Tracetest Cheat Sheet

Read the complete documentation for the Tracetest Selector Language here

The Tracetest Selector Language defines the pattern to select **spans** to which a set of **assertions** are then applied. Tracetest Selectors can be grouped into the following categories based on the type of **spans** they can select:

Basic selectors

Empty selector

Select all spans in a trace.

Example:

```
// An empty selector is an empty string
```

Attribute selector

Select all spans that have the given attribute.

Syntax:

```
span[attr = "value"]
span[attr contains "value"]
```

Examples:

```
// Select all spans of type http
span[tracetest.span.type = "http"]

// Select all spans where the service name
contains the word "api"
span[service.name contains "api"]
```

Grouping selectors

AND operator

Select all the matching spans by applying an AND condition between the given attributes.

Syntax:

```
span[attr1 = "value" attr2 = "value"]
```

Examples:

```
// Select all spans of type http and with
service name "cart-api"
span[tracetest.span.type = "http"
service.name = "cart-api"]
```

OR operator

Select all the matching spans by applying an OR condition between the given selectors.

Syntax:

```
span[attr1 = "value"], span[attr2 = "value"]
```

Examples:

```
// Select all spans with service name "api-
product" or service name "api-cart"
span[service.name = "api-product"],
span[service.name = "api-cart"]
```

Pseudo-classes

Trace structural pseudo-classes

These pseudo-classes relate to the location of a span within the trace waterfall.

Syntax:

```
// Select the first span from the matching list
span[attr = "value"]:first

// Select the last span from the matching list
span[attr = "value"]:last

// Select a span from the matching list based
on a given index (n starts at 1)
span[attr = "value"]:nth_child(n)
```

Examples:

```
// Select the first span of type http
span[tracetest.span.type = "http"]:first

// Select the last span of type http
span[tracetest.span.type = "http"]:last

// Select the third span of type http
span[tracetest.span.type = "http"]:nth_child(3)
```

Combinators

Descendant combinator

The “ ” (space) combinator selects spans that are descendants of the first span.

Syntax:

```
span[attr1 = "value"] span[attr2 = "value"]
```

Examples:

```
// Select all spans of type http that are
descendants of the span with service name
"cart-api"
span[service.name = "cart-api"]
span[tracetest.span.type = "http"]
```