# CS PROFESSIONAL

# ELECTIVE

Creating Strings in Python

In Python, you can create strings using single quotes ('), double quotes ("), or triple quotes (single or double, three in a row) for multiline strings:

my\_string = 'Hello, world!' another\_string = "It's a beautiful day. multiline\_string = """This is a string that spans multiple lines.""" print(my\_string) print(another\_string) print(multiline\_string)

This will be the output:

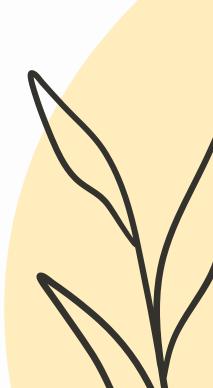
Hello, world! It's a beautiful day. This is a string that spans multiple lines.

Accessing Characters in Strings\* Strings are like sequences of characters. You can access individual characters using their zerobased index within square brackets:

first\_character = my\_string[0] # H last\_character =
my\_string[-1] # ! (using negative indexing) substring =
my\_string[2:7] # llo, w print(first\_character)
print(last\_character) print(substring)

This will be theoutput:

H!llo, w



## Removing Spaces from a String

The replace() method comes in handy for removing spaces: no spaces spaces string = my\_string.replace(" ", "")

print(no\_spaces\_string) # HelloWorld!

### **Exploring Python String Methods**

Python offers a rich set of string methods for various tasks: - \*\*lower():\*\* Converts the string to lowercase:

- lowercase\_string = my\_string.lower() print(lowercase\_string) # hello, world!
- \*\*upper():\*\* Converts the string to uppercase:
- \*\*split():\*\* Splits the string into a list based on a delimiter (e.g., space, comma):
- words = my\_string.split(",") print(words) # ['Hello', ' World!']
- \*\*in:\*\* Checks if a substring is present within the string:
- if "world" in my\_string.lower(): # Case-insensitive check print("World is present")
- \*\*find() and index():\*\* Locate the starting index of a substring (the difference is that find() returns -1 if not found, while index() raises a ValueError):
- world\_index = my\_string.find("world") print(world\_index) # 7
- \*\*startswith() and endswith():\*\* Check if the string starts or ends with a specific substring:
- if my\_string.startswith("Hello"): print("String starts with 'Hello")
  if my\_string.endswith("!"): print("String ends with '!")



### 1. Launch Jupyter Notebook:

- Open your terminal or command prompt.
- Type jupyter notebook and press Enter.
- This will start the Jupyter Notebook server and open a new tab in your default web browser with the Jupyter Dashboard.

#### 2. Open a notebook file:

- Navigate to the directory where your notebook file is located.
- Click on the notebook file with the extension .ipynb.
- This will open the notebook in a new tab in your web browser, allowing you to view and edit it using Jupyter Notebook interface.

#### 3. Start writing a Jupyter Notebook:

- In the Jupyter Dashboard, click on the "New" button in the top right corner.
- Choose "Python 3" (or any other available kernels) from the dropdown menu under "Notebook".
- This will open a new untitled notebook, where you can start writing code and text cells.
- If you want to add content to an existing notebook:
  - Open the notebook file as mentioned in step 2.
  - Click on an empty cell or create a new cell using the "+ button in the toolbar.
  - Type or paste your code or text into the cell and run it using Shift + Enter.