

# WizardLM-2

[WizardLM-2](#)

[Model scale](#)

[Got WizardLM-2](#)

[Use WizardLM-2](#)

[Run WizardLM-2](#)

[Dialogue](#)

[End conversation](#)

[Reference material](#)

## Demonstration environment

### Development Board : Raspberry Pi 5B

**SD(TF)card:** 64G (Above 16G, the larger the capacity, the more models can be experienced)

Raspberry Pi 5B (16G RAM): Run 14B and below parameter models

Raspberry Pi 5B (8G RAM): Run 8B and below parameter models

Raspberry Pi 5B (4G RAM): Run 3B and below parameter models, can't run WizardLM-2 model

Raspberry Pi 5B (2G RAM): Run 0.5B and below parameter models, can't run WizardLM-2 model

WizardLM-2 is Microsoft's latest advanced large-scale language model, which improves performance in complex dialogue, multilingualism, inference, and intelligent proxy capabilities.

## Model scale

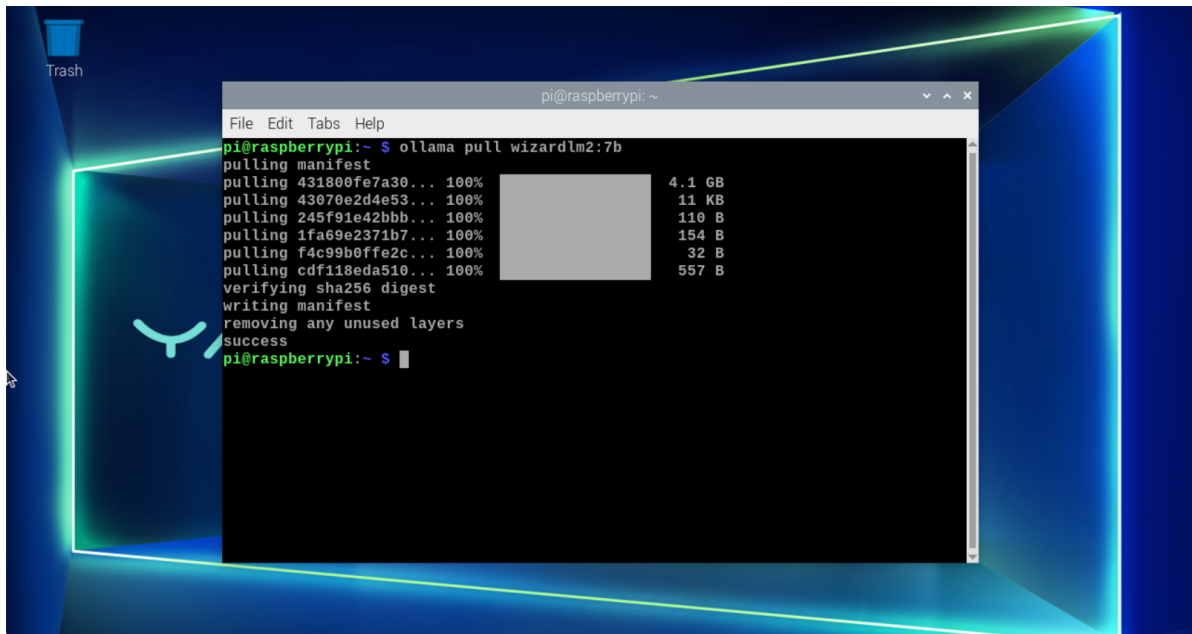
Model	Parameter
WizardLM-2	7B
WizardLM-2	70B
WizardLM-2	8x22B

Raspberry Pi 5B (8G RAM): WizardLM-2 model testing with 7B parameters.

## Got WizardLM-2

Using the pull command will automatically pull the models from the Ollama model library.

```
ollama pull wizardlm2:7b
```



## Use WizardLM-2

### Run WizardLM-2

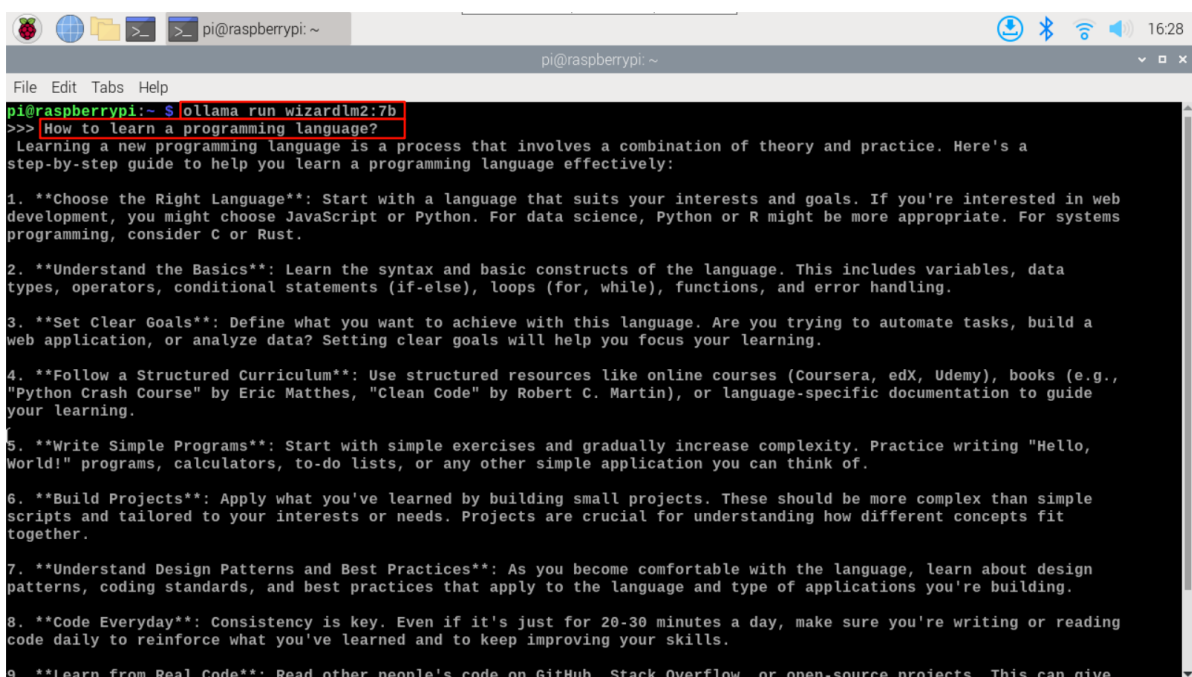
If the system does not have a running model, the system will automatically pull the WizardLM-2 7B model and run it.

```
ollama run wizardlm2:7b
```

### Dialogue

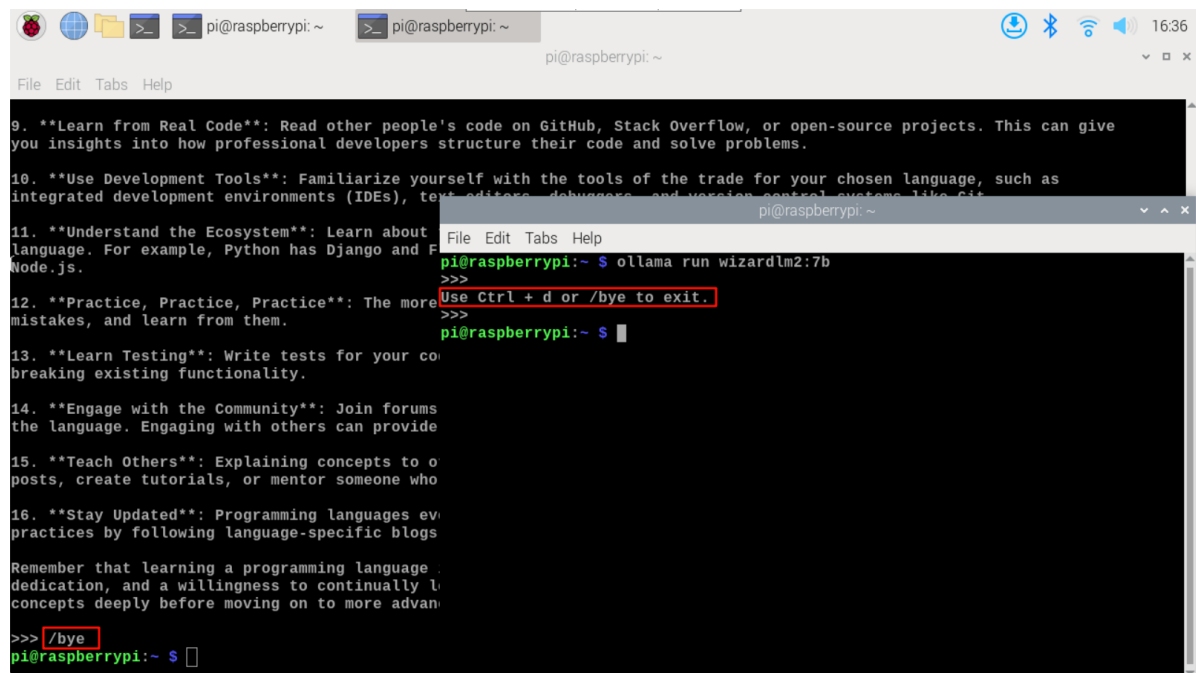
```
How to learn a programming language?
```

The response time to the question is related to the hardware configuration, please be patient and wait.



## End conversation

The response time to the question is related to the hardware configuration, please be patient and wait.



The screenshot shows a terminal window on a Raspberry Pi. The terminal displays a list of 16 tips for learning programming, followed by a terminal session with Ollama. The tips are:

9. **Learn from Real Code**: Read other people's code on GitHub, Stack Overflow, or open-source projects. This can give you insights into how professional developers structure their code and solve problems.
10. **Use Development Tools**: Familiarize yourself with the tools of the trade for your chosen language, such as integrated development environments (IDEs), text editors, debuggers, and version control systems like Git.
11. **Understand the Ecosystem**: Learn about the ecosystem of the language. For example, Python has Django and Flask, and JavaScript has Node.js.
12. **Practice, Practice, Practice**: The more you practice, the more you learn. Don't be afraid of making mistakes, and learn from them.
13. **Learn Testing**: Write tests for your code to ensure it works as expected and to prevent breaking existing functionality.
14. **Engage with the Community**: Join forums, chat rooms, or local meetups. Engaging with others can provide support, feedback, and new ideas.
15. **Teach Others**: Explaining concepts to others can help you understand them better. You can write tutorials, create videos, or mentor someone who is just starting out.
16. **Stay Updated**: Programming languages and technologies evolve rapidly. Stay up-to-date by following language-specific blogs, newsletters, and social media channels.

Remember that learning a programming language is a journey, not a destination. It requires dedication, patience, and a willingness to continually learn and grow.

The terminal session shows the following commands and output:

```
pi@raspberrypi:~$ ollama run wizardlm2:7b
>>>
Use Ctrl + d or /bye to exit.
>>>
pi@raspberrypi:~$ /bye
pi@raspberrypi:~$
```

## Reference material

### Ollama

Website: <https://ollama.com/>

GitHub: <https://github.com/ollama/ollama>

### WizardLM-2

Ollama model: <https://ollama.com/library/wizardlm2>