Nicole Angel B. Lopena

CS3C

Strings

▪ Creating a String

A string in Python is a sequence of characters enclosed within either single quotes (`'`) or double

quotes (`"`). You can create a string by simply assigning a sequence of characters to a variable.

▪ Accessing Characters in the String

You can access individual characters in a string using indexing. Python uses zero-based indexing,

meaning the first character is at index 0, the second at index 1, and so on. You can also use negative

indices to access characters from the end of the string.

▪ Removing Space from a String

You can remove spaces from a string using various methods. One common approach is to use the

`strip()` method, which removes leading and trailing whitespaces by default. If you want to remove

spaces from within the string, you can use `replace()` method or list comprehension.

▪ Python String Methods

Python provides various built-in string methods that allow you to manipulate strings efficiently. Some

commonly used string methods include `upper()`, `lower()`, `split()`, `join()`, `find()`, `replace()`,

`startswith()`, `endswith()`, etc.

1. `upper()`\*\*: Converts all characters in the string to uppercase.

2. `lower()`\*\*: Converts all characters in the string to lowercase.

3. `capitalize()`\*\*: Converts the first character of the string to uppercase and the rest to

lowercase.

4. `title()`\*\*: Converts the first character of each word in the string to uppercase.

5. `strip()`\*\*: Removes leading and trailing whitespace characters from the string.

6. `replace(old, new)`\*\*: Replaces occurrences of the substring `old` with the substring `new` in

the string

Nicole Angel B. Lopena

CS3C

Lunch Jupyter Notebook

Open a Notebook File

Start writing a Jupyter Notebook