



SOFTWARE TESTING ESSENTIALS COURSE

The Software Testing Essentials course is a one day introductory course explaining the fundamentals of software testing. The course explains the importance of testing and reasons why we test software. It looks at how testing fits into the overall Development Lifecycle and what can be done to improve effectiveness and efficiency of testing within our organisation.

SOFTWARE TESTING ESSENTIALS COURSE

Course Objectives

To provide delegates with an understanding of the principles of testing and explain different testing terminology. The course also provides an overview of the different testing techniques, tools and how to manage the test process.

Who Will Benefit?

This 1-day course is appropriate for anyone new to testing or who wants a basic understanding of the concepts of testing. It will benefit testers, test team leaders, developers, project managers, users, business analysts, members of the QA team and senior managers.

Skills Gained

- Learn about the principles and importance of testing and how it can make a difference within your organisation
- To recognise 6 common types of bug that exists within our software and the strategies to use to attack and destroy them
- To gain an understanding of the various types of testing tools and where they fit into the overall lifecycle
- To understand and apply a powerful monitoring technique known as S-Curves
- To improve your skills in test estimation using 6 different estimation methods. And to understanding some of the problems associated with test estimation
- To be able to measure the value of testing using a simple yet powerful measurement technique called DDP (Defect Detection Percentage)

Prerequisites

None.

Course Content

Principles of testing

Importance of Testing discusses how much project effort is typically spent testing and shows that although the purpose of testing is to assess the quality of the software, there can only be confidence in this assessment if the testing itself is of good quality. Describes the uses of software testing to assess, achieve and preserve software quality and highlights the importance of prioritising the tests.

Principles of Testing explains that looking for errors is the most effective approach to testing and that in order to have justified confidence in the quality of software it is necessary to try to break it.

Causes of Error describes a number of common causes of error and, although they cannot be eliminated, shows ways in which their numbers can at least be reduced.

Testing in the Lifecycle

The V life cycle model uses the V-Model software development life cycle to explain the relationship between testing and development activities and describes the different types of testing activities, their purposes and benefits.

Levels of testing explains the main levels of testing including unit testing, integration testing, system testing, acceptance testing, system integration testing and maintenance testing. Also compares these with regression testing.

The testing process breaks down the testing process into a number of distinct activities to show what can be done to improve effectiveness and efficiency.

Overview on Testing Techniques and Tools

Types of bug explains that there are many different types of bug that we can find in our software. Each of these will require a different technique to find and destroy.

What is a testing technique?

Explains what testing techniques are and the different types available. Also outlines a number of techniques and the differences between them. This section explains the advantages of using testing techniques.

Black Box, White Box and Coverage concepts

The difference between Black and White Box testing. What coverage is and how it can be used to give an objective measure, in some part, of test 'thoroughness'. Also highlights the dangers of considering structural coverage alone.

Testing Tools and Automation Explains the different types of testing tool and where they can best be used throughout the development life cycle. Describes some of the issues involved in automating test execution. Explains why using capture / replay alone does not result in good automated tests.

Overview of Managing Testing

Estimating How to prioritise test effort across a system and prepare estimates.

Monitoring progress Explains a simple technique to help keep testing on track.

Effectiveness of testing A simple but powerful measure of the 'goodness' of the testing process enabling a prediction of corrective maintenance.

Booking & Information +44 (0)8702 406172
 courses@grove.co.uk
 www.grove.co.uk

 Grove
 consultants
 software testing