

## Summary of Day 6 and 7

### Day 6: Functions

- **Definition:** A function is a block of reusable code designed to perform a single action. Functions help organize code, make it reusable, and improve readability.

### Key Topics

#### 1. Built-in Functions:

- Functions provided by Python, e.g., `print()`, `len()`, `type()`, `sum()`, etc.
- Example:

```
print("Hello, World!")  
  
print(len([1, 2, 3])) # Outputs 3
```

#### 2. Function Calling:

- A function is called to execute its code.
- Example:

```
def greet():  
    print("Hello!")  
  
greet() # Calls the function
```

#### 3. Parts of a Function:

- **Definition:** Declares the function using `def` keyword.
- **Calling:** Executes the function by its name.

#### 4. Types of Functions:

- **Built-in Functions:** Predefined functions in Python.
- **User-Defined Functions:** Functions created by the programmer.

```
def add(a, b):  
    return a + b  
  
print(add(3, 4)) # Outputs 7
```

#### 5. Parameters and Arguments:

- **Parameters:** Variables specified in the function definition.
- **Arguments:** Values passed to a function when calling it.

```
def greet(name): # Parameter
```

```
print(f"Hello, {name}!")  
greet("Alice") # Argument
```

## 6. Conditional Statements in Functions:

- Using if, elif, and else inside functions for decision-making.

```
def check_even(number):  
    if number % 2 == 0:  
        return "Even"  
    else:  
        return "Odd"  
print(check_even(5)) # Outputs 'Odd'
```

---

## Day 7: Libraries

- **Definition:** Libraries are collections of pre-written code that can be used to perform common tasks. They help save time and effort.
- **Popular Python Libraries:**
  - **Matplotlib:** For data visualization.
  - **NumPy:** For numerical computations.
  - **Pandas:** For data manipulation and analysis.
  - **Requests:** For making HTTP requests.
  - **Django:** For web development.
- **Using Libraries:**
  - Libraries are imported using the import keyword.

Example:

```
import math  
print(math.sqrt(16)) # Outputs 4.0
```

---

## Creating a Simple Graph Using Matplotlib

```
import matplotlib.pyplot as plt  
x = [1, 2, 3, 4, 5]  
y = [2, 4, 6, 8, 10]  
plt.plot(x, y, marker='o', label='y = 2x')  
plt.xlabel('X-axis')  
plt.ylabel('Y-axis')  
plt.title('Simple Line Graph')  
plt.legend()  
plt.show()
```

This code plots a line graph where  $y = 2x$ . The markers ('o') highlight the points on the graph.