

setMetric

August 18, 2026

setMetric

Kind: Function

Source: `src/middleware/timing/timing.ts`

Part of: [Middleware](#)

Set a metric for the timing middleware.

`setMetric` adds a named metric to the timing data associated with the current request context. The timing middleware reads these metrics when producing timing information for the response.

Signature

```
function setMetric(c: Context, name: string, valueDescription: number | string | undefined, des
```

Parameters

Name	Type
<code>c</code>	<code>Context</code>
<code>name</code>	<code>string</code>
<code>valueDescription</code>	<code>number</code>
<code>description</code>	<code>string</code>
<code>precision</code>	<code>number</code>

Diagram

```
graph LR
  Request --> TimingMiddleware
  TimingMiddleware --> Handler
```

```
Handler --> setMetric
setMetric --> RequestMetrics
RequestMetrics --> TimingMiddleware
TimingMiddleware --> ResponseTiming
```

Usage

```
import { timing, setMetric } from './middleware/timing/timing'

app.use('*', timing())

app.get('/reports', async (c) => {
  const startedAt = performance.now()

  const report = await loadReport()

  setMetric(c, 'report_load', performance.now() - startedAt)

  return c.json(report)
})
```

AI Coding Instructions

- Register the timing middleware before handlers that call `setMetric`.
- Pass the current request context to `setMetric` so the metric is attached to the correct response.
- Use stable, descriptive metric names to keep response timing output consistent.
- Record metric values in the unit expected by the timing middleware.