



If Statements (Chapter 5)

```
In [1]: # if statements control conditional code execution
#     executes a block of code if the condition is True

if True:
    print("This is printed!")

if False:
    print("But not this")

# rather than use True and False directly, we instead use conditional statements that evaluate
# to either True or False

weight = 80
if weight > 50: # this is True
    print("Warning, 2 person lift")
if weight > 1000: # this is False
    print("Warning, machine lift only")
```

```
This is printed!
Warning, 2 person lift
```

Conditional Tests

Statements that evaluate to True or False, can be used with `if` statements to control code execution

```
In [2]: # check for equality
car = 'bmw' # assign the value 'bmw' to the variable car
car == 'bmw' # conditional test -> is the value of car equal to 'bmw'?

# check for inequality
```

```
car = 'bmw' # assignment
car != 'Audi' # conditional test -> is the value of car not equal to 'Audi'?

# string comparison is case sensitive
car == 'BMW' # False
car.upper() == 'BMW' # True
```

Out[2]: True

In [8]: # Numerical comparison

```
x = 5 # assignment NOT A CONDITIONAL TEST!

print(x > 4) # greater than

print(x >= 4) # greater than or equal to

x < 4 # Less than
x <= 4 # Less than or equal to
x == 4 # equality
x != 4 # inequality
```

True

True

Out[8]: True

In [4]: # Checking for value in a list

```
toppings = ['pepperoni','mushrooms','olives']

'sardines' in toppings # False

'mushrooms' in toppings # True

# Checking for value not in a list

'sardines' not in toppings # True
```

Out[4]: True

If Statement

```
In [9]: # TODO: write an if statement to conditional print a message for each of the test types above
```

```
# Equality
x = 'blue'
if x == 'blue':
    print('is blue')
# Numerical Comparison
y = 15
if y<20:
    print('y<20')

# Membership in a List
z = ['dog', 'cat', 'fish']
if 'dog' in z:
    print('Yay!')
```

```
is blue
y<20
Yay!
```

```
In [10]: # Checking multiple conditions
```

```
# and requires both conditions to be True
x = 5
y = 6
if x==5 and y==6:
    print("both conditions must be True")

# or requires only one condition to be True
if x==5 or y==6:
    print("either or both conditions are True")

# can combine multiple conditions with parenthesis
z = 10
if (x==5 or y==6) and z == 10:
    print("z condition is met and either x or y condition is met")
```

```
both conditions must be True
either or both conditions are True
z condition is met and either x or y condition is met
```

```
In [11]: # if-else statements
```

```
# else means "if not true" or "otherwise"
```

```
x=84
```

```

if x>10:
    print("x is big")
else:           # NOTE: else has no conditional statement!
    print("x is small")

# if you need an additional conditional statement use an elif clause
x=5
if x>10:
    print("x is big")
elif x>2:
    print("x is small")
else:
    print("x is tiny")

# elif clauses do not have to test the same variable
y=6
if x>10:
    print("x is big")
elif x+y > 10:
    print("x is small but x+y is big")
elif y>5:
    print("change x and y so this line prints") #TODO

# else is always optional

```

```

x is big
x is small
x is small but x+y is big

```

In [22]:

```

# multiple if statements versus if-elif chain
x = 5

print("multiple if statements")
if x>4:
    print("x > 4")
if x>3:
    print("x > 3")
if x>2:
    print("x > 2")

print("single if-elif chain")
if x>4:          #NOTE: place most restrictive clause first
    print("x > 4")
elif x>3:
    print("x > 3")

```

```
elif x>2:  
    print("x > 2")
```

multiple if statements

x > 4

x > 3

x > 2

single if-elif chain

x > 4

if statements with lists

```
In [12]: # use the in operator to check for membership in a list  
vowels = ['a','e','i','o','u']
```

```
letter = 'b'  
if letter in vowels:  
    print(f"{letter} is a vowel")  
  
if letter not in vowels:  
    print(f"{letter} is not a vowel")
```

```
# empty lists are False
```

```
favorite_veggies = []  
if favorite_veggies:  
    for veggie in favorite_veggies:  
        print(veggie)  
else:  
    print("It looks like you don't like veggies")
```

b is not a vowel

It looks like you don't like veggies

```
In [23]: # a more complicated example using multiple lists
```

```
available_toppings = ['mushrooms','olives','green peppers','pepperoni','pineapple','extra cheese']  
requested_toppings = ['mushrooms','french fries', 'extra cheese']  
  
# TODO: For each requested topping either print "adding {topping}" or "we don't have {topping}"  
i=0  
while i<3:  
    if requested_toppings[i] in available_toppings:  
        print(f'adding {requested_toppings[i]}')
```

```
        else:
            print(f"we don't have {requested_toppings[i]}")
            i = i+1
# if the requested topping is not the available_toppings list
```

```
adding mushrooms
we don't have french fries
adding extra cheese
```

Helpful Hints

```
In [29]: # shortcut for testing numerical value within a range:
x=6
if 2<x<10:
    print("x is between 2 and 10")
# this is the same as
if 2<x and x<10:
    print("same test")

# you can use the in operator with strings
secret_word = "python"
guess = "p"
if guess in secret_word:
    print(f"There is a {guess} in the secret word")
```

```
x is between 2 and 10
same test
There is a p in the secret word
```

Homework Problems

5-1. Conditional Tests: Write **ten** conditional tests, half (5) should evaluate to **True** and the other half should evaluate to **False**. Print a statement describing each test and your prediction for the results of each test. Your code should look something like this:

```
car = 'subaru'
print("Is car =='subaru'? I predict True.")
print(car == 'subaru')
```

```
print("\nIs car =='audi'? I predict False.")  
print(car == 'audi')
```

```
In [3]: favorite_animal = 'dog'  
print("Is favorite_animal == 'dog'? I predict True.")  
print(favorite_animal == 'dog')  
  
color = 'blue'  
print("Is color == 'orange'? I predict False.")  
print(color == 'orange')  
  
drink = 'soda'  
print("Is drink == 'water'? I predict False.")  
print(drink == 'water')  
  
day = 'Friday'  
print("Is day == 'Friday'? I predict True.")  
print(day == 'Friday')  
  
name = 'Erin'  
print("Is name == 'Erin'? I predict True.")  
print(name == 'Erin')  
  
season = 'Fall'  
print("Is season == 'winter'? I predict False.")  
print(season == 'winter')  
  
state = 'GA'  
print("Is state == 'GA'? I predict True.")  
print(state == 'GA')  
  
food = 'Pizza'  
print("Is food == 'burger'? I predict False.")  
print(food == 'burger')  
  
energy = 'low'  
print("Is energy == 'high'? I predict False.")  
print(energy == 'high')  
  
month = 'Sep'  
print("Is month == 'Sep'? I predict True.")  
print(month == 'Sep')
```

```
Is favorite_animal == 'dog'? I predict True.  
True  
Is color == 'orange'? I predict False.  
False  
Is drink == 'water'? I predict False.  
False  
Is day == 'Friday'? I predict True.  
True  
Is name == 'Erin'? I predict True.  
True  
Is season == 'winter'? I predict False.  
False  
Is state == 'GA'? I predict True.  
True  
Is food == 'burger'? I predict False.  
False  
Is energy == 'high'? I predict False.  
False  
Is month == 'Sep'? I predict True.  
True
```

5-2. More Conditional Tests: Write at least one test for each of the following conditions:

- Tests for equality and inequality with strings
- Tests using the `lower()` method
- Numerical test involving equality, greater than, less than, etc.
- Tests using the `and` keyword and the `or` keyword (use both in the same statement)
- Test whether an item is in a list
- Test whether an item is not in a list

```
In [8]: color = 'red'  
if color == 'red' or color == 'blue' or color == 'yellow':  
    print("It's a primary color!")  
elif color == 'green' or color == 'orange' or color == 'purple':  
    print("It's a secondary color!")  
else:  
    print("It's not a primary or secondary color.")  
  
number = '5'  
if number == '5':  
    print("The number is 5.")
```

```

if number != '5':
    print("The number is not 5.")

word = 'GIANT'
if word == 'giant':
    print("The word is actually small.")
if word.lower() == 'giant':
    print("The word is not completely small.")

new_number = 15
if 10<new_number<20:
    print("The number is in the teens.")
if new_number != 13:
    print("The number is not unlucky")
if new_number == 15:
    print('The number is 15!')

season = 'Fall'
day = 'Friday'
if (season == 'Fall') and (day == 'Friday' or day == 'Saturday'):
    print("It's a good day!")
else:
    print("It's an ok day.")

best_days = ['Friday', 'Saturday', 'Sunday']
if 'Saturday' in best_days:
    print('Hooray!')

favorite_nums = ['4', '5', '6', '8']
if '3' not in favorite_nums:
    print("I don't like this number")

```

It's a primary color!
 The number is 5.
 The word is not completely small.
 The number is in the teens.
 The number is not unlucky
 The number is 15!
 It's a good day!
 Hooray!
 I don't like this number

5-3. Alien Colors #1: Imagine an alien was just shot down in a game. Create a variable called `alien_color` and assign it a value of `'green'`, `'yellow'`, or `'red'`.

- Write an `if` statement to test whether the alien's color is green. If it is, print a message that the player just earned 5 points.
- Change the value of `alien_color` so the `if` test fails. (The version that fails will have no output)

```
In [10]: alien_color = 'yellow'  
if alien_color == 'green':  
    print("you earned 5 points.")
```

5-4. Alien Colors #2:

Set `alien_color` a value of your choice, then write an `if-else` chain as follows:

- If the alien's color is green, print a statement that the player just earned 5 points for shooting the alien.
- If the alien's color isn't green, print a statement that the player just earned 10 points.
- Change the value of `alien_color` and make sure you understand the output.

```
In [12]: alien_color = 'yellow'  
if alien_color == 'green':  
    print("you earned 5 points for shooting the alien")  
elif alien_color != 'green':  
    print("you earned 10 points")
```

you earned 10 points

5-5. Alien Colors #3:

Turn the `if-else` chain above into an `if-elif-else` chain as follows:

- If the alien is green, print a message that the player earned 5 points.
- If the alien is yellow, print a message that the player earned 10 points.
- If the alien is red, print a message that the player earned 15 points.
- Change the value of `alien_color` and run the code multiple times. Make sure you understand the output

```
In [16]: alien_color = 'green'  
if alien_color == 'green':  
    print("you earned 5 points")  
elif alien_color == 'yellow':  
    print("you earned 10 points")  
elif alien_color == 'red':  
    print("you earned 15 points")
```

you earned 5 points

5-6. States of Life:

Write an `if-elif-else` chain that determines a person's stage of life. Set a value for the variable `age`, and then:

- If the person is less than 2 years old, print a message the the person is a baby.
- If the person is at least 2 years old but less than 4, print a message that the person is a toddler.
- If the person is at least 4 years old but less than 13, pint a message that the person is a kid.
- If the person is at least 13 years old but less than 20, print a message that the person is a teenager.
- If the person is at least 20 years old but less than 65, print a message that the person is an adult.
- If the person is age 65 or older, print a message that the person is an elder.

```
In [17]: age = 20
if age < 2:
    print('a baby')
elif 2<= age <4:
    print('a toddler')
elif 4<= age <13:
    print('a kid')
elif 13<= age <20:
    print('a teenager')
elif 20<= age < 65:
    print('an adult')
else:
    print('an elder')
```

an adult

5-7. Favorite Fruit: Make a list of your favorite fruits (at least 4 items). Then write at least three independent `if` statements that check for certain fruits in your list. If the fruit is in your list, the `if` block should print a statement, such as *You really like bananas!*

```
In [20]: fruits = ['apple','watermelon','peach','grape']
if 'apple' in fruits:
    print('You really like apples!')
if 'watermelon' in fruits:
    print('you really like watermelon!')
#to check if an item is not in a list must use NOT IN
if 'banana' not in fruits:
    print('You must not like bananas.')
```

You really like apples!
you really like watermelon!
You must not like bananas.

5-8. Hello Admin: Make a list of five usernames, including the name '`'admin'`'. Imagine you are writing code that will print a greeting to each user after they log in to a websie. Loop through the list, and print a greeting to each user:

- If the username is 'admin' , print a speacial greeting, such as *Hello admin, would you like to see a status report?*
- Otherwise, print a generic greeting, such as *Hello Jaden, thank you for logging in again.*

```
In [23]: users = ['admin', 'Erin', 'Josh', 'Avery', 'Leah']
i = 0
while i<len(users):
    if users[i] == 'admin':
        print(f"Hello {users[i]}, would you like to see a status report?")
    else:
        print(f"Hello {users[i]}, thank you for logging in again")

    i=i+1
```

```
Hello admin, would you like to see a status report?
Hello Erin, thank you for logging in again
Hello Josh, thank you for logging in again
Hello Avery, thank you for logging in again
Hello Leah, thank you for logging in again
```

5-9. No Users: Copy the code above and add an `if` test to make sure the list of users is not empty.

- If the list is empty, print the message *We need to find some users!*
- Remove all of the usernames from your list, and make sure the correct message is printed

```
In [27]: users = []
if users == []:
    print("We need to find some users!")
## NEED THIS IF STATEMENT BEFORE THE WHILE LOOP BECAUSE CANNOT BEGIN WHILE LOOP ITERATION IF LIST DNE
i = 0
while i<len(users):
    if users[i] == 'admin':
        print(f"Hello {users[i]}, would you like to see a status report?")
    else:
        print(f"Hello {users[i]}, thank you for logging in again")

    i=i+1
```

```
We need to find some users!
```

5-10. Checking Usernames: Do the following to create a program that simulates how websites ensure that everyone has a unique username.

- Make a list of five or more usernames called `current_users` .

- Make another list of five usernames called `new_users`. Make sure one or two of the new usernames are also in the `current_users` list.
- Loop through the `new_users` list to see if each new username has already been used. If it has, print a message that the person will need to enter a new username. If a username has not been used, print a message saying that the username is available.
- Make sure your comparison is case insensitive. If `'John'` has been used, `'JOHN'` should not be accepted.

```
In [29]: current_users = ['admin', 'Erin', 'Josh', 'Avery', 'Leah']
new_users = ['Bob', 'Joe', 'Erin', 'Betty', 'Sue']

i = 0
while i < len(current_users):
    if current_users[i] in new_users:
        print(f"Please enter a different username, {current_users[i]} is not available.")
    else:
        print(f"The username {current_users[i]} is available!")
    i = i + 1
```

The username admin is available!
 Please enter a different username, Erin is not available.
 The username Josh is available!
 The username Avery is available!
 The username Leah is available!

5-11. Ordinal Numbers: Ordinal numbers indicate their position in a list, such as *1st* or *2nd*. Most ordinal numbers end in *th*, except 1, 2, and 3.

- Store the numbers 1 through 9 in a list.
- Loop through the list.
- Use an `if-elif-else` chain inside the loop to print the proper ordinal ending for each number. Your output should read "1st
2nd 3rd 4th 5th 6th 7th 8th 9th", and each result should be on a separate line.

```
In [39]: nums = [1,2,3,4,5,6,7,8,9]

i = 0
while i < 9:
    if nums[i] == 1:
        print(f"{nums[i]}st")
    elif nums[i] == 2:
        print(f"{nums[i]}nd")
    elif nums[i] == 3:
        print(f"{nums[i]}rd")
```

```
    else:  
        print(f"{nums[i]}th")  
    i = i+1
```

1st
2nd
3rd
4th
5th
6th
7th
8th
9th

5-12: Styling if statements: Review your code above and make sure you styled your conditional tests appropriately

5-13: Your ideas: Write down at least three ideas for programs you would like to write (games, websites, applications, etc).

games, program for something remote control, program for complicated bread board