Conditional Statements

1. If-Else Statements

Conditional statements allow your code to make decisions based on conditions.

```
score = 85

if score >= 90:
    grade = 'A'
elif score >= 80:
    grade = 'B'
elif score >= 70:
    grade = 'C'
elif score >= 60:
    grade = 'D'
else:
    grade = 'F'

print(f"Score: {score}, Grade: {grade}")
```

```
Score: 85, Grade: B
```

2. Ternary Operator

A concise way to write simple if-else statements in one line.

```
score = 85

# Ternary operator
status = "Pass" if score >= 60 else "Fail"
print(f"Status: {status}")

# Traditional if-else equivalent
if score >= 60:
    status = "Pass"
else:
    status = "Fail"
print(f"Status: {status}")
```

```
Status: Pass
Status: Pass
```

3. Multiple Conditions

Combine multiple conditions using logical operators (and, or, not).

```
age = 25
has_license = True

if age >= 18 and has_license:
    print("Can drive")
elif age >= 18 and not has_license:
    print("Need license")
else:
    print("Too young")
```

```
Can drive
```

4. Checking Membership

Use "in" and "not in" to check if an element exists in a collection.

```
numbers = [1, 2, 3, 4, 5]

# Check if element is in list
if 3 in numbers:
    print("3 is in the list")

if 10 not in numbers:
    print("10 is not in the list")

# Works with strings too
text = "Hello, World!"
if "World" in text:
    print("Found 'World' in text")
```

```
3 is in the list
10 is not in the list
Found 'World' in text
```

5. Nested Conditionals

You can nest if statements inside other if statements for more complex logic.

```
age = 25
income = 50000

if age >= 18:
    if income >= 30000:
        print("Eligible for loan")
    else:
        print("Income too low")
else:
    print("Too young for loan")
```

```
Eligible for loan
```

Key Takeaways

- if-elif-else for multiple conditions
- Ternary operator for simple conditional assignments
- Use and, or, not to combine conditions
- "in" and "not in" for membership testing
- Indentation is crucial in Python