

# Sinon.JS & Jasmine BDD Cheat Sheet

including froots/jasmine-sinon

## Stub

*sinon*

stub()  
stub(obj, "fn")  
stub(obj, "fn", fn)  
stub(obj)

*stub*

withArgs(\*)  
returns(obj)  
throws()  
throws('err')  
throws(obj)  
callsArg(index)  
callsArgWith(index, \*)  
yields(\*)  
yieldsTo(property, \*)

## Mock

*sinon*

mock(obj)  
mock()

*mock*

verify()  
restore()

*expectation --- expects('fn')*

atLeast(n)  
atMost(n)  
never()  
once()  
twice()  
thrice()  
exactly(n)  
withArgs(\*)  
withExactArgs(\*)  
on(obj)  
verify()

## Spy

*spy*

callCount  
called  
calledOnce  
calledTwice  
calledThrice  
calledBefore(spy)  
calledAfter(spy)  
calledOn(obj)  
alwaysCalledOn(obj)  
calledWith(\*)  
alwaysCalledWith(\*)  
calledWithExactly(\*)  
alwaysCalledWithExactly(\*)  
threw()  
threw("TypeError")  
threw(obj)  
alwaysThrew()  
alwaysThrew('err')  
alwaysThrew(obj)  
returned(obj)  
alwaysReturned(obj)  
thisValues  
args  
exceptions  
returnValues

*sinon*

spy()  
spy(fn)  
spy(obj, 'fn')

*spyCall --- getCall(n)*

calledOn(obj)  
calledWith(\*)  
calledWithExactly(\*)  
threw()  
threw('err')  
threw(obj)  
thisValue  
args  
exception  
returnValue

*\* argument list*

## Matchers

### Jasmine Matchers

*expect(x)*

toEqual(y)  
toBe(y)  
toMatch(pattern)  
toBeDefined()  
toBeNull()  
toBeTruthy()  
toBeFalsy()  
toContain()  
toBeLessThan(y)  
toBeGreaterThan(y)

*expect(fn)*

toThrow(e)

*expect(e).not* - negate criteria

### Sinon.JS Matchers

*expect(spy)*

toHaveBeen  
Called()  
CalledOnce()  
CalledTwice()  
CalledThrice()  
CalledBefore(spy)  
CalledAfter(spy)  
CalledOn(obj)  
AlwaysCalledOn(obj)  
CalledWith(arg1, arg2, ...)  
AlwaysCalledWith(arg1, arg2, ...)  
CalledWithExactly(arg1, arg2, ...)  
AlwaysCalledWithExactly(arg1, arg2, ...)  
toHaveReturned(obj)  
toHaveAlwaysReturned(obj)

*expect(spy).not* - negate criteria