



Hewlett Packard Enterprise

Course Data Sheet

UFT120 – Unified Functional Testing 12.x Essentials

Course No.: UFT120-125	Category/Sub Category: Application Functional Testing/Unified Functional Testing-QTP
For software version(s): 12.5 Software version used in the labs: 12.51	Course length: Five days
Delivery formats: Instructor Led (ILT) and Virtual Instructor Led (VILT)	Training is available as a private session onsite.
To order visit: http://h20546.www2.hp.com/main/US/	

Course Description

This course provides a comprehensive understanding of how to use the Unified Functional Testing (UFT) 12.5 application as an automated functional testing tool. Beginning with record and playback, participants will learn how to create new automated tests. They then explore enhancements, including synchronization, checkpoints, parameterization, reusable actions, function libraries, and shared object repositories.

Included is an introduction to HP UFT for API testing, which contains an extensible framework for the construction and execution of functional tests of headless systems (systems that do not have a user interface). The course concludes with an introduction to LeanFT, a powerful, lightweight, developer-oriented testing tool.

Audience/Job Roles

Quality Assurance engineers or any new users of UFT

Course Objectives

Upon successful completion of this course, you should be able to:

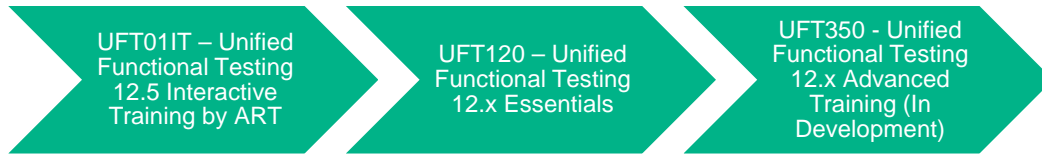
- Use UFT to automate GUI and API tests
- Add additional UFT features make the test robust and provide better coverage
- Use LeanFT to create tests

Prerequisites/Recommended Skills

To be successful in this course, you should have the following prerequisites or knowledge.

- Working knowledge of Windows and web browsers
- Any experience with programming or scripting languages

Learning Path



Certification

HPE AIS – HP0-M102 – HPE Unified Functional Testing 12.x Software

HPE ASE – HP0-M216P – Advanced HPE Unified Functional Testing 12.x Software

Course Topics

Modules	Objectives
Module 1: Course Overview	<ul style="list-style-type: none"> Identify the contents and objectives of the course Define the class schedules and class logistics Identify the related courses Discuss the lab environment details
Module 2: Software Overview	<ul style="list-style-type: none"> Describe the advantages of UFT as a testing tool Preview the UFT Start page and Help menus Recognize the sample applications used in the labs Identify resources for getting assistance
Module 3: Preparing to Record	<ul style="list-style-type: none"> Identify functional testing principles, and the benefits of automated testing Navigate the typical GUI testing workflow Document the steps of a business process Prioritize business processes using effective criteria Gather sufficient test data Prepare the test environment for automated testing
Module 4: Creating a Basic Test	<ul style="list-style-type: none"> Create a basic test from a manual test case Run a test and check for errors Save a test View test results
Module 5: Working with Objects	<ul style="list-style-type: none"> Identify objects Define a UFT for GUI Testing object Identify objects in UFT for GUI Testing Use the Object Repository to manage test objects in UFT for GUI Testing
Module 6: Utilizing a Shared Object Repository	<ul style="list-style-type: none"> Identify the types of object repositories Manage shared object repositories using the Object Repository Manager Use visual relation identifiers
Module 7: Adding Synchronization	<ul style="list-style-type: none"> Define synchronization in UFT for GUI Testing Identify the uses of synchronization in UFT for GUI Testing Add a synchronization step for a specified object
Module 8: Verifying with Standard Checkpoints	<ul style="list-style-type: none"> Define standard checkpoints Add standard checkpoints to a test Use a regular expression to add flexibility to a standard checkpoint

Modules	Objectives
Module 9: Using Parameters	<ul style="list-style-type: none"> Identify and use different parameter types Insert an input or output parameter Insert an output parameter Parameterize a checkpoint Evaluate test results for iterative tests
Module 10: Building Multiple, Reusable Actions	<ul style="list-style-type: none"> Identify actions in GUI testing Identify action types Identify action and test iterations Identify calls to existing actions and copies of actions Share values using the global data table Call actions with parameters Store action return values Create multiple actions from a single action Create a new action Call a reusable action from another test Use local and global data sheets Resolve missing actions
Module 11: Adding Steps Without Recording	<ul style="list-style-type: none"> List the types of steps that can be added to a test without using the record feature Use conditional statements in a test Use the Step Generator Use the reporter object to report events in the test results
Module 12: Creating Tests on a Web Application	<ul style="list-style-type: none"> Record and run a test on a web application Insert standard checkpoints on web objects Insert a text checkpoint in a test for a web application
Module 13: Testing web services with UFT API	<ul style="list-style-type: none"> Define Service Oriented Architecture (SOA) Identify components of SOA Define the terminology and principles surrounding component testing Navigate the UFT UI for API testing Create a basic Service Test (API Test) in UFT
Module 14: Using UFT API	<ul style="list-style-type: none"> Import a service Define test step inputs and outputs Identify data drive steps Work with controls flows Use the Results Viewer
Module 15: Enhancing UFT API Testing	<ul style="list-style-type: none"> Use checkpoints and reporting Configure database validation with UFT Configure security with UFT
Module 16: Introduction to LeanFT	<ul style="list-style-type: none"> Build a LeanFT web based test using Visual C# Build a LeanFT Windows based test using Visual C# Run the tests with NUnit
Module 17: Using the Object Identification Center	<ul style="list-style-type: none"> Use the Object Identification Center (OIC) to add steps Use the OIC to capture values from the AUT Output data to the Visual Studio log

Modules	Objectives
Module 18: Iterating with LeanFT	<ul style="list-style-type: none"> • Create an array • Set test iterations based on the TestCaseSource area of the LeanFT script • Create a loop for a section of a test • Launch HTML reports
Module 19: LeanFT Checkpoints	<ul style="list-style-type: none"> • Build checkpoints by inserting Assert methods for the framework being used • Report results to the test results • Control test pass/fail status • Use the Assert within a try-catch statement
Appendix A: Using Database Checkpoints	<ul style="list-style-type: none"> • Identify the purpose of a database checkpoint • Create a Structured Query Language (SQL) statement using Microsoft Query • Create a database checkpoint • Parameterize a database query
Appendix B: Significant Checkpoints for GUI Testing	<ul style="list-style-type: none"> • Create table checkpoints • Create test and text area checkpoints • Create file content checkpoints • Create XML checkpoints
Appendix C: Object Identification Techniques	<ul style="list-style-type: none"> • Configure object identification • Describe the mandatory and assistive properties • Use ordinal identifiers • Use smart identifiers • Describe when to use Smart Identification • Use the Smart Identification process • Describe how UFT for GUI Testing uses Smart Identification – Use Case Scenario
Appendix D: Using Recovery Scenarios	<ul style="list-style-type: none"> • Identify exceptions in a test • Create a recovery scenario • Associate a recovery scenario with a test • Set an optional step in a test