
Übungsprotokoll

TDD with Groovy

Softwareentwicklung
4CHIT 2016/17

Martin Wölfer

Version 0.2
Begonnen am 1.3.2017
Beendet am 4.3.2017

Inhaltsverzeichnis

1	Aufgabenstellung	1
2	Testreports	1
2.1	Zusammenfassung	1
2.2	Einzelne Tests	2
3	Coverage	3
4	Github	3

1 Aufgabenstellung

Schreiben Sie in einem Package bruch eine Klasse Bruch

Ziel: Coverage > 90

Empfohlene Vorgehensweise:

1. Projekt in Eclipse erstellen
2. Package test erstellen und vorhandenen Testklasse BruchTest.groovy hineinkopieren
3. Package bruch erstellen
4. Klasse Bruch.groovy im Package bruch erstellen
5. Klasse BruchTest.groovy lauffähig machen

Abgabe:

Protokoll mit Testreports (inkl. Coverage) und Dokumentation (html)

Abgabe des Groovy-files

Hinweis: Setzen Sie auf Kürze! Verwenden Sie alle Vorteile von Groovy gegenüber Java!

2 Testreports

2.1 Zusammenfassung

Unit Test Results. Designed for use with [JUnit](#) and [Ant](#).

Package test

Classes

Name	Tests	Errors	Failures	Skipped	Time(s)	Time Stamp	Host
BruchTest	11	0	0	0	1.510	2017-03-04T17:38:13	Hubert

Abbildung 1: Alle Tests wurden erfolgreich abgeschlossen

2.2 Einzelne Tests

Tests

Name	Status	Type	Time(s)
testDivideBruch	Success		0.185
testDivideBruch2	Success		0.001
testGcd	Success		0.009
testcompareToGreather	Success		0.001
testGetN	Success		0.001
testGetZ	Success		0.002
testSetN	Success		0.002
testSetZ	Success		0.000
testToString	Success		0.018
testasDouble	Success		0.052
testNegative2	Success		0.001
testPlusBruch	Success		0.070
testcompareToLower	Success		0.002
testBruchBigInteger	Success		0.001
testDivideBigInteger	Success		0.001
testMinusBruch2	Success		0.006
testasString	Success		0.000
testMultiplyBruch2	Success		0.003
testPlusBruch2	Success		0.002
testPlusBigInteger	Success		0.001
testcompareToEquals	Success		0.001
testBruchBruch	Success		0.000
testDoubleValue	Success		0.001
testMultiplyBigInteger	Success		0.001
testMinusBigIntegerBigInteger	Success		0.002
testCorrect	Success		0.002
testDivideBigInteger2	Success		0.013
testDivideBigInteger3	Success		0.002
testMinusBigInteger	Success		0.001
testDivideBigIntegerBigInteger	Success		0.002
testMinusBruch	Success		0.001
testPlusBigIntegerBigInteger	Success		0.002
testBruchBigIntegerBigInteger	Success		0.000
testasBigDecimal	Success		0.001
testasInteger	Success		0.000
testNegative	Success		0.000
testShorten	Success		0.003
testMultiplyBruch	Success		0.000
testasBigInteger	Success		0.000
testMultiplyBigIntegerBigInteger	Success		0.002
testDoubleValue	Success		0.000
testDivideBruchIllegalArgumentException	Success		0.000
testasLong	Success		0.001

Abbildung 2: Einzelne Tests wurden auch erfolgreich abgeschlossen

3 Coverage






























Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
▼ A12	 87,7 %	2.822	394	3.216
▼ src	 87,7 %	2.822	394	3.216
▼ bruch	 82,2 %	946	205	1.151
▼ Bruch.groovy	 82,2 %	946	205	1.151
▼ Bruch	 82,2 %	946	205	1.151
gcd(int, int)	 50,9 %	58	56	114
minus(Object, Object)	 86,1 %	161	26	187
lcm(int, int)	 47,9 %	23	25	48
asType(Class)	 50,0 %	24	24	48
Bruch(Bruch)	 0,0 %	0	21	21
equals(Object)	 51,4 %	18	17	35
compareTo(Object)	 51,5 %	17	16	33
div(Object)	 77,8 %	7	2	9
div(Object, Object)	 98,2 %	107	2	109
doubleValue()	 87,5 %	14	2	16
minus(Object)	 77,8 %	7	2	9
multiply(Object)	 77,8 %	7	2	9
multiply(Object, Object)	 98,2 %	107	2	109
negative()	 88,9 %	16	2	18
plus(Object)	 77,8 %	7	2	9
plus(Object, Object)	 98,9 %	185	2	187
toString()	 94,4 %	34	2	36
Bruch(BigInteger)	 100,0 %	11	0	11
Bruch(BigInteger, BigInteger)	 100,0 %	49	0	49
correct()	 100,0 %	35	0	35
shorten()	 100,0 %	45	0	45
▼ test	 90,8 %	1.876	189	2.065
▼ BruchTest.groovy	 90,8 %	1.876	189	2.065
> BruchTest	 90,8 %	1.876	189	2.065

Abbildung 3: Code-Coverage

4 Github

<https://github.com/mwoelfer-tgm/java/tree/master/A12>