

In the first line of the code, you import the `pywapi` module. Then you code the `weather` function, which takes `city_name` and `city_code` as arguments. You store the result in the `weather_com_result` variable. You then access the current conditions' text and temperature from the result received from the `pywapi.get_weather_from_weather_com()` function. You store the appropriate message for the user by assigning it to the `weather_result` variable, which is then spoken by the `tts()` function.

Finally, it is time to make the edits in `brain.py`. Edit the first line so that it imports the `weather` module you built:

```
from GreyMatter import tell_time, general_conversations, weather
```

Now, edit the `brain()` function's declaration:

```
def brain(name, speech_text, city_name, city_code):
```

And add the code snippet to detect a weather query in the `if/else` ladder:

```
    elif check_message(['how', 'weather']) or check_message(['hows', 'weather']):
        weather.weather(city_name, city_code)
```

This concludes the construction of the `weather` module for Melissa's logical engine. Now you can ask questions such as, "How is the weather?" and "How is the weather today?" and Melissa will let you know!

Adding the `weather` feature was straightforward because it involved only simple use of the module, and you didn't have to write code to retrieve the weather information from `weather.com`. The next example also uses a module, but its implementation is interesting and will help you brainstorm about adding new features to Melissa and the procedure of implementing them.

Define Artificial Intelligence!

In this example, you retrieve definitions of and information about particular keywords from Wikipedia. This will let you ask Melissa about anything that has an article on Wikipedia. For this command, you use a specific format: "Define subject."

■ **Note** I would like to point out that for proper implementation of this type of functionality, a question like "Who is Tanay Pant?" should be synonymous with "Define Tanay Pant." This would be possible by implementing natural language processing (NLP). Many NLP-based tools are available for research and development work, such as Natural Language Processing Toolkit (NLTK). You may want to read up on this topic, but let me warn you that it is a vast field. Covering NLP is beyond the scope of this book.

Before you start building the module, install the `wikipedia` module via `pip` by entering the following command in the terminal:

```
$ pip install wikipedia
```