Lab 4: Define Your Own Function

Task: Calculate Series Resistance

Objective

The objective of this task is to teach students how to define their own Python functions with parameters, return values, docstrings, and type annotations. The function will calculate the total resistance of resistors connected in series.

Task Description

The goal of this task is to create a Python function that accepts a list or tuple of resistances, computes the total series resistance, and demonstrates the use of docstrings and type annotations. Students will also learn to access the function's annotations.

Procedure

Defined a function series() that accepts a list or tuple of resistances.

Used the function to calculate the total resistance by summing all values.

Added a docstring explaining the function's purpose and parameters.

Added type annotations to indicate the expected input and output types.

Printed the function's **annotations** to verify the annotations.

In [1]:
```python
def series(resistances: list[float] | tuple[float, ...]) -> float:
    """
    Calculate the total resistance of resistors connected in series.

    Parameters:
        resistances (list[float] or tuple[float]): A list or tuple containing

    Returns:
        float: The total resistance of the series combination.
    """
    return sum(resistances)

R_list = [10, 20, 30, 40, 50]
total_R = series(R_list)

print("Resistance List:", R_list)
print("Total Series Resistance:", total_R, "Ω")

print("\nFunction Annotations:")
print(series.__annotations__)
```

```
Resistance List: [10, 20, 30, 40, 50]
Total Series Resistance: 150 Ω

Function Annotations:
{'resistances': list[float] | tuple[float, ...], 'return': <class 'float'>}
```