

You can nest this. The 'if' and 'else' blocks can contain another 'if' statement. Like this:



```
name = 'Fred'
if name == 'Fred':
    print('Hi Fred, good to see you again')
else:
    if name:
        print('Hi', name, 'welcome')
    else:
        print('Hello stranger')
```

The diagram illustrates the execution of the code for `name = 'Fred'`. A blue box labeled `'if' block` highlights the `if name == 'Fred':` condition and its corresponding `print('Hi Fred, good to see you again')` statement. A blue box labeled `'else' block` highlights the `else:` branch. Within the `'else' block`, a red box labeled `Nested 'if' block` highlights the `if name:` condition and its `print('Hi', name, 'welcome')` statement. Another red box labeled `Nested 'else' block` highlights the `else:` branch and its `print('Hello stranger')` statement. Below the code, the output `Hi Fred, good to see you again` is shown, indicating that the first `if` block was executed.

In the example above:

- If the name is 'Fred', the (blue) 'if' block is executed. You will see the text 'Hi Fred, good to see you again'. The (blue) 'else' block is not executed at all.
- If the name is not 'Fred', the (blue) 'if' block is not executed. The (blue) 'else' block is executed. Within the 'else' block:
 1. If the name has at least one character, the (red) nested 'if' block is executed. For instance, if the name is 'Sue' you will see 'Hi Sue, welcome'
 2. If the name is empty, if it contains no characters at all, the (red) nested 'else' block is executed. You will see 'Hello stranger'

Here is the same code, with three different values for the name variable in these three examples:

```
name = 'Fred'
if name == 'Fred':
    print('Hi Fred, good to see you again')
else:
    if name:
        print('Hi', name, 'welcome')
    else:
        print('Hello stranger')
```

Hi Fred, good to see you again

```
name = 'Sue'
if name == 'Fred':
    print('Hi Fred, good to see you again')
else:
    if name:
        print('Hi,', name, 'welcome')
    else:
        print('Hello stranger')
```

Hi, Sue welcome

```
name = ''
if name == 'Fred':
    print('Hi Fred, good to see you again')
else:
    if name:
        print('Hi', name, 'welcome')
    else:
        print('Hello stranger')
```

Hello stranger

As you can see, this code does one of three different things, depending on the name.

If you want to do four, five or more different things you will get some quite messy code. There would be more and more indentation. Instead we can do this:

```
name = ''
if name == 'Fred':
    print('Hi Fred, good to see you again')
elif name == 'Sue':
    print('Hello Sue, how have you been?')
elif name:
    print('Hi', name, 'welcome')
else:
    print('Hello stranger')
```

Hello stranger

```
name = 'Sue'
if name == 'Fred':
    print('Hi Fred, good to see you again')
elif name == 'Sue':
    print('Hello Sue, how have you been?')
elif name:
    print('Hi', name, 'welcome')
else:
    print('Hello stranger')
```

Hello Sue, how have you been?

Python will keep trying the tests. First the 'if' test, then the first 'elif' test, then the next one, etc. As soon as a test passes its block is executed. Then Python continues after the whole 'if' statement, ignoring any other 'elif' or 'else' blocks.

Congratulations. You now know how to control the flow of your Python code, and make it respond to the data.

Get the free Daily Team To-do list [here](#) to take your team to the next level!

Scrum To-do List

**Grab it
now !**

- to update list of requirements from daily scrums, emails and other sources
 - to plan your requirements, stories
 - to order any new requirements
 - to write sprint report to the team (sprint)
 - to keep up any information holding an sprint backlog log. Work partly supplied with links
 - to make your team more and make them more visible to team work
 - to arrange meetings and have them to each day new or weekly team members, product owners or stakeholders
- NOTE:** When to learn how to update your team's productivity, get them asking their own team and get them planned for the day? [Check out the Daily Scrum Checklist](#) so you can learn the steps you need to coordinate and communicate with your team every day! Get your [Daily Scrum Checklist](#) [now!](#)



**Download the FREE
Check
Limited Time Offer!**

Download The "How to
Manage Your Team"