

Selection in Programming and Python



Admin



- Quiz
- Reading: read chapter 5 in the book with these slides

Don't always do the same thing



- Sometimes we don't want to do the same thing every time
 - Do you come to comp151 every day at this time?
 - How do you know when to come here?
 - Lucky volunteer?

Don't always do the same thing



- Sometimes we don't want to do the same thing every time
 - Do you come to comp151 every day at this time?
 - How do you know when to come here?
 - Lucky volunteer?
 - If the day is (fill in today) then come in,
 - If the day is Saturday, don't come in
 - Holidays etc?

Selection



- The programming language agnostic term for doing something only if some condition is true is **selection**
 - As in we select this code only in this case.
- Selection in python starts with the keyword **if**
- Syntax of basic if statement
- **if** <condition>:
 - do this code block if <condition> was True
- Word on BNF and <stuff> format

condition



- The condition can be anything that evaluates to a **boolean** value
 - Fancy comp sci/math way of saying a value that can be either True or can be False (but nothing else)
 - So what kinds of conditions might we have?
 - What are easy things you can think of that we might check?

condition



- The condition can be anything that evaluates to a **boolean** value
 - Fancy comp sci/math way of saying a value that can be either True or can be False (but nothing else)
 - So what kinds of conditions might we have?
 - What are easy things you can think of that we might check?
 - Common easy ones in CS
 - Check equality between two values
 - Check inequality (is age greater than 21 well then you can get into the bar)
 - Check function to ask about the world
 - Is the stove on?

Equality and Assignment



- What does = mean in python?

Equality and Assignment



- What does = mean in python?
 - Assignment
- `age = int(input("How old are you"))`
 - The thing on the right is stuffed into the variable on the left.
- Equality in python is
 - ==
- `if course == "comp151":`
 - `print("you are here")`

Equalities and Inequalities



- In Python
 - Equality
 - `==`
 - Greater than
 - `>`
 - Less than
 - `<`
 - Not equal
 - `!=`
- Greater than or equal to
 - `>=`
- Less than or equal to
 - `<=`

If example 2



- Inequality example
-
- Notice that like for, if has a **code block**,
 - every line indented from the if line is executed if and only if (iff) the condition was true.

```
age = int(input("How old are you"))  
if age >= 21:  
    print("welcome to the Poker Table")  
    print("come as you are")
```

Lists, for-each, and if



- Lets put the last couple of weeks together,
 - Take the list of numbers (years that the US began a recession) that you find here and
 - Count how many were election years (evenly divisible by 4)
 - So for-each through the list, then if the year is evenly divisible by four add one to total_election_years.
 - Finally, print out the number
 - Lets work through this one together.

```
us_recession_start_years=[1920, 1923, 1926, 1929, 1937,  
1945, 1949, 1953, 1958, 1960, 1969, 1973, 1980, 1981,  
1990, 2001, 2008, 2020]  
total_election_years = 0
```

Checking for values in/not in a list



- Python has a “pretend it’s English” approach to checking if something is in a list.
-
-
- Also for checking if something is not in the list
- ```
courses = ["comp151", "comp152", "math130", "math120"]
```
- ```
if "comp250" in courses:
```

```
    print("yes it is there")
```
-
-
- ```
if "comp250" not in courses:
```

```
 print("You haven't taken Data Structures yet.")
```
-

# Comparing text



- Oh those troublesome users!
  - Users unpredictable. Type yes to continue.
  - Underneath it all computers just manipulate what?
  - Lucky volunteer?

# Comparing text



- Oh those troublesome users!
  - Users unpredictable. Type yes to continue.
  - Underneath it all computers just manipulate what?
  - Numbers. And the numbers that associate with a capital letter character is different than its lower case equivalent.

|    |   |     |   |
|----|---|-----|---|
| 65 | A | 97  | a |
| 66 | B | 98  | b |
| 67 | C | 99  | c |
| 68 | D | 100 | d |
| 69 | E | 101 | e |
| 70 | F | 102 | f |
| 71 | G | 103 | g |
| 72 | H | 104 | h |
| 73 | I | 105 | i |
| 74 | J | 106 | j |
| 75 | K | 107 | k |
| 76 | L | 108 | l |
| 77 | M | 109 | m |
| 78 | N | 110 | n |
| 79 | O | 111 | o |
| 80 | P | 112 | p |
| 81 | Q | 113 | q |

## Comparing text II



- What will happen when running the code to the right?
  - 
  - 
  - 
  - How could we adjust it to be what we want?
- `answer = "Yes"`
  - `if answer == "yes"`
    - `print("you were right")`

## Comparing Text III



- Ask the string to lower case itself and compare downcase
- `answer = input("enter yes to continue")`
- `if answer.lower() == "yes":`
  - `print("we will keep going")`

# If-else



- Sometimes we want to do one thing if the condition is true, but another if the condition is false
  - In python use if-else

```
answer = input("would you like to continue:")
if answer.lower() == "yes":
 print("we will continue")
else:
 print("ok we will quit the program")
 exit(0)
```