



















Question

•What's the difference between employing nested decision making and a logical AND?

Decision-Making With Multiple Alternatives/Questions

James Tam

- IF (single question)
 - Checks a condition and executes a body if the condition is true
- IF-ELSE (single question)
 - Checks a condition and executes one body of code if the condition is true and another body if the condition is false
- Approaches for multiple (two or more) questions
 - Multiple IF's
 - IF-ELIF-ELSE





















Extra Practice (2)		
Number	Roman Numeral	
1	1	
2	П	
3	III	
4	IV	
5	V	
6	VI	
7	VII	
8	VIII	
9	IX	
10	X	

Recap: What Decision Making Mechanisms Are Available /When To Use Them

Mechanism	When To Use
If	Evaluate a Boolean expression and execute some code (body) if it's true
If-else	Evaluate a Boolean expression and execute some code (first body: 'if') if it's true, execute alternate code (second body: 'else') if it's false
Multiple if's	Multiple Boolean expressions need to be evaluated with the answer for each expression being independent of the answers for the others (non-exclusive). Separate instructions (bodies) can be executed for each expression.
If-elif- else	Multiple Boolean expressions need to be evaluated but zero or at most only one of them can be true (mutually exclusive). Zero bodies or exactly one body will execute. Also it allows for a separate body (else-case) to execute when all the if-elif Boolean expressions are false.

Recap: When To Use Compound And Nested Decision Making

Mechanism	When To Use
Compound decision making	There may have to be more than one condition to be considered before the body can execute. All expressions must evaluate to true (AND) or at least one expression must evaluate to true (OR).
Nested decision making	The outer Boolean expression ("gate keeper") must be true before the inner expression will even be evaluated. (Inner Boolean expression is part of the body of the outer Boolean expression).



Testing Decisions: An Example

Program name: 10testing_example.py

Learning objective of example: illustrating an example of the *minimum* number of test cases that should be run for a condition that tests a numeric value.

```
num = int(input("Type in a value for num: "))
if (num >= 0):
    print("Num is non-negative. ")
else:
    print("Num is negative. ")
```

Lesson: Avoid Using A Float When An Integer Will Do

James Tam

James Tam

Program name: 11real_test.py

Learning objective of example: illustrating the imprecise storage mechanism used for floating point variables.

num = 1.0 - 0.55
if (num == 0.45):
 print("Forty five")
else:

print("Not forty five")

[csl branches 13]> python real_test.py Not forty five

<u>Epsilon</u>

- •Because floating point numbers are only approximations of real numbers when performing a comparison "seeing if two numbers are 'close' to each other" sometimes an Epsilon is used instead of zero.
- •Epsilon is a very small number.
- If the absolute difference between the numbers is less than the Epsilon then the numbers are pretty close to each other (likely equal).











Extra Practice: Grades

```
    Write a program that converts percentages to one of the

following letter grades: A (90 – 100%), B (80 – 89%), C (70 –
79%), D (60-69%), F (0-59%).
  # First approach
   if ((percentage <= 100) or (percentage >= 90)):
       letter = 'A'
  elif ((percentage <= 89) or (percentage >= 80)):
       letter = 'B'
   Etc.
  # Second approach
   if ((percentage <= 100) and (percentage >= 90)):
       letter = 'A'
  elif ((percentage <= 89) and (percentage >= 80)):
       letter = 'B'
   Etc.
                                                                James Tam
```



Decision Making: Checking Matches (2)

Example:

```
(String):
if "the" in ("thetheretheir"):
    print("the is a sub-string of thetheretheir ")
else:
    print("not sub-string")
answer = input("Selection: ")
if answer in ("one two seven"):
    print("selection taken")
else:
    print("selection taken")
else:
    print("selection available")
(Numeric):
if num in (1, 2, 3):
    print("in set")
```

```
<text><text><text><code-block></code>
```





Copyright Notification

• "Unless otherwise indicated, all images in this presentation are used with permission from Microsoft."