Computer Programming Summer Interest Group

Session 3 / Conditionals

6/18/2025

Joshua 24:15

And if it is evil in your eyes to serve the Lord, choose this day whom you will serve, whether the gods your fathers served in the region beyond the River, or the gods of the Amorites in whose land you dwell. But as for me and my house, we will serve the Lord.
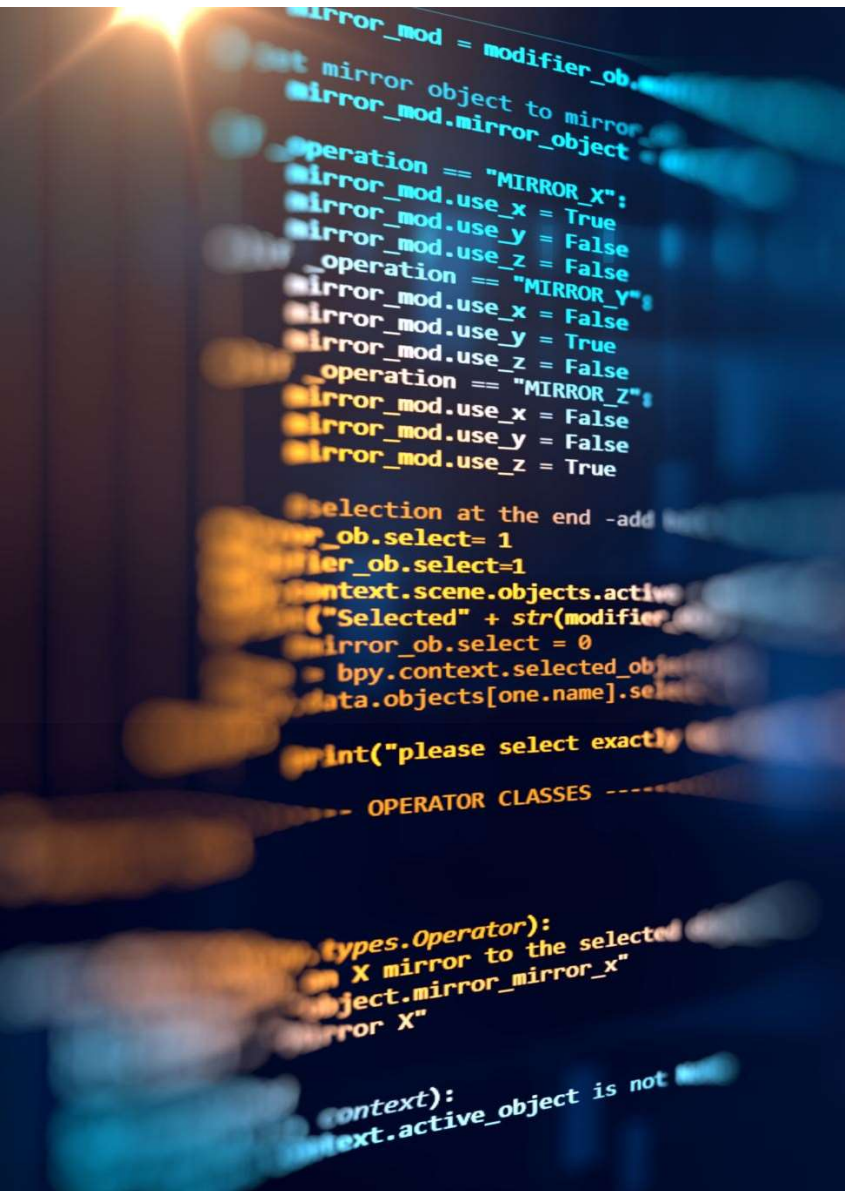
Objectives

- By the end of this session, you will be able to:
  - Understand how programs can make decisions
  - Use if, elif, and else to implement conditional logic
  - Apply comparison and logical operators in conditions
  - Write simple Python scripts that respond to user input and follow multiple execution paths
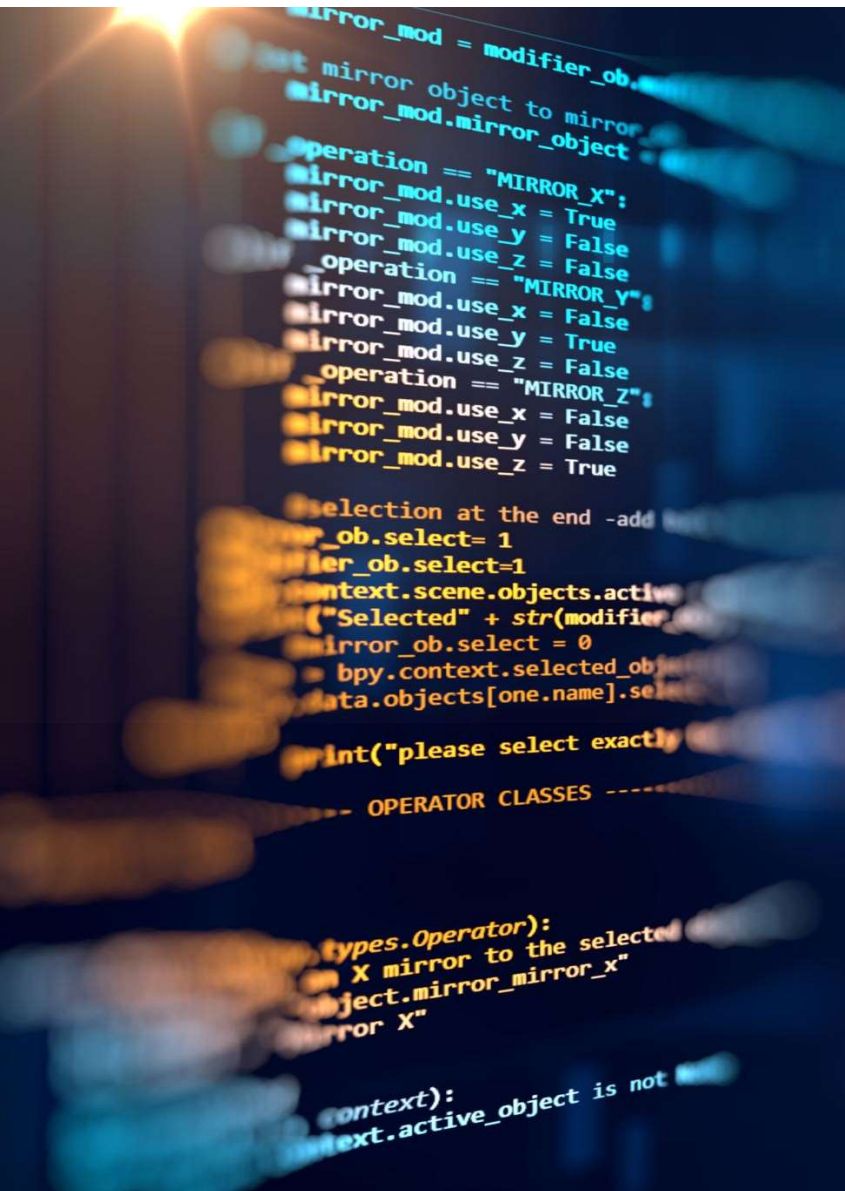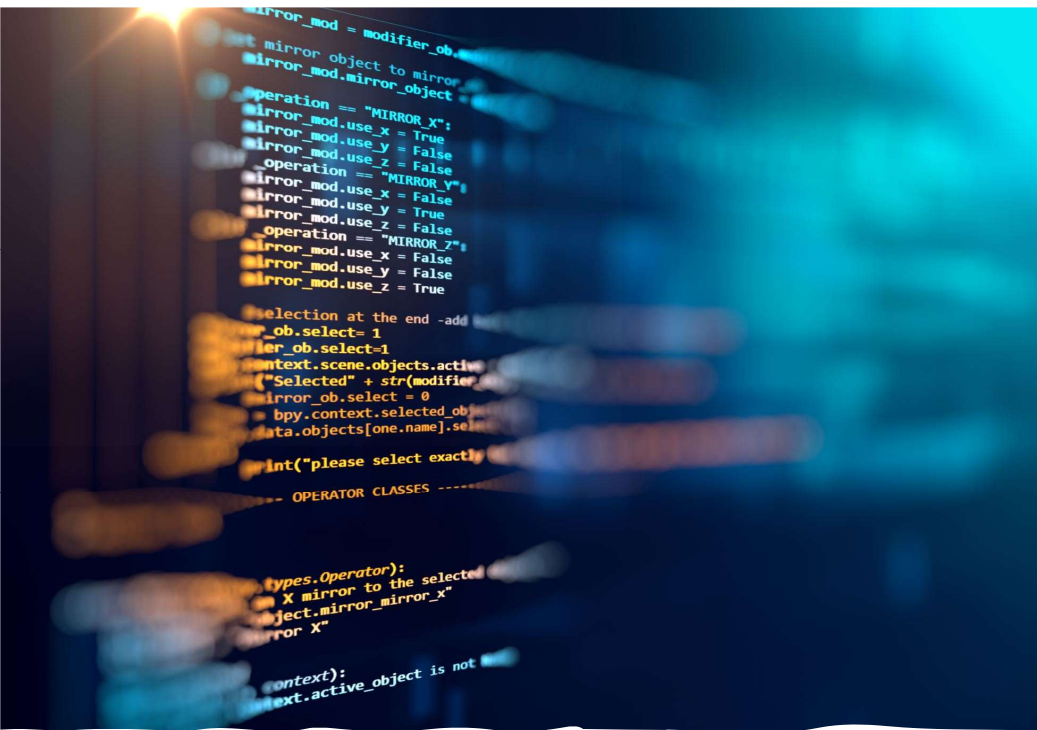
Sequential Flow

- No decisions

Sequential Flow

- No decisions

Conditional Flow

- Make a decision to do … one thing … or some other thing

Python if, elif and else statements

- If the first condition is True, run the "do something" code block

- If the first condition is False but the second is True, run the "do something else" code block

- If neither are True, run the "fallback" code block

- The words `if`, `elif` and `else` are known as ***reserved words*** in Python… they have special meaning
  - You can't use them for variable names

```python
if <condition>:
    # do something
elif <other_condition>:
    # do something else
else:
    # fallback
```

Python if, elif and else statements

- A conditional requires the `if` statement to occur first
- The `elif` and `else` are both optional
- You can have multiple `elif`'s between the `if` and `else` portions
- The end of the line with the `if`, `elif` or `else` statement must have a colon (':')
- The code block in each section must be indented
  - Python uses indentation to decide how to structure a block of code.
  - Most programming languages do no use indentation to structure code

```python
if <condition>:
    # do something
elif <other_condition>:
    # do something else
else:
    # fallback
```

Logical Operators

Operator	Meaning	Example
==	Equal To	`if x == 9:`
!=	Not Equal To	`if x != 81:`
>	Greater Than	`if s > t:`
<	Less Than	`if u < v:`
>=	Greater Than or Equal To	`if a >= b:`
<=	Less Than or Equal To	`if e <= pi:`
and	Logical 'and'	`if s > t and u < v:`
or	Logical 'or'	`if m < r or f >= 90:`
not	Logical 'not'	`if not x == 81:`

The order is important

- Notice in this sequence we go from the highest to lowest, always checking if the score is larger than some breakoff value

- We could have checked from lowest to highest, but we would have had to check that the values were less than the breakoff

A	>= 90
B	>= 80
C	>= 70
D	>= 60
F	

F	<= 60
D	<= 70
C	<= 80
B	<= 90
A	

```python
score = float(input("Enter score: "))

if score >= 90:
    print("A")
elif score >= 80:
    print("B")
elif score >= 70:
    print("C")
elif score >= 60:
    print("D")
else:
    print("F")
```

Activity

- Write a program that
  - Requests the user to enter a temperature in degrees Fahrenheit (should be a float value)
  - If the temperature is below 40, tell the user to wear a coat
  - If the temperature is less than 70 but not less than 40, tell the user to wear a sweater
  - Otherwise tell the user it's T-shirt weather.