



# How To Become A Software Tester

Self-study course on software QA and Testing

[Home](#)   [Course](#)   [QA Glossary](#)   [Downloads](#)   [Feedback](#)   [About/Contact](#)   [License](#)

Search →

## Bad Test Case Practices

**Like**   **Share** Be the first of your friends to like this.

Lecture 3 - Test Cases and Test Suites -> Quick Intro -> Test Case Structure ->  
Results Of The Test Case Execution -> Useful Attributes Of The Test Case ->  
Data-Driven Test Cases -> Maintainability Of Test Cases -> The Number Of Expected  
Results Inside One Test Case -> Bad Test Case Practices -> Test Suites

Let's take a look at 3 bad test case practices and learn how to avoid them.

### 1. Dependency between test cases

### 2. Poor description of the steps

### 3. Poor description of the IDEA and/or expected result

#### 1. DEPENDENCY BETWEEN TEST CASES

“Dependency” is the opposite of “independency.” An independent test case is a test case that does not **rely** on any other test cases.

##### Example

###### Test Case #1

The steps:

1. Enter living room.
2. Head for large leather chair.
3. Open **right** external pocket of backpack.

Expected result: huge beer mug

###### Test Case #2

The steps:

1. Enter living room.
2. Head for large leather chair.
3. Open **left** external pocket of backpack.

Expected result: potato chips

As we can see, Steps 1 and 2 are the same in both test cases. Hence, there can be a temptation to improve the steps (especially after learning about test case maintainability). Let's do this and see what happens.

### Example

#### Test Case #1

The steps:

1. Enter living room.
2. Head for large leather chair.
3. Open **right** external pocket of backpack.

Expected result: huge beer mug.

#### Test Case #2

The steps:

1. See Steps 1 and 2 from Test Case #1
2. Open **left** external pocket of backpack

Expected result: potato chips

Does Test Case #2 look more modular now? Maybe.

Should we use this practice? **NEVER**.

Why not?

What if:

- Test Case #1 is deleted because we don't need it anymore (e.g., no more beer)

during fishing)?

- Steps 1 and 2 of Test Case #1 are changed (e.g., huge beer mug is put into another backpack, which is in the kitchen)?

In both cases, it's not clear how to execute Test Case #2 because:

- We either don't know the steps at all, or
- The steps that we have are not correct.

Another frequent situation occurs when there is an **assumption** that the software and/or DB is brought to a certain state because previous test cases have been executed.

### Example

#### Test Case #1

The steps:

1. Set the count of MasterCard transactions in the DB to 0.
2. Buy a book using MasterCard.

Expected result: the transaction success code is "20".

#### Test Case #2

The steps:

1. Buy a book using MasterCard.

Expected result: the count for MasterCard transactions is "2".

So, Test Case #2 was written with the assumption that before its execution:

- Test Case #1 had already been executed.
- The code tested in Test Case #1 doesn't have any bugs, so all data is generated in the correct way.

What is wrong with this approach? In real life, many things concerning Test Case #1 might happen. For example:

- Test Case #1 might be deleted.
- Test Case #1 might be modified so that it creates transactions with a different credit card; e.g., Visa.

- Test Case #1 might not create any transactions simply because the code is broken.

In all three cases, the execution of Test Case #2 will at least lead to confusion because its execution relies on data generated during the execution of Test Case #1.

Thus, a good test case:

- Doesn't refer to other test cases.**
- Doesn't depend on "tracks" left by the execution of other test cases.**

Hence, if we have ten test cases which are independent, we can execute them in any order: Test Case #10, Test Case #2, Test Case #7, etc.

Sometimes the repetition of steps in several test cases seems redundant, but the advantages of creating independency in the test cases outweigh the inconvenience of the copy-paste operation.

## 2. POOR DESCRIPTION OF THE STEPS

Imagine that after visiting beautiful San Francisco, you have to drive to Los Angeles for your high-school reunion. You ask your friend Arnold who lives in San Francisco to write down driving directions from Haight Street in San Francisco to Vine Street in Los Angeles.

a. What if Arnold gives you these directions:

Trip To LA

1.	Take US-101*
2.	Take I-80
3.	Take I-580
4.	Take I-5
5.	Take CA-170
6.	Take US-101
7.	Take exit 9A

\*number of the freeway (high-speed road).

I sincerely doubt that anyone who is unfamiliar with driving from SF to LA can use this. The problems will start when you reach US-101 (Step 1), because you will not know whether to go North (**N**) or South (**S**) to end up at I-80. Here is a situation **where you have steps, but these steps lack clarity, and thus it's hard to use them.**

b. What if you are given driving directions like these:

1.	Take <b>US-101 S</b>
2.	Take <b>I-80 E</b>
3.	Take <b>I-580 E</b>
4.	Take <b>I-5 S</b>
5.	Take <b>CA-170 S</b>
6.	Take <b>US-101 S</b>
7.	Take Exit <b>9A</b>

This set of directions is much better, because once you approach the freeway it's clear what direction (North, South, East, or West) to take.

But there is another problem. How would you first get to US-101 S if you don't know San Francisco? Yes, you could ask some unconcerned hippie on Haight Street, or you could follow the signs on the electricity poles, but what is the guarantee that you'll merge onto US-101 at the part that is north of I-80, so that Step 1, "Take US-101 S," would eventually bring you to Step 2? In the driving directions above, it's not clear how to reach Step 1, because Arnold:

- had an **assumption** that it's not a big deal to get to US-101, and
- **he knows** that the closest part of US-101 is to the north of I-80.

I suggest that you fire Arnold as your how-to-get-there consultant and use Google maps instead:

1.	Head <b>north</b> on <b>Divisadero St</b> toward <b>Page St</b>
2.	Turn <b>right</b> at <b>Oak St</b>
3.	Turn <b>right</b> at <b>Octavia Blvd</b>
4.	Slight <b>right</b> at <b>Central Fwy/US-101 S</b> (signs for <b>US-101 S</b> )
5.	Take the exit on the <b>left</b> onto <b>I-80 E</b> toward <b>Bay Bridge/Oakland</b>
6.	Take the exit onto <b>I-580 E</b> toward <b>CA-24/Downtown Oakland/Hayward-Stockton</b>
7.	Merge onto <b>I-5 S</b>
8.	Slight <b>right</b> at <b>CA-170 S</b> (signs for <b>Hollywood/CA-170 S</b> )
9.	Merge onto <b>US-101 S</b>
10.	Take Exit <b>9A</b> for <b>Vine St</b>
11.	Turn <b>right</b> at <b>Vine St</b>

Very nice! The lesson here is that all the steps should be clear and concrete.

You should always keep in mind that:

**Whatever is obvious for you now can become absolutely unclear in several months.**

For example, unexplained abbreviations (e.g., “DCBT”), compacted wording (like “svrl” instead of “several”) and other similar items can become Egyptian hieroglyphs even to the author of the test case, so it will be easier to create and execute a new test case rather than cut through the jungle of these vague steps.

BTW, “DCBT” stands for ... “do cross-browser testing.” Did you guess it? No? Neither did I, so I had to call the test case author (who had already left the company) to find out what he meant by “DCBT.”

**“**

**A test case that can be executed only by its author  
should be publicly denounced, burnt, and its ashes  
thrown into the ocean to make sure that no one will ever  
try to execute that test case again.**

Why? Very simple. What if the test case author gets sick, takes a vacation, departs from the company, or even carelessly dies, leaving behind an unexecutable test case?

**“**

**Each test case should be created with the thought that  
another person will have to execute it in the future.**

But even with all of this, try to remember that excessive details are bad for the maintainability of test cases. So, my friends, let's try to find a middle way.

### **3. POOR DESCRIPTION OF THE IDEA AND/OR EXPECTED RESULT**

Both principles that we just talked about:

- The future can erase from one's memory those things which are crystal clear today
- You don't create test cases for yourself; you create them for the company

are also applicable for describing an IDEA and/or ER. Here are the nuances:

#### **a. I don't recommend making a reference to the external document in the descriptions of an IDEA and/or ER.**

When we've talked about sending repetitive scenarios to the QA Knowledge Base, the reason for doing so was to improve the maintainability of the test cases. In the case of IDEA and ER, it's a different story.

##### **Example**

Would it be **comfortable** to execute a test case with this IDEA:

“In this test case, we check item 21.b of the spec #3934 “Adding credit card to user's account.””

Or how about a test case with this ER?

"Verify that the value of the last step equals the result of calculation according to the first formula from the top on page 233 of the book Stock Market Finances where X is the value from Step 2, Y is the value from Step 24, and Z is a value in the column "Number of Calls" corresponding to the value from Step 11 in Table 1 of the Spec #S122709."

I doubt that you would be happy to execute a test case like that. I got sweaty even reading that ER!

**b. We have to remember that the purpose of an IDEA is TO EXPLAIN.**

If the section IDEA is empty or modestly states "10," everyone who executes this test case will have to spend several minutes of his or her time to clarify what the heck he or she is testing here.

**c. We have to remember that the ER is the information we use (along with the AR) to make a decision as to whether a test case passed or failed.**

Therefore, precision and clarity play a most critical role in ER.

**Example**

The Expected Result states: "Verify if error message is displayed." Mr. Wei executes a test case and sees that the error message is displayed. But what if the error message says, "Please provide your Social Security Number," while according to the spec it should say, "Your Social Security Number is invalid"? In this case, the bug will be missed. It wouldn't be Mr. Wei's fault, but the fault of the tester who created that ER.

**BTW**

Please note that I don't advocate that the test case author has to necessarily provide the concrete text of an error message. For example, in a startup environment, the text of an error message can be changed 3 times a week, and, so for maintainability purposes, it makes much more sense to create this kind of ER: "Verify that the error message about an invalid SSN is displayed," rather than to put the concrete text of the error message and then spend time for maintenance every time the wording of that error message is changed.

Another example of a bad ER: "Everything works".

**Lecture 3 - Test Cases and Test Suites -> Quick Intro -> Test Case Structure ->  
 Results Of The Test Case Execution -> Useful Attributes Of The Test Case ->  
 Data-Driven Test Cases -> Maintainability Of Test Cases -> The Number Of Expected  
 Results Inside One Test Case -> Bad Test Case Practices -> Test Suites**

**Like** **Share** Be the first of your friends to like this.

**0 Comments**

**QA Tutor**

 **Login** ▾

Sort by Best ▾

**Share**  **Favorite** 



Start the discussion...

Be the first to comment.

 **Subscribe**

 **Add Disqus to your site**

 **Privacy**

**Like us on  
Facebook**

**Find us on Facebook**



**How to  
become  
a  
software  
tester**

**Like**

254 people like [How to become a software tester](#).



**Buy teacher a  
beer**



**About QA Tutor**

QATutor.com contains Practical QA Course for everybody who wants to become a software tester. It's FREE for personal use.

That course is geared to teach QA in simple, practical and effective way.

If you want to become a software tester, but don't know where to begin - you've found the right place! This course is self-study and easy to follow.

**Copyright**

© 2008-2014 Roman Savin Consulting. All rights reserved.

Materials contained in the website may be copied, reproduced or redistributed only with a link back to qatutor.com.

This course is FREE for personal use only. If you represent any kind of educational or commercial organization you will need to get a license to use materials from this site.