Data types:

Integer – whole numbers

Float – decimal numbers

String – text, anything side of single or double quotes

Boolean – true or false

Basic functions:

int() – casts data to an integer

float() – casts data to a float

str() – casts data to a string

bool() – casts data to a Boolean

type() – returns the data type of a piece of data

print() – writes a string to the terminal

input() – takes in data from terminal/user

len() – returns the number of items in a list (or characters in a string)

.upper() – CONVERTS STRING TO ALL UPPERCASE

.lower() – converts string to all lowercase

.title() – Converts String to Title Format

.replace() – substitutes a new phrase for every instance of an existing phrase in a string

open() – opens a specified file for a specified operation (r, w, a, x)

.read() – takes text from an opened file and returns it as a string

.write() – takes a string and puts it as text into an opened file

.close() – closes an opened file

json.loads() – takes in a JSON string and converts to python list of dictionaries; must import json

json.dumps() – takes in a python list of dictionaries and converts to JSON string; must import json

Control flow:

if – checks if a condition evaluates to true and executes code inside of it if so

elif – must be after an if; optional; can have multiple; checks an additional condition to the if statement

else – must be after an if or elif; optional; does not check a condition and runs if no other branches did

while loop – repeats a block of code while a given condition evaluates to true

for loop – repeats a block of a code a specified number of times

break – forcefully exits current loop

continue – stops the current loop iteration and moves on to next loop iteration

try – runs some code block that may cause an error. code block stops as soon as error occurs

except – must be after try; runs a code block when a specified error occurs in the try block

else – must be after an except; optional; runs only when the try has no error occur

finally – must be after an except or else; optional; runs always, regardless of if an error occurs or not

Data structures:

list – a single variable that holds multiple pieces of data, each being their own item or element

dictionary – similar to list except each item is a pair with a key and a value

Functions:

```
def function_name(parameters) :
        return value
function_result = function_name(parameters)
```

Imports:

import – used to import python libraries and modules

import [module_name] as [alias name] – can use alias name to reference module in code

from [module_name] import [functions] – best practice; imports only specific functions from the module

__init__.py – needs to be included inside directories when trying to import your own python modules

File I/O:

with open() as [file_alias] – opens a file and will automatically close it when done