

# 05 Prepare: Preparation Material

## Overview

---

You can now write fairly involved programs using many different variables of different types. In this lesson, you will learn one of the most important concepts of program, and that is the ability for a program to make decisions, or for it to do different things depending on a users response.

For example, in the assignment from last week, it would have been nice if you could ask the user if they wanted to pay with cash or a credit card, or ask if they had any coupons, and if so, ask what the discount was, and then to have the program behave appropriately depending on how the user responded.

You see this in almost every program you use. If I click this button, it should take me to that page. If I type my phone number incorrectly, it should ask me again. If I have more lives remaining, the game should restart, otherwise, it should inform me that the game is over.

In all of these cases, the program is following the structure of "if ... then ..." and this is what you'll be learning in this lesson.

## Preparation Material

---

Watch the following videos:

- » [Conditional Logic](#) (5 mins)
- » [Demo: Conditional Logic](#) (5 mins)
- » [Handling Multiple Conditions](#) (6 mins)

» [Demo: Multiple Conditions](#) (8 mins)

## I. THE BASIC FRAMEWORK

As shown in these videos, the basic framework for "if statements," or conditional statements is this:

```
if condition:
    then-stuff-goes-here
```

If there are things that should happen if the condition is *not* true, then we can list other things in the "else" clause like this:

```
if condition:
    then-stuff-goes-here
else:
    the-otherwise-stuff-goes-here
```

There are a few things in the syntax that are really important. The first is the colon :, it says, there is a block of code that is coming that applies to either the "if" part or the "else" part.

The next thing that is important is the indentation of the block. It technically doesn't matter how many spaces or tabs you use, as long as it is consistent. It is very common to either use two or four spaces, but again the important thing is to be consistent.

In Visual Studio Code, by default when you press the "tab" key, it will indent your code four spaces, which is what you should use.

## II. THE CONDITION

There are many options for the condition, but usually it has the form of  $x == y$  or  $x > y$  or something similar.

Don't forget that when you want to see if two items are equal **you must use two equals symbols ==** not just one. In most

programming languages, including Python, one equals sign = is used to assign a value to a variable, whereas two equal signs == are used to check if two variables are equal.

### III. COMPARING STRINGS

When comparing two strings, remember that case matters (upper case versus lower case). Because of this, it is common to convert strings that come from the user to lowercase before comparing them against a value you expect.

```
color = input("What is your favorite color? ")

# This if statement will only match "blue" not "Blue" or "BLUE"
if color == "blue":
    print("I like blue too.")

# This if statement will match the word blue, regardless of the capitalization
if color.lower() == "blue":
    print("I like blue too.")
```