



TESTCASE TWO - PAYMENT PROCESS

TESTCASE TWO

Exercise 7a – Create a Library

Objective

By the end of this exercise, you will be able to create a Library of Reusable TestStepBlocks.

Why is this important?

A Library greatly reduces the amount of work needed to create the multiple TestCases that are needed to fully automate TestCases in the SUT.

Key elements:



Instructions

1. On the TestCases **Root Folder**, create a **TestStepLibrary**.
2. Drag and drop all the Folders from the TestCase in Ex. 6c into the **Library Folder**, except: "**Verification of Prices**".

Hints

- » Library folders are sorted alphabetically.

Exercise 7b - Create Reusable TestStepBlocks

Objective

By the end of this exercise, you will be able to use the Reusable TestStepBlocks to create References in a TestCase.

Why is this important?

This exercise will demonstrate how it is much more efficient to use a Library to create automated TestCases.

Instructions

1. On the TestCases **Root Folder**, create a new TestCaseFolder named **"TestCase 2"**.
2. Within your new Folder, create a subfolder named **"7b Create Reusable TestStepBlocks"**. Create a new TestCase named **"Payment Process"**. Add the **Test configuration parameter** named **"Browser"** with the value **"InternetExplorer"**.
 Drag the following Folders from the **Library**, and drop them into the TestCase:
 - Precondition
 - Order Product
3.
 - Start Checkout
 - Confirmation
 - Verification of Success
 - Postcondition
4. Create two new **TestStepFolders** and name them:
 - Checkout Process
 - Verification of Prices
 Reorder the Folders to follow the Web Shop process flow (see TestCase 1)
5. Navigate to **(Webshop>>Check out process)** the **Module Folder** named **"7b Payment Information Check Money Order"**
 Within this Folder, create a new **Module** of the Web Shop **"Payment Information"** page that appears when you select **"Check Money Order"** and click **"Continue"**
 Scan this page and add the controls:
 - **"Message"** table
 - **"Back"** button
 - **"Continue"** button
 Make sure the controls are uniquely identified.
 Name the Module **"Payment Information Check Money Order"** and save.
7. Add the following Modules into the **"Checkout Process"** TestStepFolder and rename them accordingly:
 - Billing Address
 - Shipping Address
 - Shipping Method
 - Payment Method
 - Payment Information Check Money Order
8. Add the **"Confirm Order"** Module into the **"Verification of Prices"** TestStepFolder and rename the TestStep **"Verification of Prices"**.
9. Enter the following Values for the new TestStepValues as per the following table:

TESTCASE TWO

TestStep	TestStepValue	Value	ActionMode
<i>TestStepFolder Checkout Process</i>			
Billing Address	Continue	X	Input
Shipping Address	Continue	X	Input
Shipping Method	Shipping Methods	Ground	Input
	Continue	X	Input
Payment Method	Payment Methods	Check / Money Order	Input
	Continue	X	Input
Payment Information Check Money Order	Continue	X	Input
<i>TestStepFolder Verification of Prices</i>			
Verification of Prices	Cart Total		Select
	Column: Rename to #2		Select
	Cell: Choose Sub-Total	{MATH[{B[PriceBluejeans]}*25]}	Verify DataType: Numeric
	Cell: Choose Sub-Total	SubTotal	Buffer
	Cell: Shipping:*	10.00	Verify DataType: Numeric
	Cell: Payment method additional fee:	5.00	Verify DataType: Numeric
	Cell: Total	Use the SubTotal Buffer to verify the total cost adding any shipping / additional fees.	Verify DataType: Numeric

10. Run the TestCase in the **ScratchBook**.
11. Set the TestCase WorkState to **"COMPLETED"**.