

## clojure.test cheatsheet

<https://clojure.github.io/clojure/clojure.test-api.html>

### ns form

```
(ns your.namespace-test
  (:require [your.namespace :refer :all]
            [clojure.test :refer :all]))
```

name your test namespace with the same name but add -test. Don't forget to convert hyphens to underscores in the filename.

pull in everything from the namespace you're testing and clojure.test

### defining tests

```
(deftest <name>
  <body>)
```

make a new test called <name>.

put assertions in the <body>.

### assertions

```
(is <test-expr>)
(is (thrown? <class> <test-expr>))
(is (thrown-with-msg? <class> <regex> <test-expr>))
```

assert that <test-expr> returns truthy (not false or nil)

assert that <test-expr> throws an exception of type <class>

assert that <test-expr> throws an exception of type <class> and the message matches <regex>

### adding context

```
(testing <text>
  <body>)
```

add <text> (a string) to the report for all failing assertions in <body>

### running tests at the REPL

```
user> (require 'your.namespace-test :reload-all)
user> (in-ns 'your.namespace-test)
your.namespace-test> (run-tests)
```

load (or reload) your tests

switch to the test namespace if you need to

run tests in the current namespace

### fixtures

```
(use-fixtures :once
  (fn [tests]
    <before>
    (tests)
    <after>))
```

run code around all tests in namespace

takes one or more functions. each function takes the tests as argument

the function should define what runs before and after tests. tests are called like a function.

  

```
(use-fixtures :each
  (fn [test]
    <before>
    (test)
    <after>))
```

run code around each test in namespace

takes one or more functions. each function takes the test as argument

the function should define what runs before and after the test. test is called like a function.