

```
else:  
    print("It's after 11 - it's lunch time.")
```

Now instead of nothing happening if the condition of the *if statement* is *False* (time ends up being greater than 11), the *else statement* will be executed. In this example, what do you think would happen if the time was set to 11 exactly? Copy, paste, and run the code sample above if you're unsure. Do you think we should amend the code at all to handle this case? How would you do that? Try adding to the code sample and running it.

Another example of using an *else statement* with an *if statement* can be found below, but this one includes an *elif*. The value that the variable *hour* holds determines which string is assigned to the variable *greeting*:

```
if hour < 18:  
    greeting = "Good morning"  
elif hour < 20:  
    greeting = "Good evening"  
else:  
    greeting = "Good night"
```

We are faced with decisions like this on a daily basis. For instance, if it is cold outside you would likely wear a jacket. However, if it is not cold, you might not find a jacket necessary. This, as you have learned, is a type of branching: if one condition is true, do one thing, and if the condition is false, do something else. This type of branching decision-making can be implemented in Python programming using 'if-elif-else' statements.

Instructions

Read through the **example** programs that accompany this task in your task folder. They will give you some helpful tips on how to implement conditional checks in your compulsory task below.

Compulsory Task 1

Follow these steps:

- Create a Python file called **full_name.py** in this folder.

This program will be used to validate that a user inputs at least two names when asked to enter their full name.

- Ask the user to input their full name.
- Perform some validation to check that the user has entered a **full** name. Give an appropriate error message if they haven't. One of the following messages should be displayed based on the user's input:
 - "You haven't entered anything. Please enter your full name."
 - "You have entered less than 4 characters. Please make sure that you have entered your name and surname."
 - "You have entered more than 25 characters. Please make sure that you have only entered your full name."
 - "Thank you for entering your name."

The error message examples should help you to determine the sorts of checks your program will need to perform on the data that the user provides.



Rate us

Share your thoughts

HyperionDev strives to provide internationally-excellent course content that helps you achieve your learning outcomes.

Think that the content of this task, or this course as a whole, can be improved, or think we've done a good job?

[Click here](#) to share your thoughts anonymously.

