

FLASH CARDS

Python and AI

Terminology and quiz cards for the lesson on wrangling data and building a safe local agent with Python.

What is in this set

40 cards: **24 term cards** (term on the front, definition on the back) and **16 quiz cards** (question on the front, answer on the back). Bracketed numbers on the backs, like [1], match the reference list on the last page and in the instructor guide.

How to print

1. Print double-sided on US Letter at 100% scale (no "fit to page").
2. Choose **flip on the long edge**. The backs are arranged so each answer lands behind its card.
3. Cut along the dashed lines. Each card is 3.75 by 2.5 inches.

How to use them

- **Self-test:** read the front, answer aloud, flip to check.
- **Pairs:** one reads the front, the other answers. Swap every five cards.
- **Boxes:** keep a "missed" pile and a "got it" pile. Re-test the missed pile after a break, then again the next day.
- **Warm-up:** deal a few quiz cards at the start of the next session.

Facts checked October 27, 2026. Tools and models change quickly, so confirm details against the current documentation. These materials are dedicated to the public domain under CC0 1.0 and shared for the public good. The PrivacySafe and PrivacySafe Foundation names and logos are trademarks and are not included in that dedication.

TERM

1 / 40

Script

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

2 / 40

Module and import

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

3 / 40

Function

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

4 / 40

Docstring

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

5 / 40

Type hint

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

6 / 40

List

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

7 / 40

Dictionary

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

8 / 40

CSV

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

2 / 40

A Python file whose functions you can reuse in another file with import. `local_agent.py` imports `wrangle`.

[\[1\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

1 / 40

A Python file you run from the command line, such as `python wrangle.py`.

[\[1\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

4 / 40

Text in triple quotes just under a `def` line. It documents the function, and the Ollama library uses it to describe a tool to a model.

[\[1\]](#) [\[11\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

3 / 40

A named block of code that takes inputs and returns a result. You define one with `def`.

[\[1\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

6 / 40

An ordered collection in square brackets, such as a list of rows.

[\[1\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

5 / 40

A note such as `group_by: str` saying what type a value should be. Tool descriptions are built from them.

[\[1\]](#) [\[11\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

8 / 40

Comma-separated values: a plain-text table with one row per line.

[\[2\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

7 / 40

A collection of key and value pairs in braces. `csv.DictReader` turns each CSV row into one.

[\[1\]](#) [\[2\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

9 / 40

csv module

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

10 / 40

Virtual environment

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

11 / 40

pip

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

12 / 40

pandas

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

13 / 40

DataFrame

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

14 / 40

NumPy

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

15 / 40

Data cleaning

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

16 / 40

None

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

10 / 40

A private folder of installed packages for one project, made with `python -m venv`.

[\[3\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

9 / 40

The standard-library module for reading and writing CSV files. No install needed.

[\[2\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

12 / 40

A library for tables of data, with fast grouping, filtering and cleaning.

[\[6\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

11 / 40

Python's package installer. For example: `pip install ollama`.

[\[4\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

14 / 40

A library for fast arrays and math that many data and AI libraries build on.

[\[7\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

13 / 40

The table object in pandas, with labeled columns.

[\[6\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

16 / 40

Python's value for nothing or missing. Test for it with `is None`.

[\[1\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

15 / 40

Fixing or removing bad data, such as stray spaces, odd spellings, blanks and duplicates, before analysis.

General

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

17 / 40

Group by

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

18 / 40

Agent

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

19 / 40

Tool (function calling)

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

20 / 40

Tool call

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

21 / 40

Agent loop

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

22 / 40

Allowlist

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

23 / 40

Excessive agency

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

24 / 40

Least privilege

Say it in your own words, then flip.

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

18 / 40

A model plus tools plus a loop, so the model can decide which tool to ask for next.

[\[11\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

17 / 40

Splitting rows into groups by a column and summarizing each group, such as total loans per branch.

[\[6\]](#) [\[17\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

20 / 40

The model's request to run a tool, with a name and arguments. Your script, not the model, runs it.

[\[11\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

19 / 40

A function the model can ask your program to run. You describe it with a name, parameters and a docstring.

[\[11\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

22 / 40

A list of permitted values, such as `group_by` must be branch, category or month. Anything else is refused.

[\[15\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

21 / 40

Ask the model, run the tools it requests, send the results back, and repeat until it answers or a step limit is reached.

[\[11\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

24 / 40

Giving a tool only the access it needs, such as read-only.

[\[15\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

TERM

23 / 40

Giving an AI system too much functionality, permission or autonomy.

[\[15\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

25 / 40

Why is Python called the glue for AI?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

26 / 40

What does csv.DictReader give you?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

27 / 40

Why never change the raw data file?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

28 / 40

What does row["loans"] is None check?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

29 / 40

Why use a virtual environment?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

30 / 40

When an agent calls a tool, who runs the code?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

31 / 40

What goes back to the model after a tool runs?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

32 / 40

Why check arguments before running a tool?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

26 / 40

Each row of the CSV as a dictionary keyed by column name.

[\[2\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

25 / 40

Readable Python code steers fast compiled libraries underneath, and the ecosystem covers data, machine learning and models.

[\[1\]](#) [\[6\]](#) [\[7\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

28 / 40

Whether the value is missing. Use is None, not == 0, because zero is a real value.

[\[1\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

27 / 40

So you can start again and audit what cleaning changed. Write a clean copy instead.

[General](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

30 / 40

Your script. The model only produces a tool name and arguments.

[\[11\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

29 / 40

It keeps one project's packages separate from other projects and from the system Python.

[\[3\]](#) [\[4\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

32 / 40

A model can send wrong or unexpected values. An allowlist limits what can happen.

[\[15\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

31 / 40

A message with the tool role containing the result.

[\[11\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

33 / 40

Why put a step limit on the agent loop?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

34 / 40

Name three safety measures for the agent script.

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

35 / 40

What are the three root causes of excessive agency?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

36 / 40

Why is the notes column risky?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

37 / 40

Why not paste a whole table into the prompt?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

38 / 40

What goes wrong if you skip cleaning names?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

39 / 40

Why log every tool call?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

40 / 40

When might you choose pandas over the csv module?

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

34 / 40

For example: read-only narrow tools, allowlisted arguments, a step limit, a log, human approval before writes.

[\[15\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

33 / 40

To stop endless loops and runaway cost. If the limit is reached, stop and say so.

[\[11\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

36 / 40

It is untrusted free text. A hidden instruction in it is an indirect prompt injection.

[\[16\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

35 / 40

Excessive functionality, excessive permissions and excessive autonomy.

[\[15\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

38 / 40

One branch is counted as several groups, so the totals are silently wrong.

[General](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

37 / 40

It may not fit in the context window, costs tokens, and models miscount. Let Python do the arithmetic.

[\[13\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

40 / 40

For larger tables and heavier grouping, with less code. The csv module needs no install.

[\[6\]](#) [\[2\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

QUIZ

39 / 40

To debug, audit what the agent did, and notice misuse.

[\[15\]](#)

Python and AI

PrivacySafe Foundation | CC0 1.0

Reference list

Links were checked on October 27, 2026.

#	SOURCE	WHAT IT SUPPORTS
[1]	Python documentation, The Python Tutorial https://docs.python.org/3/tutorial/	Functions, docstrings, lists, dictionaries, None, modules and imports.
[2]	Python documentation, csv module https://docs.python.org/3/library/csv.html	Reading and writing CSV files, including csv.DictReader.
[3]	Python documentation, venv: creation of virtual environments https://docs.python.org/3/library/venv.html	Creating and activating a virtual environment.
[4]	Python Packaging User Guide, Installing packages using pip and virtual environments https://packaging.python.org/en/latest/guides/installing-using-pip-and-virtual-environments/	Installing packages with pip inside a virtual environment.
[5]	PEP 8, Style Guide for Python Code https://peps.python.org/pep-0008/	Readable code style.
[6]	pandas documentation https://pandas.pydata.org/docs/	DataFrames, grouping, cleaning and summarizing tables.
[7]	NumPy documentation https://numpy.org/doc/stable/	Fast array computing that many data and AI libraries build on.
[8]	scikit-learn documentation https://scikit-learn.org/stable/	Classical machine learning in Python.
[9]	PyTorch documentation https://pytorch.org/docs/stable/index.html	Deep learning framework used to build and run neural networks.
[10]	Hugging Face, Transformers documentation https://huggingface.co/docs/transformers	Loading and running pretrained language models in Python.
[11]	Ollama documentation, Tool calling https://docs.ollama.com/capabilities/tool-calling	Passing Python functions as tools, reading tool_calls, sending tool results with the tool role, and the multi-turn agent loop.
[12]	Ollama Python library (GitHub) https://github.com/ollama/ollama-python	The ollama package, installed with pip.
[13]	Ollama documentation, FAQ https://docs.ollama.com/faq	Local server on 127.0.0.1, default context, and turning off cloud features.
[14]	Ollama library, olmo-3 https://ollama.com/library/olmo-3	Olmo 3 tags and sizes. The page lists tools among its capabilities.
[15]	OWASP GenAI Security Project, LLM06:2025 Excessive Agency https://genai.owasp.org/llmrisk/llm062025-excessive-agency/	Excessive functionality, permissions and autonomy; minimizing tools, least privilege, human approval, logging and rate limiting.
[16]	OWASP GenAI Security Project, LLM01:2025 Prompt Injection https://genai.owasp.org/llmrisk/llm01-prompt-injection/	Direct and indirect prompt injection, including instructions hidden in content a model reads.
[17]	Python documentation, collections module https://docs.python.org/3/library/collections.html	defaultdict, used for grouping.
[18]	Creative Commons, CC0 1.0 Universal https://creativecommons.org/publicdomain/zero/1.0/	The public-domain dedication that covers these lesson materials.