



Elastic APM is an open source application performance monitoring tool that helps developers monitor the performance of their applications in real-time. It provides detailed information about application errors, transaction traces, and other important metrics that can help developers identify and fix performance issues quickly.



To use Elastic APM in Python, you can follow these steps:

1. Install the Elastic APM agent for Python using pip:

```
sh                                                                    Copy code

pip install elastic-apm
```

2. Import the Elastic APM agent in your Python application:

```
python                                                                    Copy code

import elasticapm
```

3. Initialize the Elastic APM agent with your APM server URL and other required configuration parameters:

```
python                                                                    Copy code

elasticapm.instrument(
    # Set your APM Server URL
    service_name='myapp',
    service_version='1.0',
    server_url='http://localhost:8200',
)
```

4. Use the Elastic APM agent to instrument your application code by wrapping the code you want to monitor with a ``capture_span`` context manager:

```
python                                                                    Copy code

with elasticapm.capture_span('my_span'):
    # Code to monitor here
```

5. Optionally, you can also use the Elastic APM agent to capture custom errors and log messages:

```
python                                                                    Copy code

elasticapm.capture_message('Something went wrong')
try:
    # Code that might raise an exception
except Exception as e:
    elasticapm.capture_exception(e)
```

6. Start your application and monitor the performance data in the Elastic APM dashboard.

These are the basic steps to use Elastic APM in a Python application. You can refer to the official Elastic APM documentation for more detailed instructions and examples.