Tool Usage

- LLMs can be trained to call developer-defined functions by predicting special tokens

- Developers write code and documentation, and prompt the model with a list of available tools

- Models predict what tool to use and what inputs to give the tool for a given query

- Not generating any code itself!

- Provide the models with additional capabilities or improve on weaknesses

- e.g. calculator, document retrieval, internet search, web browser, query other LLMs

3

```python
def get_weather(self, location: str):
    """Get the weather in a city."""
    return weather_api.get(location)
```

```
<|im_start|>system
You can use the following functions:
get_weather(location: str)<|im_end|>
```

<|im_start|>assistant

```
weather({
```

```
"location":
```

"Philadelphia,

PA"

})

```<|im_end|>
```

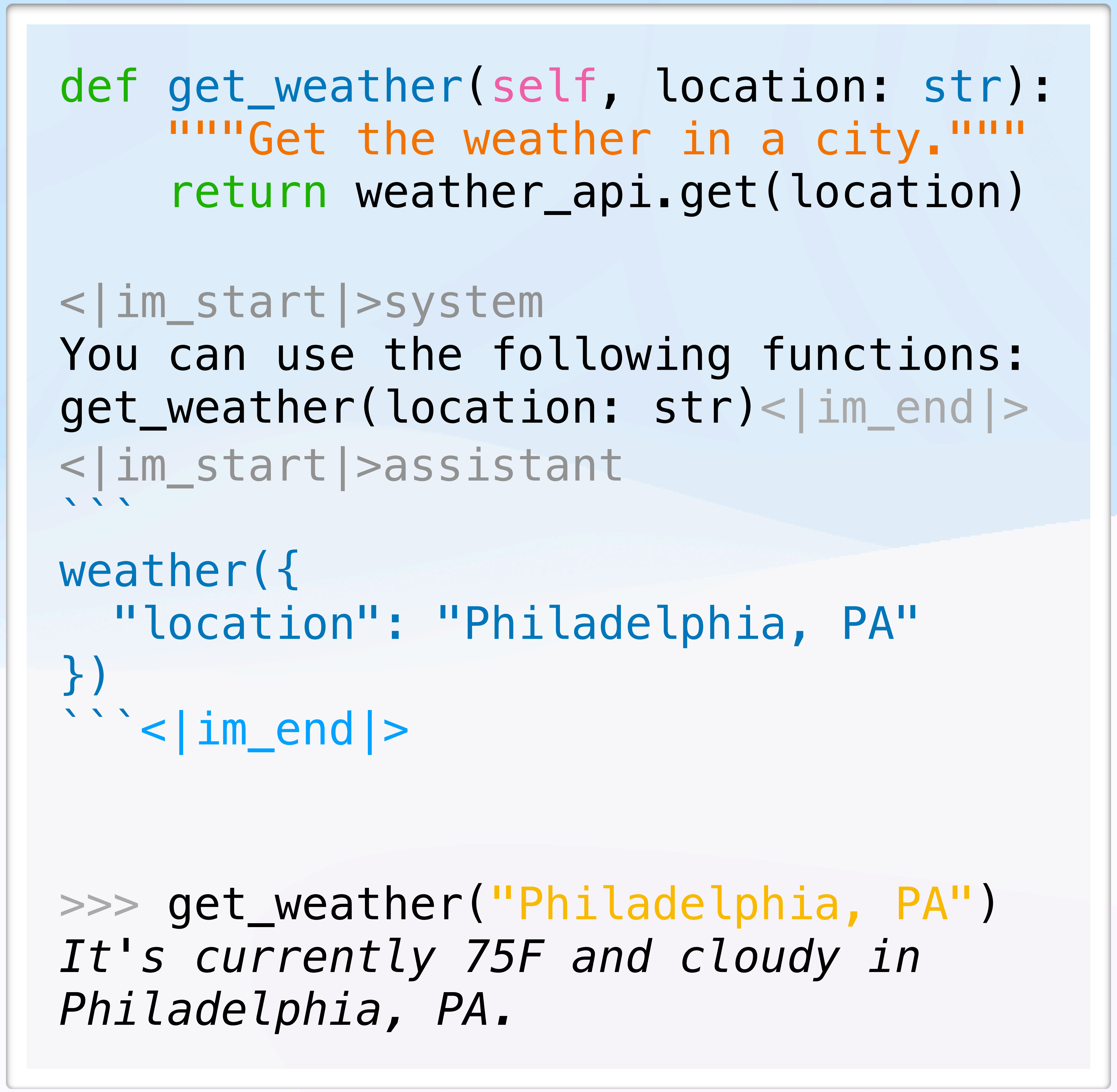
```
>>> get_weather("Philadelphia, PA")
It's currently 75F and cloudy in
Philadelphia, PA.
```

~~~
<|im_start|>assistant
```

weather({
  "location": "Philadelphia, PA"
})
```<|im_end|>
~~~

Tool Usage

- LLMs can be trained to call developer-defined functions by predicting special tokens
- Developers write code and documentation, and prompt the model with a list of available tools
- Models predict what tool to use and what inputs to give the tool for a given query
    - Not generating any code itself!
- Provide the models with additional capabilities or improve on weaknesses
    - e.g. calculator, document retrieval, internet search, web browser, query other LLMs



```
def get_weather(self, location: str):
    """Get the weather in a city."""
    return weather_api.get(location)

<|im_start|>system
You can use the following functions:
get_weather(location: str)<|im_end|>
<|im_start|>assistant
```
weather({
  "location": "Philadelphia, PA"
})
```<|im_end|>



>>> get_weather("Philadelphia, PA")
It's currently 75F and cloudy in
Philadelphia, PA.
```

Outline