Cheatography

Jest Cheat Sheet Cheat Sheet

by Eli Sherer (elisherer) via cheatography.com/34195/cs/10692/

Global Environment

`afterE ach(fn)`	Called once after each test
`before Eac h(fn)`	Called once before each test
`afterA ll(fn)`	Called only once after all the tests in describe finish
`before All(fn)`	Called only once before all the tests in describe
`descri be( name, fn)`	Test suite describer
`expect (value)`	Starts an expect-ation chain
`expect.ex ten d(m atc hers)`	Adds more matchers to expect
`expect.<a sym met ric -ma tch >()`	Used as value to match to
`it(name, fn)`	Test describer (another alias for it is `test`)
`it.onl y(name, fn)`	Skip all tests after this one
`it.ski p(name, fn)`	Skip this test
`fit(name, fn)`	Runs only this test
`jest`	The `jest` object
`requir e.r equ ire Act ual (mo du l eName)`	Returns the actual module instead of a mock
`requir e.r equ ire Moc k(m odu le Name)`	Returns a mock module instead of the actual module
`xdescr ibe /xi t/xtest`	Makes the test pending
`fdescr ibe/fit`	Makes the test/suite the only one running

Writing assertions with expect

`.not`	Expects the opposite of the rest of the expectation.
`.toBe( value)`	Strict equality (`===`)
`.toEqu al( value)`	Recursively checks the equality of all fields
`.toMat ch( regexp)`	
`.toMat chO bje ct( obj ect)`	Matches a subset of the properties of an object
`.toMat chS nap shot()`	Matches the most recent snapshot (or creates one)
`.toBeD efi ned()`	
`.toBeU nde fined()`	
`.toBeN ull()`	
`.toBeT ruthy()`	
`.toBeF alsy()`	
`.toBeG rea ter Tha n(n umber)`	
`.toBeG rea ter Tha nOr Equ al( number)`	
`.toBeL ess Tha n(n umber)`	
`.toBeL ess Tha nOr Equ al( number)`	
`.toBeC los eTo (nu mber, numDigits)`	
`.toBeNaN()`	
`.toBeI nst anc eOf (Class)`	
`.toCon tai n(item)`	Strict equality (`===`)
`.toCon tai nEq ual (it em)`	Recursively checks the equality of all fields
`.toHav eLe ngt h(n umber)`	
`.toThrow()`	

By **Eli Sherer** (elisherer)
cheatography.com/elisherer/

Not published yet.
Last updated 30th January, 2017.

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

Writing assertions with expect (cont)

`.toThr owE rro r(e rror)`	`error` can be a string, RegExp or Class
`.toThr owE rro rMa tch ing Sna - pshot()`	Will expect an error while matching (the snapshot will be the error's message)
`.toHav eBe enC alled()`	
`.toHav eBe enC all edT ime s(n umber)`	
`.toHav eBe enC all edW ith (arg1, arg2, ...)`	
`.toHav eBe enL ast Cal led Wit h(arg1, arg2, ...)`	

Mock functions

`jest.fn()`	Creates a mock function
`mockFn.mo ck.c alls`	An array that represents all calls that have been made into this mock function
`mockFn.mo ck.i ns tances`	An array that contains all the object instances
`mockFn.mo ckC lear()`	Resets all information stored in the above arrays.
`mockFn.mo ckR eset()`	Resets all information stored in the mock
`mockFn.mo ckI mpl eme nta tio n(f n)`	Sets a function that should be used as the implementation of the mock
`mockFn.mo ckI mpl eme nta tio nOn ce(fn)`	Sets a function that should be used as the implementation of the mock for one call
`mockFn.mo ckR etu rnT his()`	Suger for using `jest. f n(( )=> this)`

Mock functions (cont)

`mockFn.mo ckR etu rnV alu eOn ce( value)`	

The jest object

`jest.r ese tAl lMo cks()`	Resets the s all mocks
`jest.c lea rAl lTi mers()`	Removes a pending tim from the tim system
`jest.d isa ble Aut omock()`	Disables au mocking in t module loac
`jest.e nab leA uto mock()`	Enables aut mocking in t module loac
`jest.f n(? imp lem ent ation)`	Returns a n unused moc function
`jest.i sMo ckF unc tio n(fn)`	Determines given functi mocked fun
`jest.g enM ock Fro mMo dul e(m odu leName )`	Generate a mocked ver the module
`jest.m ock (mo dul eName, ?factory, ?opti ons)`	Mocks a mo with an autc ocked versi
`jest.r ese tMo dules()`	Resets the r registry - the of all require modules
`jest.r unA llT icks()`	Exhausts th **micro**-task c
`jest.r unA llT imers()`	Exhausts th **macro**-task
`jest.r unA llI mme dia tes()`	Exhausts al queued by s `m edi ate`

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

The jest object (cont)	
`jest.r unT ime rsT oTi me( msT oRun)`	Executes only the macro task queue
`jest.r unO nly Pen din gTi mers()`	Executes only the macro-tasks that are currently pending
`jest.s etM ock (mo dul eName, module Ex p orts)`	Explicitly supplies the mock object that the module system should return for the specified module
`jest.u nmo ck( mod ule Name)`	Indicates that the module system should never return a mocked version of the specified module
`jest.u seF ake Tim ers()`	Instructs Jest to use fake versions of the standard timer functions
`jest.u seR eal Tim ers()`	Instructs Jest to use the real versions of the standard timer functions

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com