

if, elif, else Statements

`if` Statements in Python allows us to tell the computer to perform alternative actions based on a certain set of results.

Verbally, we can imagine we are telling the computer:

"Hey if this case happens, perform some action"

We can then expand the idea further with `elif` and `else` statements, which allow us to tell the computer:

"Hey if this case happens, perform some action. Else, if another case happens, perform some other action. Else, if *none* of the above cases happened, perform this action."

Let's go ahead and look at the syntax format for `if` statements to get a better idea of this:

```
if case1:
    perform action1
elif case2:
    perform action2
else:
    perform action3
```

```
In [1]: if True:
        print('It was true!')

        It was true!
```

Let's add in some else logic:

```
In [2]: x = False

        if x:
            print('x was True!')
        else:
            print('x is not true')

        x is not true
```

Multiple Branches

Let's get a fuller picture of how far `if`, `elif`, and `else` can take us!

We write this out in a nested structure. Take note of how the `if`, `elif`, and `else` line up in the code. This can help you see what `if` is related to what `elif` or `else` statements.

```
In [3]: loc = 'Bengaluru'

if loc == 'Kolkata':
    print('Welcome to West Bengal!')
elif loc == 'Bengaluru':
    print('Welcome to Karnataka!')
else:
    print('Where are you?')
```

Welcome to Karnataka!

Note how the nested if statements are each checked until a True boolean causes the nested code below it to run. You should also note that you can put in as many elif statements as you want before you close off with an else.

```
In [4]: person = 'Abhishek'

if person == 'Abhishek':
    print('Welcome, Abhishek!')
else:
    print("Welcome, what's your name?")
```

Welcome, Abhishek!

```
In [5]: person = 'Rajan'

if person == 'Abhishek':
    print('Welcome, Abhishek Sarkar!')
elif person == 'Karan':
    print('Welcome Karan Srivastav!')
else:
    print("Welcome, what's your surname?")
```

Welcome, what's your surname?

Indentation

It is important to keep a good understanding of how indentation works in Python to maintain the structure and order of your code. We will touch on this topic again when we start building out functions!