♦ Ensuring code quality in Java Q&As

Posted on September 27, 2014 by Arulkumaran Kumaraswamipillai — No Comments ↓

Q1. How do you ensure code quality in your application?
A1. Code quality means writing **readable**, **maintainable** and robust code, that conforms as much as possible to the style-guideline that is used, and that has as little as possible defects. It also means writing maintainable code with proper automated and manual tests.

1. **Write a number of automated tests**
    - Unit tests using **JUnit** or **TestNG**. For unit tests use mock objects to ensure that your tests don't fail due to volatility of the data changes. There are mocking frameworks like **EasyMock**, **Mockito**, and **PowerMock**.
    - Integration testing of your services with **JUnit** or **TestNG**. Your integration tests are only as good as the quality of the data. You could either use dedicated test databases or use frameworks like **DBUnit** to manage extraction and insertion of data.

600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs

open all | close all

⊞ Ice Breaker Interview
⊞ Core Java Interview (
⊞ JEE Interview Q&A (3
⊞ Pressed for time? Jav
⊞ SQL, XML, UML, JSC
⊞ Hadoop & BigData Int
⊞ Java Architecture Inte
⊞ Scala Interview Q&As
⊞ Spring, Hibernate, & I
⊟ Testing & Profiling/Sa
  ⊞ Automation Testing
  ⊞ Code Coverage (2)
  ⊟ Code Quality (2)
    ♥ 30+ Java Code
    ♦ Ensuring code
  ⊞ jvisualvm profiling (
  ⊞ Performance Testir
  ⊞ Unit Testing Q&A (2
⊞ Other Interview Q&A 1
⊞ ▶ Free Java Interviev

- Web testing using **Selenium + WebDriver**. Selenium + WebDriver (Selenium interview questions and answers) allows you to reenact web user experience and run it as an automated unit test using JUnit or TestNG.Your tests are only as good as the quality of the data. You could either use dedicated system test databases or use frameworks like DBUnit. DBUnit allows you to extract the data from databases into flat XML files, and then refresh (i.e. insert or update) the data into the database during setup phase of running the unit tests. There are handy proxy JDBC driver tool called **P6SPY**, which logs the SQL queries that are executed against the database by the **DBUnit**. This P6SPY also very handy in debugging Hibernate's generated SQL by acting as a proxy driver between JDBC and the real driver so that all generated SQL will be logged. There are other Web testing tools like **Badboy**.
- Load testing your application with tools like **JMeter**, **OpenSTA**, etc. The **Badboy** compliments **JMeter** by allowing you to record scripts and then exporting the scripts as a JMeter file to be used in JMeter.JMeter Interview Questions and Answers

2. Have regular code reviews. There are tools like **Crucible** from Atlassian that gives your team an efficient way to benefit from the power of constant code review with features like inline commenting, simple workflow, asynchronous reviews, email and RSS notifications, JIRA integration and much more.

3. Using a number of code quality tools.
   - **Checkstyle** ensures the style of your Java code is standardized and "nice". It checks white spaces, new lines, formatting, etc. (i.e. it looks on the code line by line). This only ensure style of your code.
   - On the other hand there is **PMD** which not necessarily checks the style of your code but it checks the structure of the whole code. PMD scans Java source code and looks for potential problems like possible bugs, dead code,

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 I
⊞ Low Latency (7)
⊞ Memory Managemen
⊞ Performance (13)
⊞ QoS (8)
⊞ Scalability (4)
⊞ SDLC (6)
⊞ Security (13)
⊞ Transaction Managen

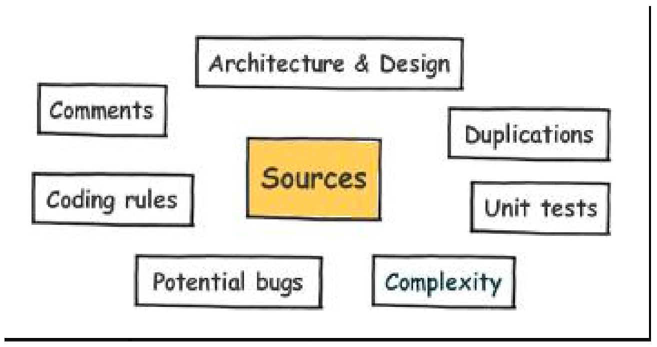
80+ step by step Java Tutorials

open all | close all

⊞ Setting up Tutorial (6)
⊞ Tutorial - Diagnosis (2
⊞ Akka Tutorial (9)
⊞ Core Java Tutorials (2
⊞ Hadoop & Spark Tuto
⊞ JEE Tutorials (19)
⊞ Scala Tutorials (1)
⊞ Spring & HIbernate Tu
⊞ Tools Tutorials (19)
⊞ Other Tutorials (45)

100+ Java pre-interview

suboptimal code, overcomplicated expressions, duplicate code, etc.
- **FindBugs** is a static analysis tool to look for bugs in Java code. It discovers possible *NullPointerExceptions* and a lot more bugs.
- **Sonar** is a very powerful tool covering 7 axes of code quality as shown below.



Sonar code quality tool

4. Using continuous integration servers (on a clean separate machine) like **Bamboo**, **Hudson**, **CruiseControl**, etc to continuously integrate and test your code.
5. Not stopping to code once the code works. Too many developers feel their job stops at making something happen. It is a best practice to constantly **refactor code** with proper unit tests in place.

Q2. Do you use test driven development? Why / Why not?
A2. [Hint] Yes.

- Gives you a better understanding of what you're going to write. Gets you to clearly think what the inputs are and what the output is. Helps you separate the concerns by getting you to think about the single responsibility principle (SRP).
- Enforces a better test coverage. This gives you the confidence to refactor your code in the future, since you have a good coverage.
- You won't waste time writing features you don't need.

Q3. Write a program that will return whichever value is nearest to the value of 100 from two given int numbers.

coding tests

open all | close all
Can you write code? (
♦ Complete the given
Converting from A to I
Designing your classe
Java Data Structures
Passing the unit tests
What is wrong with th
Writing Code Home A
Written Test Core Jav
Written Test JEE (1)

How good are your .....?

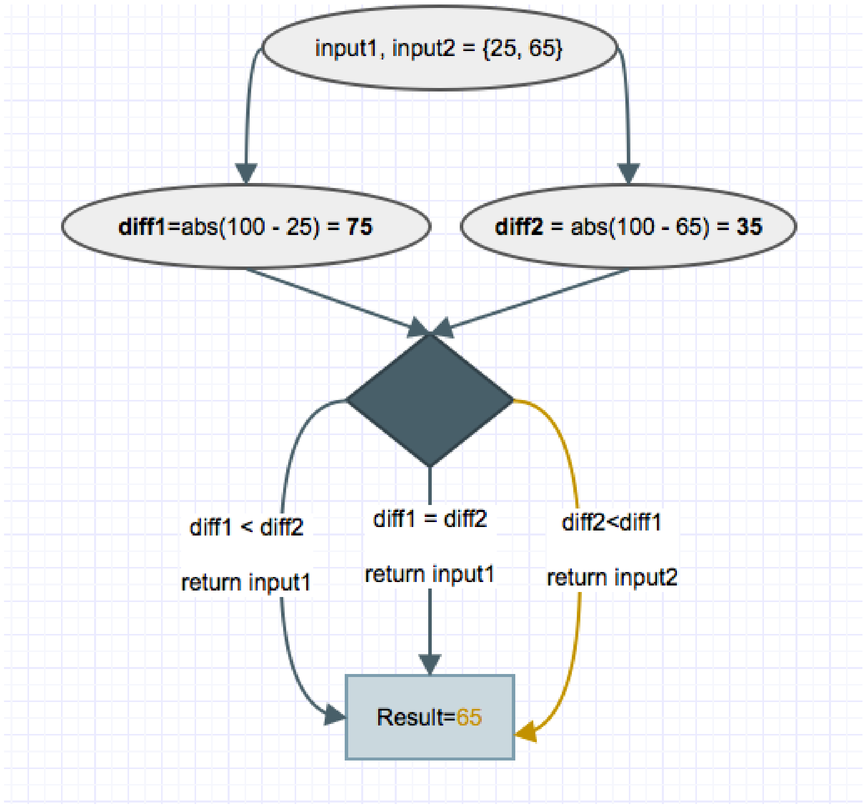
open all | close all
Career Making Know-
Job Hunting & Resum

A3.

1. Write pseudo code first:

- Compute the difference to 100.
- Find out the absolute difference as negative numbers are valid.
- Compare the differences to find out the nearest number to 100.

2. Draw a diagram if it helps



3. Consider the edge cases and write unit tests

Write test cases for +ve, -ve, equal to, > than and < than values. For example, {25, 65}, {-25, -65}, {30, 30}, {65, 25}, {110, 145}, etc.

```java
import org.junit.Assert;
import org.junit.Test;

```

```
//{25, 65}, {-25, -65}, {30, 30}, {65, 25}, {110
public class CloseTo100Test {

        @Test
        public void testPositiveNumbers(){
            Assert.assertEquals(65,CloseTo100.cal
        }

        @Test
        public void testNegativeNumbers(){
            Assert.assertEquals(-25,CloseTo100.ca
        }

        @Test
        public void testEqualNumbers(){
            Assert.assertEquals(30,CloseTo100.cal
        }

        @Test
        public void testLessThan100Numbers(){
            Assert.assertEquals(65,CloseTo100.cal
        }

        @Test
        public void testGreaterThan100Numbers(){
            Assert.assertEquals(110,CloseTo100.ca
        }

        @Test
        public void testNegativeNumbers2(){
            Assert.assertEquals(-110,CloseTo100.c
        }
}
```

junit-xxx.jar and hamcrest0core-xxx.jar files need to be in the classpath.

4. Write code

```
public class CloseTo100 {

    public static int calculate(int input1, int

       //compute the difference. Negative values
        int iput1Diff = Math.abs(100 - input1);
        int iput2Diff = Math.abs(100 - input2);

        //compare the difference
        if (iput1Diff < iput2Diff) {
           return input1;
        }
        else if (iput2Diff < iput1Diff) {
               return input2;
        }
        else{
              return input1;
        }
    }

}
```

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

311 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

0 Like Share reddit 0 G+1 **0** Share

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**

‹ 05: ♥ What job interviews are, and are not?

Reverse Polish Notation (RPN) ›

Posted in Code Quality**,** Code Quality Job Interview Essentials**,** Coding**,** member-paid

Leave a Reply

Logged in as geethika. Log out?

Comment

Post Comment

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