Summary of Day 5: Slicing and Repeated Statements

Slicing

- **Definition**: Slicing is a method to extract specific portions of data from sequences such as strings, lists, or tuples.
- **Syntax**: sequence[start:stop:step]
    - start: The index where the slice begins.
    - stop: The index where the slice ends.
    - step: The interval between elements in the slice.

Examples of Slicing:

```
# String slicing

text = "Python"

print(text[1:4])   # Outputs 'yth'

print(text[:3])    # Outputs 'Pyt'

print(text[::2])   # Outputs 'Pto'


# List slicing

numbers = [10, 20, 30, 40, 50]

print(numbers[1:4])  # Outputs [20, 30, 40]

print(numbers[::-1]) # Outputs [50, 40, 30, 20, 10]
```

Repeated Statements

For Loop

- **Definition**: Used to iterate over a sequence (like a list, tuple, dictionary, set, or string) or a range of numbers.
- **Syntax**:

```
for item in sequence:
```

While Loop

- **Definition**: Executes as long as a specified condition is True.
- **Syntax**:

```
while condition:
```

Examples Using Conditional Statements and Loops

Example 1: Using a For Loop with Conditional Statement

```
# Print even numbers from a list
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
for num in numbers:
    if num % 2 == 0:
        print(f"{num} is even.")
```

Example 2: Using a While Loop with Conditional Statement

```
# Print numbers from 1 to 5
count = 1
while count <= 5:
    print(f"Count is: {count}")
    count += 1
```