

Testing and **Debugging** are two words that seem to share a similar meaning. However, they are very different from each other. They have a quite analogous function. However, they are different in their designs, specifications, execution, and benefits.

In this article, we will do a detailed analysis and comparison of these two techniques. It will further allow us to differentiate both of them accurately.

What is Testing and How it is different from Debugging?

What is Debugging?

Various Differences between Testing and Debugging?

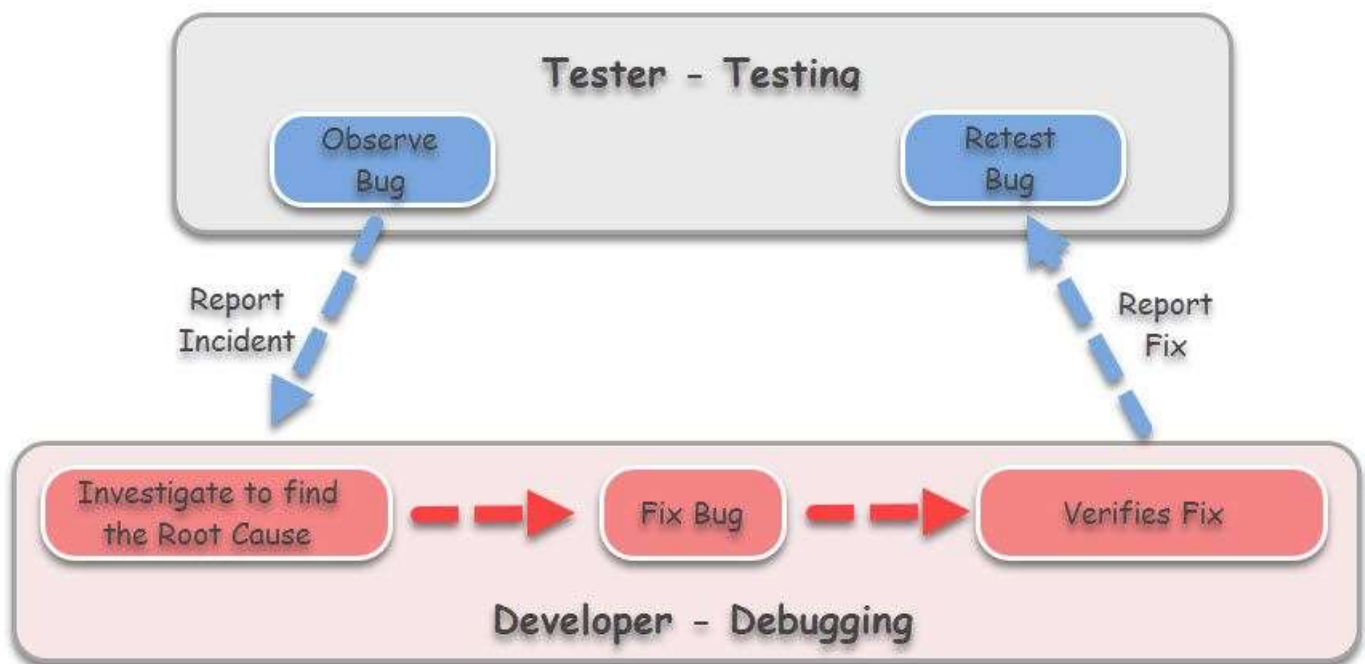
How to optimize the debugging process?

What is Testing and how it is different from Debugging?

Testing is a process to verify whether the system is working in the same way as expected and to ensure that the software is **Defect-free**. The tester does this to identify the errors/bugs in the system. Testing can happen manually, as well as through automation.

If an error or a bug appears in a program, the developer identifies the corresponding failure and fixes it.

Testing and **Debugging** are significant activities during software development and maintenance. **Testing** aims at finding a problem while **Debugging** aims at solving the problem. Only after the testing team reports the **defect**, **debugging** can take place. With debugging, the developer identifies the problem in the system/application/code. Once the developer has fixed the bug, tester re-tests to ensure that the error/bug no longer exists. The figure given below demonstrates the fact very well.



What is Debugging?

Debugging is an unstructured, step-by-step, manual process to find and eliminate a specific system error. Depending on the defects reported, the developer begins to diagnose a program/code to find a **bug**. Developers must correct the code so that the actual result is the same as expected. With the help of the added details by the tester, the developer finds out the error and then fixes it. After correcting the error, the developer tests the fix and sends the issue back to the tester to re-test.

What are the Differences between Testing and Debugging?

Testing and **Debugging**, both help in making the software defect-free but we have some basic differences between these two techniques.

Features	Testing	Debugging
Objective	The purpose of testing is to find errors/bugs in the application.	The purpose of debugging is to correct the errors found during the testing process.
Performed by	The Tester or sometimes Developer, perform the testing.	The Developer does Debug.

Features	Testing	Debugging
Automation	Testing can be manual or automated.	Debugging cannot be automated.
Programming Knowledge	Most testing requires no source code knowledge.	Debugging needs proper source code knowledge.

How to Optimize Debugging Process

Like **Testing**, we have to follow certain underlying guidelines for optimizing **Debugging**. Which, in turn, ensures an efficient flow of the process. Some basic guidelines are as below:

It's all about Mindset: Firstly, the developer should accept that there is something wrong with the code. If he is not convinced, he is less likely to find out the root cause.

Use Log Files: Secondly, the developers who effectively use logs are more likely to debug the defect quickly than those who don't use logs effectively.

One at a time: Additionally, if there are multiple changes to be made, it should be done one at a time. Debugging becomes much more complicated if the application of all the changes happens at the same time.

One to Many Fixes: After successfully debugging one defect, root cause analysis is carried out to find out similar mistakes at other places in the code, as well. Correction of all of these happens in one go.

In this article, we learned, **Testing and Debugging** are two essential parts of the **Software Development Life Cycle**, and both play a crucial role in the detection and elimination of errors.

Testing helps to find the error in the software. However, through **Debugging**, the developer identifies, analyzes, and eliminates the software error.