

Ruby

Install:

```
gem install coralogix_logger
```

General:

You must provide the following four variables when creating a Coralogix logger instance.

Private Key - Your [Send-Your-Data API key](#) is a unique ID that represents your company.

Application Name - The name of your main application, for example, a company named "SuperData" would probably insert the "SuperData" string parameter or if they want to debug their test environment they might insert the "SuperData- Test".

SubSystem Name - Your application probably has multiple subsystems, for example, Backend servers, Middleware, Frontend servers etc. in order to help you examine the data you need, inserting the subsystem parameter is vital.

Ruby:

```
Coralogix SDK

require 'coralogix_logger'

PRIVATE_KEY = "11111111-1111-1111-1111-111111111111"
APP_NAME = "MyTestApp"
SUB_SYSTEM = "BL"

# Configure Coralogix SDK. You need to define it only once per process.
Coralogix::CoralogixLogger.configure(PRIVATE_KEY, APP_NAME, SUB_SYSTEM)

# The common practice is to get an instance of the logger in each class and
# setting the logger name to the class name.
# logger name will be used as category unless specified otherwise.
logger = Coralogix::CoralogixLogger.get_logger("my class")

# Send "Hello World!" message with severity verbose.
logger.log(Coralogix::Severity::VERBOSE, "Hello World!")
```

```
# Additional options
# Severity and message parameters are mandatory. The rest of the parameters
are optional.

logger.log(Coralogix::Severity::DEBUG, "Hello World!", category: "my
category")
logger.log(Coralogix::Severity::INFO, "Hello World!", category: "my
category", className: "my class")
logger.log(Coralogix::Severity::WARNING, "Hello World!", category: "my
category", className: "my class", methodName: "my method")
logger.log(Coralogix::Severity::ERROR, "Hello World!", category: "my
category", className: "my class", methodName: "my method", threadId:
"thread id")
logger.log(Coralogix::Severity::CRITICAL, "Hello World!", className: "my
class", methodName: "my method", threadId: "thread id")

#Using severity methods
#Only message is mandatory. The rest of the parameters are optional.

logger.debug("Hello World!")
logger.verbose("Hello World!", className: "my class")
logger.info("Hello World!", className: "my class", methodName: "my method")
logger.warning("Hello World!", className: "my class", methodName: "my
method", threadId="thread id")
logger.error("Hello World!", className: "my class", methodName: "my
method", threadId="thread id")
logger.critical("Hello World!", category: "my category", className: "my
class", methodName: "my method", threadId="thread id")
```

Need help? We love to assist our customers, simply reach out via our in-app chat, and we will walk you through, step by step.

 Last updated: October 8, 2024

Was this helpful?

Leave your feedback here.

☐ Yes ☐ No

Send

© 2025 Coralogix. All rights reserved.

Generated on: November 17, 2025

Source: <https://coralogix.com/docs/integrations/sdks/ruby/>