2.G: GLOSSARY

Contributed by Chuck Severance
Clinical Associate Professor (School of Information) at University of Michigan

assignment

A statement that assigns a value to a variable.

concatenate

To join two operands end to end.

comment

Information in a program that is meant for other programmers (or anyone reading the source code) and has no effect on the execution of the program.

evaluate

To simplify an expression by performing the operations in order to yield a single value.

expression

A combination of variables, operators, and values that represents a single result value.

floating point

A type that represents numbers with fractional parts.

integer

A type that represents whole numbers.

keyword

A reserved word that is used by the compiler to parse a program; you cannot use keywords like `if`, `def`, and `while` as variable names.

mnemonic

A memory aid. We often give variables mnemonic names to help us remember what is stored in the variable.

modulus operator

An operator, denoted with a percent sign (`%`), that works on integers and yields the remainder when one number is divided by another.

operand

One of the values on which an operator operates.

operator

A special symbol that represents a simple computation like addition, multiplication, or string concatenation.

rules of precedence

The set of rules governing the order in which expressions involving multiple operators and operands are evaluated.

statement

A section of code that represents a command or action. So far, the statements we have seen are assignments and print expression statement.

string

A type that represents sequences of characters.

type

A category of values. The types we have seen so far are integers (type `int`), floating-point numbers (type `float`), and strings (type `str`).

value

One of the basic units of data, like a number or string, that a program manipulates.

variable

A name that refers to a value.