

```

Welcome to Ollama!

Run your first model:

ollama run llama3.2

PS C:\Windows\system32> ollama
Usage:
  ollama [flags]
  ollama [command]

Available Commands:
  serve      Start ollama
  create     Create a model from a Modelfile
  show       Show information for a model
  run        Run a model
  stop       Stop a running model
  pull       Pull a model from a registry
  push       Push a model to a registry
  list       List models
  ps         List running models
  cp         Copy a model
  rm         Remove a model
  help       Help about any command

Flags:
  -h, --help            help for ollama
  -v, --version          Show version information

Use "ollama [command] --help" for more information about a command.
PS C:\Windows\system32> ollama run llama3.2
pulling manifest
pulling dde5aa3fc5ff: 100% ▢ 2.0 GB

```

```

>>> hi, write hello world in python
Here's the "Hello, World!" program in Python:
...
print("Hello, World!")

```

```

>>>
PS C:\Windows\system32> ollama ps
NAME                ID                SIZE    PROCESSOR    UNTIL
llama3.2:latest     a80c4f17acd5     4.0 GB    100% GPU     4 minutes from now
PS C:\Windows\system32>

```

```

writing manifest
success
Thinking...
Okay, user just asked "hi what model are you" in a very casual way.

Hmm, they probably want to know my technical specs since that's the common question people have when interacting
with AI assistants for the first time. The tone is friendly but straightforward - maybe they're curious about
capabilities or just verifying it's an actual language model.

I should keep it simple and clear in my response. No need for jargon overload at this point, especially since they
started with "hi" rather than technical terms. A cheerful tone would match their casual opener well.

They might not realize there are different versions (GPT-3.5 vs GPT-4), so I'll mention both but position GPT-4 as
the default/current one unless specified otherwise. The version note is important because it affects what I can
actually do for them.

I wonder if they're asking because they've used other AIs before? Maybe this will help establish trust by being
transparent about my identity. Should keep the door open for follow-up questions too - hence ending with an offer
to explain further or ask something else.
...done thinking.

Hey! ▢ I'm currently using **DeepSeek-R1**, which is based on a model called **GPT-4** (or similar advanced
architecture). You can think of me as your friendly AI assistant, powered by DeepSeek's cutting-edge language
model!

If you're curious about the differences between versions or how I work, feel free to ask! How can I help you
today? ▢ ▢

>>> Send a message (/? for help)






















```

```
PS C:\Windows\system32> ollama list
NAME                ID                SIZE      MODIFIED
gemma2:27b          53261bc9c192     15 GB     40 seconds ago
deepseek-r1:8b      6995872bfe4c     5.2 GB    48 minutes ago
llama3.2:latest      a80c4f17acd5     2.0 GB    About an hour ago
```

```
(base) C:\Users\Andrzej\llm>conda env create -f environment.yml
Channels:
 - conda-forge
 - defaults
Platform: win-64
Collecting package metadata (repodata.json): done
Solving environment: done

Downloading and Extracting Packages:
```

```
(base) C:\Users\Andrzej\llm>conda activate llms
(llms) C:\Users\Andrzej\llm>
```

llms 	 networkx	 Python package for creating and manipulating complex networks
	 notebook-shim	 A shim layer for notebook traits and config
	 notebook_shim	
	 numpy	 Array processing for numbers, strings, records, and objects.
	 oauthlib	 A generic, spec-compliant, thorough implementation of the oauth request
	 ollama	
	 onnxruntime	 Cross-platform, high performance ml inferencing and training accelerator
	 onnxruntime-gpu	
	 openai	 Python client library for the openai api
	 openjpeg	 An open-source jpeg 2000 codec written in c.