Python Commands:

Floor division is //

Floor division always rounds down.

Mod operator is %

Mod operator is the remaining amount from floor division

Take 1042/60 for example, your floor is 17, and the remaining amount is 22.

math.ceil( ) finds the ceiling of a number

math.floor( ) finds the floor of a number

to get the absolute value, use function print(abs( ))

int( ) is used to make user input an integer

a string like “86” is not the same as the numeric value 86

Constants are only able to assigned a value once.

It’s a good idea to type constants in all caps with underscores for spaces, like SALES\_TAX\_RATE = 0.06

Breaking down a program into modules is called modularization

Modularization:

Provides abstraction

Helps multiple programmers work on a single problem

Allows you to reuse work more easily

Modularized programs are easier to understand, and enable a programmer to see the big picture.

It’s like simplifying a to-do list to not include every minor detail of a task.

Dividing large programs into modules and giving those modules to individual programmers/teams allows for easier and quicker work

Reliability is the feature of programs that assures you a module has been proven to work.

Think of reusability as:

You don’t invent the plumbing and heating when you build a house, you install pre-developed systems that are proven to work.

Most programs consist of:

A main program or the mainline logic, which contains the basic steps.

The main program then accesses or calls the modules that provide detailed work

Module header includes an identifier and other necessary information.

The module body contains all statements in or details of the logic.

The return statement marks the end of the module, and identifies the point at which control returns back to the program that called it

Modules are named like variables. They follow the same general rules per program.

Module names must start with a letter, no spaces, and should have meaning.

Module names are followed by a set of parentheses

When the main program wants to use the module, it calls it. Modules can call other modules. Modules can also call themselves. Chained calls are only limited by your computer’s memory.

Recursion is when a function (module) calls itself.

You can place any statements within modules:

Input

Processing

Output

You can also include variables.

Localization is the declaration of variables within modules that do not apply outside the module. These data items are considered visible or in scope only within that module. Can also be called local to the module.

Some languages are better or worse at localization.