jBehave and BDD example

Posted on May 18, 2015 by Arulkumaran Kumaraswamipillai

To appreciate jBehave, let's look at a better example here. This example is about a science formula

Speed = distance / Time.

So,

- given distance and time, calculate speed
- given speed and time, calculate distance
- given speed and distance, calculate time.

Step 1: The story file in plain english. **speed.story** under **src/main/resources/jBehave** folder.

```
1 Narrative: As a student, I want to practice spee
2
3 scenario: calculate speed from distance and time
4
5 Given distance = 10.0
6 Given time = 2.5
7 When calculate speed
8 Then verify speed = 4.0
```

```
9
10
11 scenario: calculate distance from speed and time
12
13 Given speed = 4.0
14 Given time = 2.5
15 When calculate distance
16 Then verify distance = 10.00
17
18 scenario: calculate time from speed and distance
19
20 Given speed = 4.0
21 Given distance = 10.0
22 When calculate time
23 Then verify time = 2.50
24
25
```

Step 2: Mapping the story to Java step class. The steps are fine grained so that they can be mixed and matched depending on th scenario.

```
1  package com.mycompany.jbehave2;
2
3  import org.jbehave.core.annotations.Given;
4  import org.jbehave.core.annotations.Named;
5  import org.jbehave.core.annotations.Then;
6  import org.jbehave.core.annotations.When;
7  import org.jbehave.core.steps.Steps;
8  import org.springframework.util.Assert;
9
10 public class SpeedVelocityFormulaSteps extends S
11 {
12
13     SpeedVelocityFormaula svf = new SpeedVelocit
14
15     private double speed;
16     private double distance;
17     private double time;
18
19     private double resultSpeed;
20     private double resultDistance;
21     private double resultTime;
22
23     @Given("speed = $speed")
24     public void speedInput(@Named("speed") doubl
25     {
26         this.speed = speed;
27     }
28
29     @Given("distance = $distance")
30     public void distanceInput(@Named("distance")
31     {
32         this.distance = distance;
33     }
34
35     @Given("time = $time")
36     public void timeInput(@Named("time") double
37     {
38         this.time = time;
39     }
```


- Unit Testing Q&A (2
  - BDD Testing (4)
    - Java BDD (Be
    - jBehave and E
    - jBehave and j
    - jBehave with t
  - Data Access Uni
  - JUnit Mockito Sp
  - Testing Spring T
  - ♦ 5 Java unit tes
  - JUnit with Hamc
  - Spring Boot in ur
- Other Interview Q&A f
- ▶ Free Java Interviev

16 Technical Key Areas

open all | close all

- Best Practice (6)
- Coding (26)
- Concurrency (6)
- Design Concepts (7)
- Design Patterns (11)
- Exception Handling (3
- Java Debugging (21)
- Judging Experience Ir
- Low Latency (7)
- Memory Managemen
- Performance (13)
- QoS (8)
- Scalability (4)
- SDLC (6)
- Security (13)
- Transaction Managen

80+ step by step Java

```
40
41      @When("calculate speed")
42      public void calcSpeed()
43      {
44          Assert.notNull(distance);
45          Assert.notNull(time);
46
47          resultSpeed = svf.calculateSpeed(this.di
48      }
49
50      @When("calculate distance")
51      public void calcDistance()
52      {
53          Assert.notNull(speed);
54          Assert.notNull(time);
55
56          resultDistance = svf.calculateDistance(t
57      }
58
59      @When("calculate time")
60      public void calcTime()
61      {
62          Assert.notNull(speed);
63          Assert.notNull(distance);
64
65          resultTime = svf.calculateTime(this.spee
66      }
67
68      @Then("verify speed = $speedSupplied")
69      public void verifySpeed(double speedSupplied
70      {
71          Assert.notNull(distance);
72          Assert.notNull(time);
73
74          junit.framework.Assert.assertEquals(spee
75      }
76
77      @Then("verify distance = $distanceSupplied")
78      public void verifyDistance(double distanceSu
79      {
80          Assert.notNull(this.speed);
81          Assert.notNull(time);
82
83          junit.framework.Assert.assertEquals(dist
84      }
85
86      @Then("verify time = $timeSupplied")
87      public void verifyTime(double timeSupplied)
88      {
89          Assert.notNull(speed);
90          Assert.notNull(distance);
91
92          junit.framework.Assert.assertEquals(time
93      }
94
95  }
96
97
```

Step 3: The actual interface that performs the calculations.

```
1  package com.mycompany.jbehave2;
2
```

```
3  public interface SpeedVelocityFormaula
4  {
5      abstract double calculateSpeed(double distan
6
7      abstract double calculateDistance(double spe
8
9      abstract double calculateTime(double speed,
10 }
11
```

Step 4: The implementation. This is the class under test.

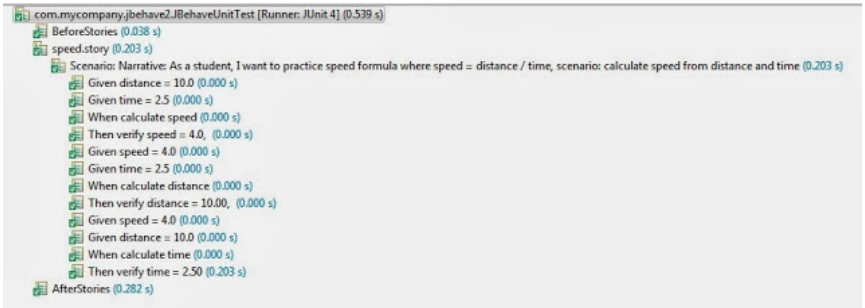
```
1  package com.mycompany.jbehave2;
2
3  public class SpeedVelocityFormulaImpl implements
4  {
5      public double calculateSpeed(double distance
6      {
7          return distance / time; // s = d/t
8      }
9
10     public double calculateDistance(double speed
11     {
12         return speed * time; //speed * time
13     }
14
15     public double calculateTime(double speed, do
16     {
17         return distance / speed;  //distance/spe
18     }
19
20 }
21
```

Step 5: Finally the jUnit test class

```
1  package com.mycompany.jbehave2;
2
3  import de.codecentric.jbehave.junit.monitoring.J
4
5  import java.util.Arrays;
6  import java.util.List;
7
8  import org.jbehave.core.junit.JUnitStories;
9  import org.jbehave.core.steps.InjectableStepsFac
10 import org.jbehave.core.steps.InstanceStepsFacto
11 import org.junit.runner.RunWith;
12
13 @RunWith(JUnitReportingRunner.class)
14 public class JBehaveUnitTest extends JUnitStorie
15 {
16
17     public JBehaveUnitTest()
18     {
19         super();
20     }
21
```

```
22      public InjectableStepsFactory stepsFactory()
23      {
24          return new InstanceStepsFactory(configur
25      }
26
27      @Override
28      protected List&lt;String&gt; storyPaths()
29      {
30          return Arrays.asList("jbehave/speed.stor
31      }
32
33  }
34
```

Step 6: Run th unit test.



Popular Member Posts

♦ 11 Spring boot interview questions & answers

905 views

♦ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers

816 views

001A: ♦ 7+ Java integration styles & patterns interview questions & answers

427 views

18 Java scenarios based interview Questions and Answers

409 views

♦ 7 Java debugging interview questions & answers

324 views

01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers

312 views

01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

304 views

♦ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

301 views

♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers

251 views

♦ Object equals Vs == and pass by reference Vs value

234 views

Bio | Latest Posts

Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.**945+** paid members. join my LinkedIn Group. **Reviews**

About Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with

this subscription based site.**945+** paid members. join my LinkedIn Group. **Reviews**

‹ Maven assembly plugin examples

♥ Unit Testing Data Access Logic in Java ›

Posted in BDD Testing, member-paid

Empowers you to open more doors, and fast-track

Technical Know Hows

☀ Java generics in no time ☀ Top 6 tips to transforming your thinking from OOP to FP ☀ How does a HashMap internally work? What is a hashing function?
☀ 10+ Java String class interview Q&As ☀ Java auto un/boxing benefits & caveats ☀ Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect

Non-Technical Know Hows

☀ 6 Aspects that can motivate you to fast-track your career & go places ☀ Are you reinventing yourself as a Java developer? ☀ 8 tips to safeguard your Java career against offshoring ☀ My top 5 career mistakes

Prepare to succeed

☀ Turn readers of your Java CV go from “Blah blah” to “Wow”? ☀ How to prepare for Java job interviews? ☀ 16 Technical Key Areas ☀ How to choose from multiple Java job offers?

Select Category

© Disclaimer

↑