Introduction to Python Variables: Assigning Values with Different Data Types

In [ ]:
```
'''
Objective:
1. Assign a string variable and print it.
2. Assign an integer variable and print it.
3. Assign a float variable and print it.
4. Use a boolean variable to store True or False values based on a condition
5. Store multiple values in a list variable.
6. Use a dictionary to store key-value pairs for more complex data.
7 Assign multiple variables at once for concise code.
'''
```

1: Create and Assign String Variables

A string is a sequence of characters. Here, city_name is assigned the string "City A" and printed.

In [1]:
```
# Create a string variable
city_name = "City A"

# Print the value of the string variable
print(f"City Name: {city_name}")
```

```
City Name: City A
```

2: Create and Assign Integer Variables

An integer is a whole number without decimals. We assign the integer 25 to the variable temperature and print it.

In [2]:
```
# Create an integer variable for temperature
temperature = 25

# Print the value of the integer variable
print(f"Temperature: {temperature}°C")
```

```
Temperature: 25°C
```

3: Create and Assign Float (Decimal) Variables

A float is a number that can have a decimal point. Here, carbon_footprint is assigned a floating-point value of 500.75 and printed.

In [3]:
```
# Create a float variable for carbon footprint
carbon_footprint = 500.75  # kg CO2

# Print the value of the float variable
print(f"Carbon Footprint: {carbon_footprint} kg CO2")
```

```
Carbon Footprint: 500.75 kg CO2
```

4: Create and Assign Boolean Variables

A boolean can be either True or False. Here, the variable is_sustainable is assigned based on whether the carbon_footprint is less than 400, which is False in this case.

In [4]:
```
# Create a boolean variable to indicate if the city is sustainable
is_sustainable = carbon_footprint < 400  # This will be False because 500.75

# Print the value of the boolean variable
print(f"Is the city sustainable? {is_sustainable}")
```

```
Is the city sustainable? False
```

5: Create and Assign List Variables

A list is a collection of items (values) stored in a single variable. Here, weekly_temperatures contains a list of integers representing temperatures recorded over a week.

In [5]:
```
# Create a list variable for temperatures recorded in the city over a week
weekly_temperatures = [25, 27, 28, 26, 24, 30, 29]

# Print the list
print(f"Weekly Temperatures: {weekly_temperatures}")
```

```
Weekly Temperatures: [25, 27, 28, 26, 24, 30, 29]
```

6: Create and Assign Dictionary Variables

A dictionary stores data in key-value pairs. Here, city_data holds information about the city, including its name, temperature, carbon footprint, and sustainability status.

In [6]:
```
# Create a dictionary variable to store city data
city_data = {
    "name": "City A",
    "temperature": 25,
    "carbon_footprint": 500.75,
    "is_sustainable": False
}

# Print the dictionary
print(f"City Data: {city_data}")
```

```
City Data: {'name': 'City A', 'temperature': 25, 'carbon_footprint': 500.7
5, 'is_sustainable': False}
```

7: Assigning Multiple Variables at Once

Python allows assigning multiple variables in one line. Here, we assign values to city_name, temperature, and carbon_footprint at the same time.

In [7]:
```
# Assign multiple variables at once
city_name, temperature, carbon_footprint = "City B", 30, 350.50

# Print the values
print(f"City Name: {city_name}, Temperature: {temperature}°C, Carbon Footpri
```

```
City Name: City B, Temperature: 30°C, Carbon Footprint: 350.5 kg CO2
```

Conclusion:

```
Objective 1: Assign a string variable and print it.
Objective 2: Assign an integer variable and print it.
Objective 3: Assign a float variable and print it.
Objective 4: Use a boolean variable to store True or False values b
ased on a condition.
Objective 5: Store multiple values in a list variable.
Objective 6: Use a dictionary to store key-value pairs for more com
plex data.
Objective 7: Assign multiple variables at once for concise code.
```

These code examples in python show us how to create and assign values to different data types, with simple comments explaining each step.