## 

## 

**Experiment 2:**

**Strings, Lists, Tuples, and Dictionaries**

CPE106L (Software Design Laboratory)

**Member 1: Cyris Ken M. Alipio**

**Member 2: Neil Emmanuel Macaro**

**Member 3: Paolo Sarmiento**

Group No.: **6**

Section: **B2**

## **PreLab**



|  |
| --- |
| **Readings, Insights, and Reflection** |
| **Fundamentals of Python: First Programs**  **9780357881132**  Alipio, Sarmiento, Macaro  Through our exploration of basic data structures in Python across various chapters and labs, we've discovered the language's remarkable versatility and adaptability. From strings to lists, tuples, and dictionaries, Python equips us with a rich toolkit for efficient data management and manipulation. We've learned that strings serve as versatile data structures, capable of representing character sequences and storing multiple pieces of data, enabling us to read files and manipulate data effectively. Lists, tuples, and dictionaries further extend Python's capabilities, each fulfilling specific roles in data representation and organization. Lists offer dynamic manipulation and access to various literals, while tuples provide ordered and unchangeable data storage. Additionally, dictionaries facilitate efficient data retrieval and organization through key-value pairs. Clear explanations, practical examples, and hands-on exercises have empowered us to harness the power of these fundamental data structures effectively in our Python programming endeavors. |

**Answers to Questions**

1. b. 20

2. b. [20, 30]

3. a. 1

4. b. [10, 20, 30, 40, 50]

5. b. [10, 5, 30]

6. c. [10, 15, 20, 30]

7. b. ["name", "age"]

8. b. None

9. b. pop

10. b.