CSCI 141
Scott Wehrwein
Nested and Chained Conditional Statements
Goals

• Understand how conditional statements can be **nested** to make decisions among more than two possibilities.

• Know how to use **if/elif/else** statements.
Nested Conditionals

If/else lets you choose between two options.

What if there are more than two possibilities?

```python
# assume x and y are numbers
if x < y:
    print("x is less than y")
else:
    if x > y:
        print("x is greater than y")
    else:
        print("x and y must be equal")
```

Note: the conditions still have to be boolean expressions (i.e., they evaluate to True or False)

the inner if/else statement is the indented code block for the else clause of the outer if/else statement.
Chained Conditionals: Syntax

- **elif** keyword

  ```python
  if isRaining and not isWindy:
      print("Bring an umbrella!")
  elif isRaining and isWindy:
      print("Wear a raincoat!")
  else:
      print("No rain gear needed!"
  ```

  - An indented code block to be executed if none of the prior conditions was true and this **elif** condition is True
  - (an **else** clause is optional)
Demo

Task: Write a program to ask the user for their 141 section number and print out when their lab section happens.

```python
>>> %Run section_times.py
    Enter your CSCI 141 section number: 20770
    Your lab is on Tuesday from 10 - 12.
>>> 
```
Chained Conditionals: Demo

- sections.py: with nested if/else statements
- sections_elseif.py: with if/elif/else
- sections_refactored.py: refactored to set variables then call print once