

## CSCI 101 Module 2 Python Basics Quick Reference

### Module 2 Video 5: Python Data Types

#### 1. Integers

- Whole numbers without decimals.
- e.g., **x = 5**

#### 2. Float

- Numbers with decimals.
- e.g., **y = 5.0**

#### 3. Strings

- Sequence of characters.
- e.g., **name = "John"**

#### 4. Lists

- Ordered collection of items (can be of different data types).
- e.g., **my\_list = [1, "Alice", 3.5]**

#### 5. Tuples

- Ordered immutable collection of items.
- e.g., **my\_tuple = (1, "Alice", 3.5)**

#### 6. Dictionaries

- Unordered collection of key-value pairs.
- e.g., **my\_dict = {"name": "Alice", "age": 25}**

#### 7. Booleans

- Represents true or false values.
- e.g., **is\_true = True**

### Module 2 Video 8: Python Assignment Operators

1. **=**: Assigns value from right side operands to left side operand (**x = 5**)
2. **+=**: Adds right operand to the left operand and assign the result to left operand (**x += 5**)
3. **-=**: Subtracts right operand from the left operand and assign the result to left operand (**x -= 5**)
4. **\*=**: Multiplies right operand with the left operand and assign the result to left operand (**x \*= 5**)
5. **/=**: Divides left operand with the right operand and assign the result to left operand (**x /= 5**)

6. **%=**: Takes modulus using two operands and assign the result to left operand (**x %= 5**)

#### **Module 2 Video 9: Python Relational Operators**

1. **==**: If the values of two operands are equal, then the condition becomes true.
2. **!=**: If values of two operands are not equal, then condition becomes true.
3. **>**: If the value of left operand is greater than the value of right operand, then condition becomes true.
4. **<**: If the value of left operand is less than the value of right operand, then condition becomes true.
5. **>=**: If the value of left operand is greater than or equal to the value of right operand, then condition becomes true.
6. **<=**: If the value of left operand is less than or equal to the value of right operand, then condition becomes true.

#### **Module 2 Video 10: Python Membership Operators**

1. **in**: Evaluates to true if it finds a variable in the specified sequence and false otherwise.
2. **not in**: Evaluates to true if it does not find a variable in the specified sequence and false otherwise.

#### **Module 2 Video 11: Python Logical Operators**

1. **and**: If both the operands are true then condition becomes true.
2. **or**: If any of the two operands are non-zero then condition becomes true.
3. **not**: Used to invert the value of a boolean expression.

#### **Module 2 Video 12: Python Conditional Statements**

1. **if**: Used to test a condition and execute some statements if the condition is true.
2. **elif**: Used to test another condition if the previous condition was false.
3. **else**: Used to execute some statements if all the previous conditions were false.

#### **Module 2 Video 13: Python For Loop**

1. **for**: Used to iterate over a sequence (like a list, tuple, dictionary, set, or string) or other iterable objects  
for item in iterable:

# your tasks

### **Module 2 Video 14: Python While Loop**

1. **while:** Used to execute a block of statements as long as a condition is true.

while condition:

# your tasks

### **Module 2 Video 15: Python User Defined Function**

1. **def:** Used to define a function in Python.

def function\_name(parameters):

# your tasks

return value