

Basics

Using Log4PHP

- ❖ Log4PHP includes PHP files we just need to make available for our code.

Appender

- ❖ Each appender defines a specific destination for log messages.
- ❖ An appender can refer a file, a database or even a the simple console.

Logger

- ❖ The log messages are written through loggers. Each logger has a name.
- ❖ We get a logger by calling the `Logger::getLogger` static method that was defined within the `Logger` class. Calling this method we should pass over the name of the logger.
- ❖ Each logger should be pre-defined in our configuration file.

Sample

```
<?xml version="1.0" encoding="UTF-8"?>
<log4php:configuration xmlns:log4php="http://logging.apache.org/log4php/">
  <appender name="fileappender" class="LoggerAppenderFile">
    <layout class="LoggerLayoutTTCC" />
    <param name="file" value="project_logs.log" />
  </appender>
  <logger name="Something">
    <appender_ref ref="fileappender" />
  </logger>
</log4php:configuration>
```



This is The XML Configuration File

Sample

```
<?php
include ('log4php/Logger.php');

Logger::configure('loggingconfiguration.xml');

class Something
{
    private $log;

    public function __construct()
    {
        $this->log = Logger::getLogger(__CLASS__);
    }
}
```

This is The PHP Code

Sample

```
public function run()  
{  
    $this->log->info("simple informative message");  
    $i=1;  
    $sum=0;  
    while ($i<=10)  
    {  
        $sum += $i;  
        $this->log->info("i=".$i);  
        $i++;  
    }  
}  
  
$ob = new Something();  
$ob->run();  
?>
```

Sample

```
INFO - simple informative message
```

```
INFO - i=1
```

```
INFO - i=2
```

```
...
```

```
INFO - i=10
```

```
10/16/11 00:25:46,601 [8420] INFO Something - simple informative message
```

```
...
```

```
10/16/11 00:25:46,619 [8420] INFO Something - i=9
```

```
10/16/11 00:25:46,620 [8420] INFO Something - i=10
```



This is The Logging File