Python Programming Reference Sheet

Built-In Data Types & Literals

- Integers: `x = 10`
- Floating Point Numbers: `y = 3.14`
- Strings and Characters: `s = "Hello"`
- Boolean: `is_valid = True`

Working with Strings

- Assignment: `s = "Hello"`
- Concatenation: `s = "Hello" + " World"`
- Comparison: `s1 == s2`, `s1 != s2`
- Construction from other types: `s = str(100)`

Simple Programming Statements

- Constant declaration: `CONSTANT = "Value"`
- Variable declaration: `x = 10`
- Assignment: `x = 20`
- Method call: `print("Hello")`
- Sequence of statements - grouped: `x = 10; y = 20; print(x + y)`

Structured Programming Statements

- If statement: `if condition: # do something`
- Case statement: Python uses `elif` and `else` instead of a case statement.
- While loop: `while condition: # do something`
- Repeat loop: Python uses `for` and `while` loops.
- For loop: `for i in range(10): # do something`

Declaring Methods

- Declare a method with parameters: `def function_name(parameter): # do something`
- Declare a method that returns data: `def function_name(): # do something; return value`
- Pass by reference: All parameters in Python are passed by reference.

Boolean Operators and Other Statements

- Comparison: `==`, `<`, `>`, `!=`, `<=`, `>=`
- Boolean: `and`, `or`, `not`
- Skip an iteration of a loop: `continue`
- End a loop early: `break`
- End a method: `return`

Custom Types

- Classes: `class ClassName: # class body`
- Enumerations: Python uses the `enum` module for creating enumerations.
- Structs: Python uses the `collections.namedtuple()` function for creating simple classes that just contain fields.

Arrays

- Declaration: `arr = [1, 2, 3]`
- Access: `arr[0]`
- Loop with index i: `for i in range(len(arr)): # do something with arr[i]`
- For each loop: `for element in arr: # do something with element`

Programs and Modules

- Creating a program: Python programs are created by writing code in a `.py` file and running it with the Python interpreter.
- Using a class from a library: `from library_name import ClassName`

Other Things

- Reading from Terminal: `input()`
- Writing to Terminal: `print()`
- Comments: `# This is a comment`