Lab Work – List Comprehension Tasks

Objective

The objective of this lab is to strengthen the understanding of Python list comprehensions by applying them to real-world engineering and AI-related data. Through tasks involving voltage filtering, power conversion, resistance analysis, current classification, and sequence generation, students learn to process and transform lists efficiently. This lab helps in building the foundational skills required for writing clean, concise, and optimized code, which is essential for introductory AI programming and data handling.

Task 1: Filter Voltage Readings (< 200 V)

Takes the list of voltages

Selects only values below 200 volts

Stores the filtered values in a new list

In [2]:
```
voltages = [224, 198, 231, 177, 205, 189, 240]
low_voltages = [v for v in voltages if v < 200]
print(low_voltages)
```

```
[198, 177, 189]
```

Task 2: Convert Power Ratings to kW

Takes power values in watts

Converts each value to kilowatts

Creates a new list of converted values

In [3]:
```
powers_w = [50, 120, 300, 750, 1500]
powers_kw = [p / 1000 for p in powers_w]
print(powers_kw)
```

```
[0.05, 0.12, 0.3, 0.75, 1.5]
```

Task 3: Square Only Even Resistances

Checks each resistor value

Selects only even numbers

Squares them and saves the results

In [4]:
```
resistors = [10, 15, 22, 47, 50, 68]
even_squares = [r*r for r in resistors if r % 2 == 0]
```

```
print(even_squares)
```

```
[100, 484, 2500, 4624]
```

Task 4: Label Currents as “High” or “Low”

Reads each current value

Compares it with 1.5 A

Assigns “High” if above 1.5, otherwise “Low”

In [ ]:
```
currents = [2.1, 0.5, 1.8, 3.0, 0.9]
labels = ["High" if c > 1.5 else "Low" for c in currents]
print(labels)
```

Task 5: Extract Appliance Names Starting With ‘F’

Reads appliance names

Selects names beginning with the letter F

In [5]:
```
appliances = ["Fan", "Fridge", "Light", "Freezer", "Pump"]
f_appliances = [a for a in appliances if a.startswith("F")]
print(f_appliances)
```

```
['Fan', 'Fridge', 'Freezer']
```

Task 6: Generate First 10 Multiples of 50

Generates values from 1 to 10

Multiplies each by 50

Produces multiples from 50 to 500

In [7]:
```
multiples_50 = [50*i for i in range(1, 11)]
print(multiples_50)
```

```
[50, 100, 150, 200, 250, 300, 350, 400, 450, 500]
```