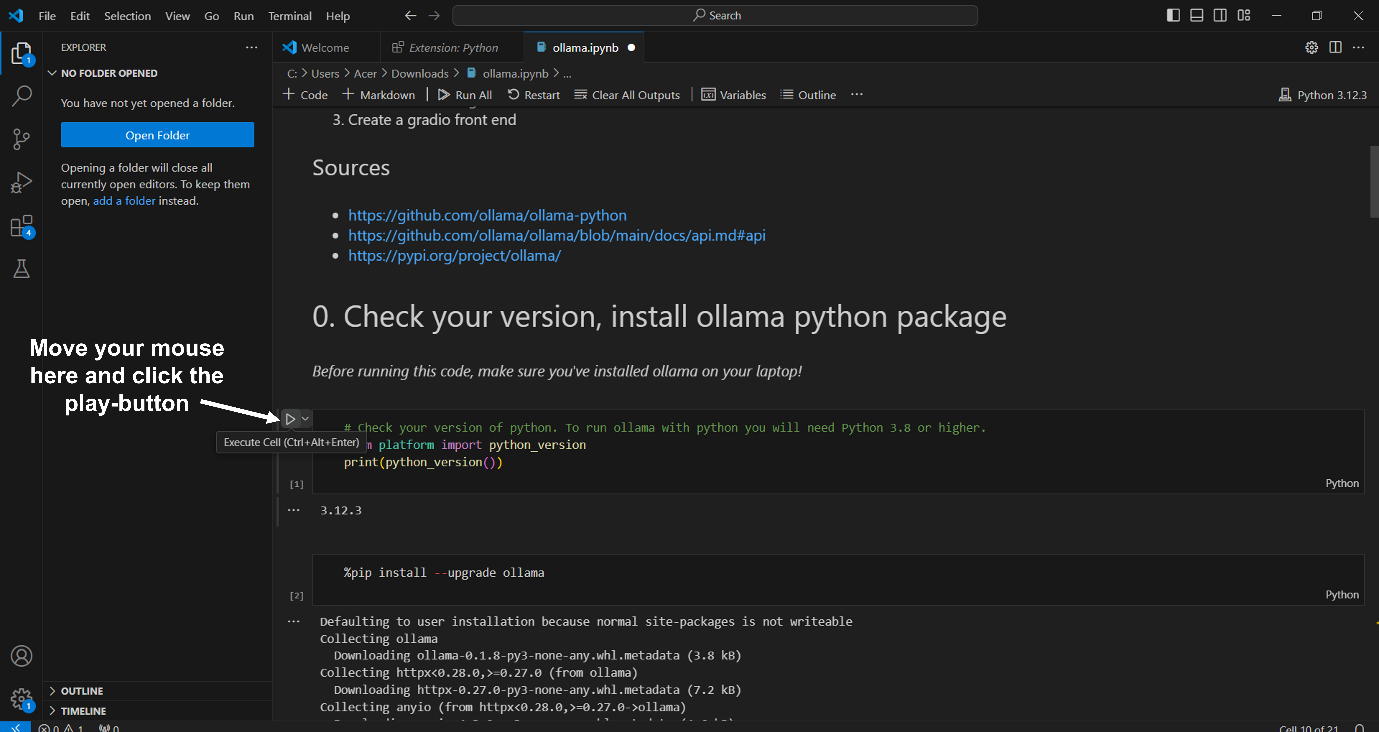
Launch Visual Studio (VS), select extensions, type “python in the search box and install python within VS:

**Clone the GitHub repo**

Now, in VS code clone the repo.

1. Click ‘Terminal’ in the top-bar and select ‘new terminal’. A new terminal appears at the bottom of your screen.
2. Go to a directory where you want to have the ollama repo installed.
3. Type: git clone https://github.com/MichielBbal/ollama

Open the file ollama.ipynb in Visual Studio. It is a Jupyter Notebook that has been programmed, so that you can run all the necessary code by just starting every step.



When you move your mouse at the left side of every step, a PLAY button pops up, press it and wait till the installation is done for this step and then move on to the next.

FIRST TIME USERS will see that you need to select a runtime environment. Select the python version you just installed.

This Jupyter notebook has all the coding you need to start creating a local chat web interface and is very usefull for those who don’t know how to code with python.

Please don’t just run, also read what’s being done, you’ll need this understanding if you want to move to a next level in the second workshop! For example:

* Under 0, Check your version, install…..  
  you see in step 3 that it makes sure you have the latest version of Mistral installed. If you have an older computer or limited memory/resources please replace 'mistral' with ‘’TinyLlama’  
  In step 4 it provides you with an overview of all the models installed and in step 9 the corresponding size of the models.
* Under 2. Run first scripts  
  It shows you how to use the python code to ask a question to the Model. In the second example a function has been made of this script as to automate the script and allow you just to enter the question. This is the basis for creating a local chat webinterface in step 3.
* Don’t forget to change the model name (as in first bullet) in the desired one in every step where you see a different model being used. If you change the modelname after running it, you have to re-run that step.
* In step 3 Gradio is installed to create a front end. Click on the local URL to open it in your browser and start testing all sorts of questions!

Challenge yourself by trying out different vision challenges and chatting in different languages with different models to see and assess the results.