# School of Computer Science https://cs.uwindsor.ca

Master of Applied Computing

# COMP-8347



Internet Applications and Distributed Systems Dr. Adel Abusitta – [adel.abusitta@uwindsor.ca](mailto:adel.abusitta@uwindsor.ca)

**LAB 1 – Introduction to Python**

**NOTE**: Use Python’s **IDLE** interactive tool. Write your answer beside each command in this sheet in **bold**. **Part 1 - Lists in Python**: Given the following two lists:

L1 = [‘university’, 'blue', 29, (4, 99, 186), [(6,'w'), 19, (5,'z')], 'DATA', 'N9E', 21, 'Delta']

L2 = ['openai', 'law', 98, 8.00, 'lion', 'extranet']

## - Work with list indexing and slicing:

Indicate the results if you type the following commands in IDLE: a) L1[2][1] => **TypeError: Int object is not subscriptable**

b) L1[3][0] => **4**

c) L1[4][2][1] => **‘z’**

* + 1. len(L1) => **9**

e) L1[14] => **IndexError: list index out of range**

f) L1[-4:-1] => **['DATA', 'N9E', 21]**

g) L1[2:14] => **[29, (4, 99, 186), [(6, 'w'), 19, (5, 'z')], 'DATA', 'N9E', 21, 'Delta']**

* + - 1. L2+L1 => **['openai', 'law', 98, 8.0, 'lion', 'extranet', 'university', 'blue', 29, (4, 99, 186), [(6, 'w'), 19, (5, 'z')], 'DATA', 'N9E', 21, 'Delta']**
      2. L2\*2 => **['openai', 'law', 98, 8.0, 'lion', 'extranet', 'openai', 'law', 98, 8.0, 'lion', 'extranet']**

j) L1[4][1] = 4 => **['university', 'blue', 29, (4, 99, 186), [(6, 'w'), 4, (5, 'z')], 'DATA', 'N9E', 21, 'Delta']**

k) del L2[-3] => **['openai', 'law', 98, 'lion', 'extranet']**

## - Work with list methods and data types:

Type python commands to do the following:

1. append the string 'ublike' to L1 => **L1.append(‘ublike’)**
2. remove the last element of L2 => **L2.pop()**
3. insert the item 4.8 at index 3 in L1 => **L1.insert(3,4.8)**
4. add the integers in the list [44, 50] at the end of L2 => **L2.extend([44,50])**

**Part 2 - Strings in Python:** Given the following two strings:

s1 = "One should note that IEEE Transactions are extremely selected"

s2 = "There are two areas in cloud computing: performance and security"

***Work with string indexing, slicing, striding, assignment, concatenation***: Indicate the results if you type the following at the Python prompt in IDLE interactive mode. Indicate the type of error if the command is wrong:

a) s1[:9] => **'One shoul'**

b) s2[-1:-4] => **‘’ # empty output as the start is greater than the end index**

c) s2[-2:] => **‘ty’**

d) s2[0:15:2] => **'Teeaetoa'**

e) s1+" "+s2 => **'One should note that IEEE Transactions are extremely selected There are two areas in cloud computing: performance and security'**

***Work with string methods***: Use **str** methods to do the following and indicate the corresponding results:

* 1. Check if the string s2 ends with the word 'security' => **s2.endswith(‘security’)**

=> **True**

* 1. Return a list of words from s2 => **s2.split(' ')**

=> **['There', 'are', 'two', 'areas', 'in', 'cloud', 'computing:', 'performance', 'and', 'security']**

* 1. Convert s1 and s2 to all uppercase letters => **s1.upper(), s2.upper()**

=> **ONE SHOULD NOTE THAT IEEE TRANSACTIONS ARE EXTREMELY SELECTED**

=> **THERE ARE TWO AREAS IN CLOUD COMPUTING: PERFORMANCE AND SECURITY**

* 1. Replace the string 'data' of s2 with empty string => **s2.replace('data', “”)**

=> **There are two areas in cloud computing: performance and security**

* 1. Count the number of times ‘E’ occurs in s1=> **s1.count(‘E’)**

=> **3**

1. **Part 3- Dictionary in Python:** Define the following *dicts*: *#dictionary literals*

d1={"name": "Alex", "age": 35, (4, 'f'):['x', 'y', 'z'], 6: "Canada", 44: 99, 19:555}

*#dictionary using sequences*

d2 = dict([("name","Nancy"), ('age', 44), ((3,4), ['a', 'b', 'c']), (0, 'black'), (33, 67)])

*#dictionary using keywords*

d3 = dict(id=777, name='Michel', siblings=['Fung', 'Martin', 'Richard'])

***Work with dict methods***: Type the following commands at the Python prompt in IDLE interactive mode and indicate the result of each command:

* 1. d1.keys() => **dict\_keys(['name', 'age', (4, 'f'), 6, 44, 19])**
  2. d2.values() => **dict\_values(['Nancy', 44, ['a', 'b', 'c'], 'black', 67])**
  3. d3.get('id') => **777**
  4. d2.get('age') => **44**
  5. d3.get('age') => **‘’ #which is actually None if it’s printed**
  6. d3.get('name', 'Tim') => **'Michel'**
  7. d2.items() => **dict\_items([('name', 'Nancy'), ('age', 44), ((3, 4), ['a', 'b', 'c']), (0, 'black'), (33, 67)])**
  8. d3['siblings'] => **['Fung', 'Martin', 'Richard']**
  9. d2['siblings'] => **KeyError, ‘siblings’ not present**
  10. d2.update(d3) => **d2 updated with d3 elements**
  11. d2['siblings']**\* => ['Fung', 'Martin', 'Richard']**
  12. d2['name']**\*=> ‘Michel’**
  13. d1 == d2 => **false**
  14. len(d2) => **7**
  15. for key in d1.keys(): print(key)
* **name**
* **age**
* **(4, 'f')**
* **6**
* **44**
* **19**
  1. for key in d2.keys(): print(d2[key]) **\***
* **Michel**
* **44**
* **['a', 'b', 'c']**
* **black**
* **67**
* **777**
* **['Fung', ‘Martin', 'Richard']**