Input statement

Conditional Statements

For Loops

While Loops

Functions

**Input Statement:** Sometimes we want to get a user input to assist our program.

user\_input = input(“Please give me some input”)

The variable above user\_input stores what the user types into the computer. So, if I typed Katie the value of user\_input = “Katie”

User input defaults to a string data type, so if you want to use user input in a math equation you need to **type cast** the data.

Ex: type cast the user input into an integer data type

user\_input = int(input(“Give me a number”))

if I typed 5 then the value of user\_input = 5

**Conditional Statements:** When writing code, we often are checking if some conditions are true or false. There are 3 types of conditional statements if, elif and else.

1. elif is short for else if
2. else statement runs if anything other than the if and else if condition is true.

If (some condition is true):

(run the code that is indented)

elif( this condition is true):

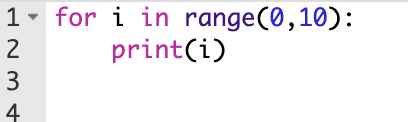
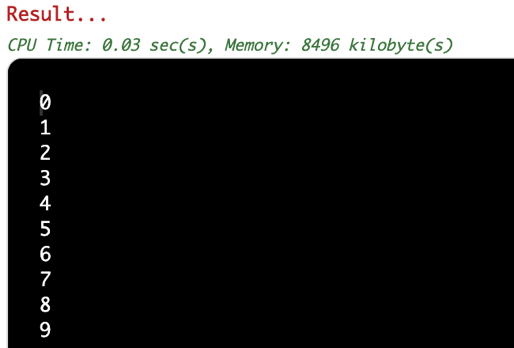
(run this code)

else:

(run this code)

**For loops:** For loops allow you to define how many times you want your code to loop.

There is more than one way to write a for loop, but for now the easiest way is using the range() function.

The range function gives us a range of numbers based on the arguments that you give it.

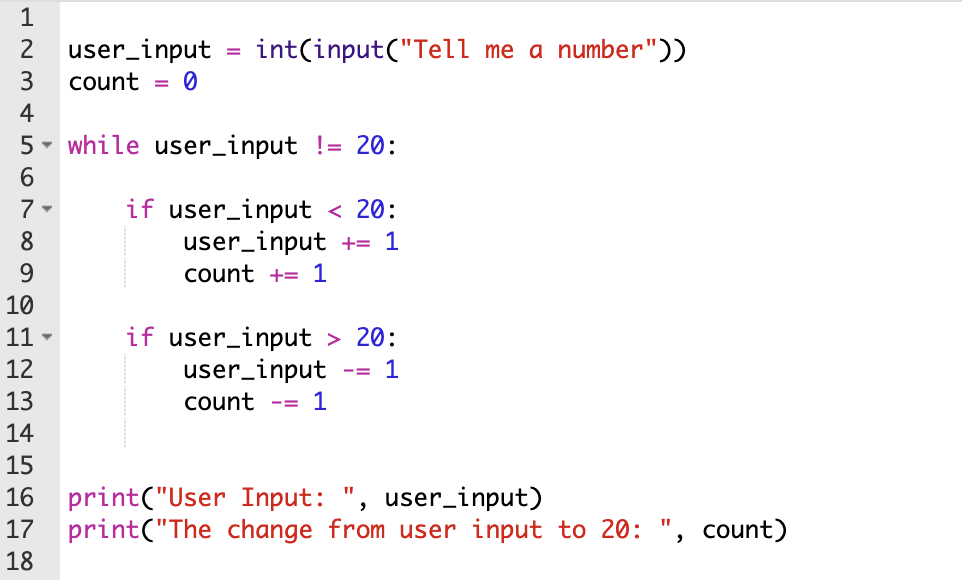
range(start, end)

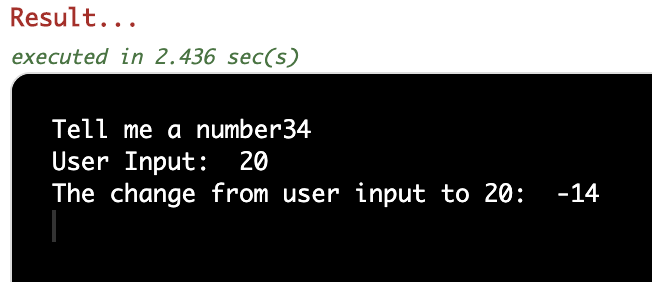
the range of numbers runs from the start value to the end value minus one. In the example above I printed out each number in the range, as you can see it prints from 0 to 9, and since the end value we gave was 10, that is correct because 10-1 = 9.

**While Loops:** While loops allow you to repeat your code until the condition you specify is met. You can think of a while loop as a way to repeat code as a conditional statement.

while (some condition):

(run the indented code)





**Functions:** In programming you may often need to use a chunk of code more than once in the same program. Functions allow you to do this.

Requirements of a function:

1. a function declaration with a name: def function():
   1. “function” is the name of this function
2. Code body (indented after the declaration statement (above)
3. A function call: function()

Parameters: functions can take zero or more input values (aka parameters)

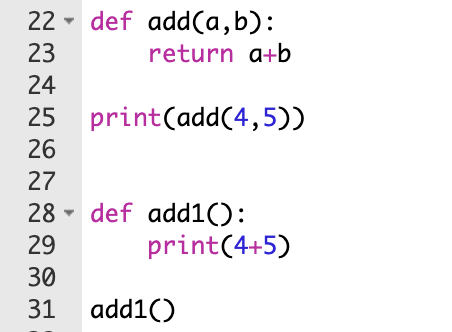
* For example: the input function we used above is a function for getting a user input
* The parameter for the input function can either be nothing or a string

Input() or input(“Give me a string!”)

Another example, the math operations like addition and subtraction buttons on a calculator are programmed using functions for each operation.

Here is how the addition function works (simplified):

2 Ways to write the addition function:



**Homework:**

1. Dice Rolling Simulator: Write a program that asks a user if they want to roll the dice. If the user types “yes” the program will generate a random number and then print that number out.
   1. The user should be asked over and over if they want to roll the dice until they give the user input “Quit” or “no”
   2. For this you will need to use the python library called Random: import random
   3. The function you will use is random.randint(a, b)
      1. Parameter a is a start value and b is the end value for which the function will generate a random number
   4. Use a while loop to keep rolling dice unless the user input equals “no”
   5. Print out the dice value after each roll

Infinite loops: while True:

1. MadLib: MadLibs are fun story games where you ask a partner for a series of nouns, verbs, and adjectives. These are then used to fill in the blank of a story. For this project use repeated user input to get a series of nouns, verbs, and adjectives to fill in a story that you write.
   1. Use input statements to get user input for each of the nouns verbs or adjectives you need
   2. Store each word you get from the user into a variable
   3. Create a story and print out the full story with the user input
2. Function Practice:
   1. Write a function that will solve the equation y = a\*\*2 + b\*\*2 for any value of a or b
   2. Write a function that will ask user input for your name and then where you are from and will print out your name and where you’re from.
   3. Define a function that will ask the user for a number between 1 and 10. If the user does not enter a number or if the user enters a number outside of the range, then the function continues to prompt the user. The function returns the first correct entry by the user.