Redmine - Defect #40603
Mocha 2.2.0 causes test failure: "unexpected invocation"
2024-04-21 09:56 - Go MAEDA

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
<th>Start date:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
<td>Due date:</td>
</tr>
<tr>
<td>Assignee:</td>
<td>Go MAEDA</td>
<td>% Done:</td>
</tr>
<tr>
<td>Category:</td>
<td>Gems support</td>
<td>Estimated time:</td>
</tr>
<tr>
<td>Target version:</td>
<td>5.0.9</td>
<td>Affected version:</td>
</tr>
</tbody>
</table>

**Description**

After updating Mocha from version 2.1.0 to 2.2.0, a test began to fail with the following output:

```
$ bin/rails test test/unit/issue_test.rb:704
Run options: --seed 38306
# Running:
F
Failure: IssueTest#test_assigning_attributes_should_assign_project_and_tracker_first [app/models/issue.rb:478]: unexpected invocation: subject=(Test)
bin/rails test test/unit/issue_test.rb:704
```

**Associated revisions**

Revision 22786 - 2024-04-23 03:43 - Go MAEDA
Fix "unexpected invocation" test failure in IssueTest with Mocha 2.2.0 (#40603).

Revision 22788 - 2024-04-27 09:13 - Go MAEDA
Merged r22786 from trunk to 5.1-stable (#40603).

Revision 22789 - 2024-04-27 09:16 - Go MAEDA
Merged r22786 from trunk to 5.0-stable (#40603).

**History**

#1 - 2024-04-22 16:36 - Go MAEDA
- Category changed from Gems support to Code cleanup/refactoring

Below is the test code that currently fails. The three lines of issue.expects expect that the three methods :project_id=, :tracker_id=, and :subject= are called in this order. However, it is :subject= that is called first because issue.attributes = {:subject => 'Test'} is executed first. It is curious that this code worked with Mocha 2.1.

```ruby
def test_assigning_attributes_should_assign_project_and_tracker_first
  seq = sequence('seq')
  issue = Issue.new
  issue.expects(:project_id=).in_sequence(seq)
  issue.expects(:tracker_id=).in_sequence(seq)
  issue.expects(:subject=).in_sequence(seq)
  assert_raise Exception do
    issue.attributes = {:subject => 'Test'}
  end
  assert_nothing_raised do
    issue.attributes = {:tracker_id => 2, :project_id => 1, :subject => 'Test'}
  end
end
```
The error can be fixed simply removing `issue.attributes = {:subject => 'Test'}`.

```
diff --git a/test/unit/issue_test.rb b/test/unit/issue_test.rb
index 599c0046d..21a295aa1 100644
--- a/test/unit/issue_test.rb
+++ b/test/unit/issue_test.rb
@@ -707,9 +707,6 @@ class IssueTest < ActiveSupport::TestCase
    issue.expects(:project_id=).in_sequence(seq)
    issue.expects(:tracker_id=).in_sequence(seq)
    issue.expects(:subject=).in_sequence(seq)
-   assert_raise Exception do
-      issue.attributes = {:subject => 'Test'}
-   end
-   assert_nothing_raised do
-      issue.attributes = {:tracker_id => 2, :project_id => 1, :subject => 'Test'}
end
```

#2 - 2024-04-23 03:43 - Go MAEDA
- Category changed from Code cleanup/refactoring to Gems support
- Status changed from New to Resolved
- Assignee set to Go MAEDA
- Target version set to 5.0.9
- Resolution set to Fixed

#3 - 2024-04-27 09:17 - Go MAEDA
- Status changed from Resolved to Closed

Committed the fix in [r22789](https://example.com/r22789).