Cheatsheet: Creating mocks in PHPUnit

Available in test cases

Name of the class to mock (= to subclass)

```
$mock = $this->getMock(

                        'Tx_Coffee_Cup',
                        array('fill', 'stir'),
                        array($size, $color),
                        'MockedCup',
                        FALSE

);
```

all parameters but the first can be omitted

names of the methods to mock (empty = all)

parameters for the constructor

name of the created class (empty = auto)

whether to call the original constructor

Why mock a method?
▷ To "disable" a method (to not write to the DB, or to not launch a cruise missile) and return NULL.
▷ To have the method return a particular return value.
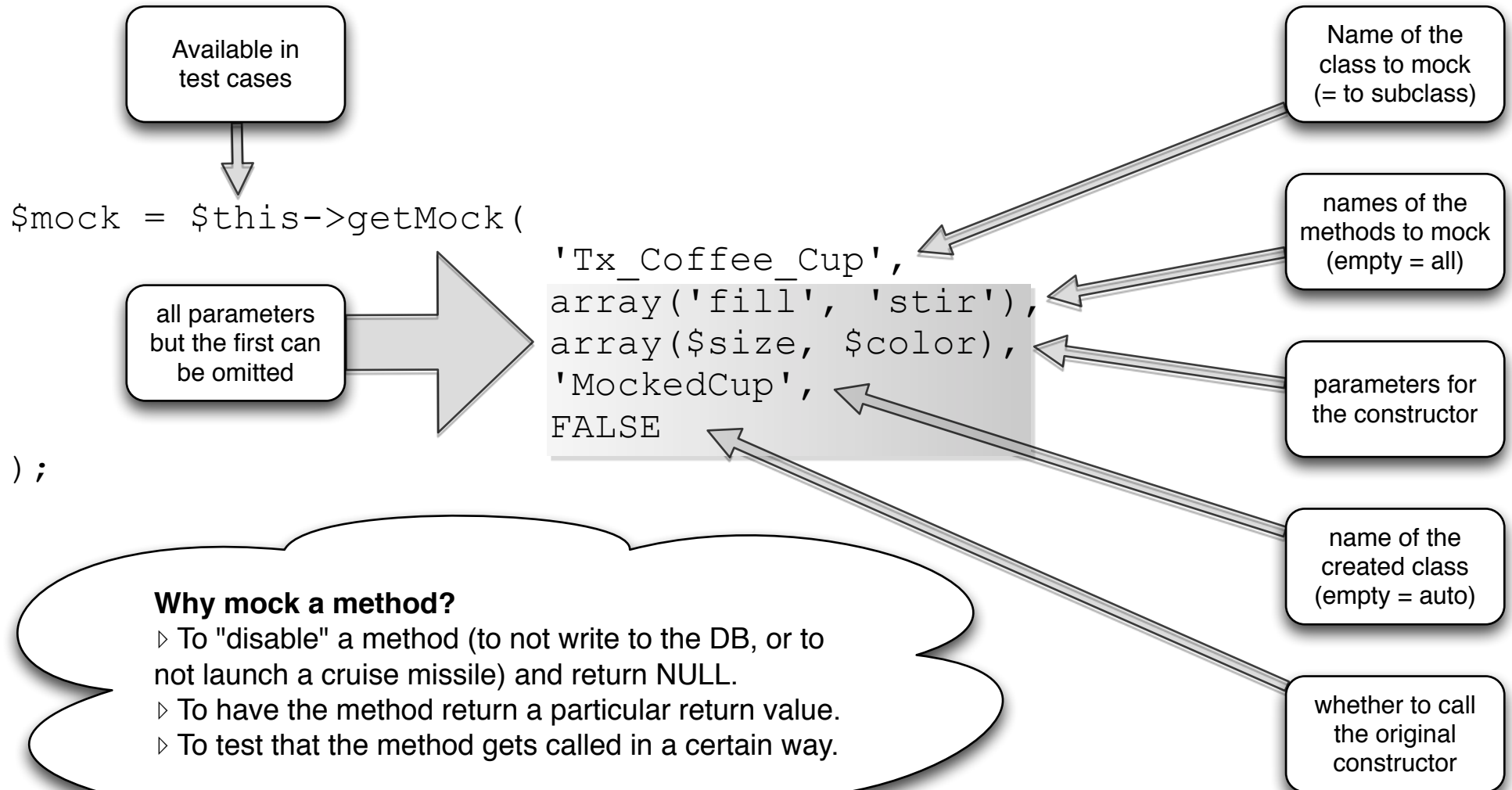▷ To test that the method gets called in a certain way.

Cheatsheet: Using mocks in PHPUnit

```php
$mock->expects($this->once())
    ->method('fill')
    ->with($coffee, 200)
    ->will(
        $this->returnValue(TRUE)
    );
```

how often:
never()
once()
any()
atLeastOnce()
exactly(4)

or which call:
at(0) for the first call
at(2) for the third call

name of the method, *must* be a mocked method

required parameters (omit for never())

return value

any of the last two parameters can be omitted