

## SUnit Debugger

Stack



```
GTSUnitExampleFailingTest(TestAsserter)>>assert:description:
GTSUnitExampleFailingTest(TestAsserter)>>assert:equals:
GTSUnitExampleFailingTest>>testMultiValuedStreaming
GTSUnitExampleFailingTest(TestCase)>>performTest
GTSUnitExampleFailingTest(TestCase)>>runCase in Block: [ ...
BlockClosure>>ensure:
GTSUnitExampleFailingTest(TestCase)>>runCase
GTSUnitExampleFailingTest(TestCase)>>debug in Block: [ (self c
BlockClosure>>ensure:
GTSUnitExampleFailingTest(TestCase)>>debug
GTSUnitExampleFailingTest class(TestCase class)>>debug:
PackageTreeNautilusUI(NautilusUI)>>runTestForAMethod:notif
```

Source



```
testMultiValuedStreaming
| headers keysAndValues |
headers := ZnHeaders new.
headers at: 'Set-Cookie' put: 'foo=1'.
headers at: 'Set-Cookie' add: 'bar=1'.
headers at: 'Content-TYPE' put: 'text/plain'.
headers at: 'Content-Length' put: '128'.
keysAndValues := OrderedCollection streamContents: [ : stream |
    headers headersDo: [ :key :value |
        stream nextPut: key; nextPut: value ] ].
self assert: keysAndValues equals: #('Set-Cookie' 'foo=1'
'Set-Cookie' 'bar=1' 'Content-Length' '128' 'Content-TYPE'
'text/plain') asOrderedCollection
```

Actual vs. expected

Inspector

☐ Pretty print

Set-Cookie

foo=1

Set-Cookie

bar=1

Content-Length

128

Content-Type

text/plain

Set-Cookie

foo=1

Set-Cookie

bar=1

Content-Length

128

Content-TYPE

text/plain