

Finally, it's worth noting, `raw_input` doesn't natively exist in Python 3, but there are some tricky ways to force the interpreter to evaluate `raw_input` in backwards compatible ways. This can be useful for modernizing legacy Python code without rewriting large portions of it. Research on this topic is better left to the reader, as there are lots of fun (and sometimes scary) ways of doing this.

Summary

Python 2 and Python 3 handle input and `raw_input` differently.

In Python 2

- `input(x)` is roughly the same as `eval(raw_input(x))`
- `raw_input()` is preferred, unless the author wants to support evaluating string expressions.
- `eval()` is used to evaluate string expressions.

Standard Library Docs:

- <https://docs.python.org/2/library/functions.html#input>
- https://docs.python.org/2/library/functions.html#raw_input
- <https://docs.python.org/2/library/functions.html#eval>

In Python 3

- Input handles string as a generic string. It does not evaluate the string as a string expression.
- `raw_input` doesn't exist, but with some tricky techniques, it can be supported.
- `eval()` can be used the same as Python 2.

Standard Library Docs:

- <https://docs.python.org/3/library/functions.html#input>
- <https://docs.python.org/3/library/functions.html#eval>