|  |
| --- |
| <https://github.com/jasmine/jasmine/blob/4be20794827a63ca5295848c58ffc478624ee4dc/src/core/SpyStrategy.js> AHK: [?**srcjss**] (**S**ou**RC**e: **J**asmine **S**py **S**trategy)  Comments are not in original source. What is **getSpy()** function?? Where is that? |
| getJasmineRequireObj().SpyStrategy = function() {  function SpyStrategy(options) {  options = options || {};  //not a fan of "identity" being both a public member function and a private variable:  //This setup is making sure nothing is null:  var identity = options.name || 'unknown',  originalFn = options.fn || function() {},  getSpy = options.getSpy || function() {},  plan = function() {};  //should be re-named "getIdentity" or maybe "getConfiguration":  this.identity = function() {  return identity;  };  this.exec = function() {  return plan.apply(this, arguments);  };  //Calls original function and returns the result. Used to observe without tampering.  this.callThrough = function() {  plan = originalFn;  return getSpy();  };  //non-plural returnValue makes more sense thant plural returnValues:  this.returnValue = function(value) {  plan = function() {  return value;  };  return getSpy();  };  //I know return values returns values... But how this works is not obvious to me from reading  //this source code.  this.returnValues = function() {  var values = Array.prototype.slice.call(arguments);  plan = function () {  return values.shift();  };  return getSpy();  };  this.throwError = function(something) {  var error = (something instanceof Error) ? something : new Error(something);  plan = function() {  throw error;  };  return getSpy();  };  //callFake is basically applying a substitution function:  this.callFake = function(fn) {  plan = fn;  return getSpy();  };  //stub is making it so empty function is invoked when calling spy:  this.stub = function(fn) {  plan = function() {};  return getSpy();  };  }  **return SpyStrategy;**  }; |