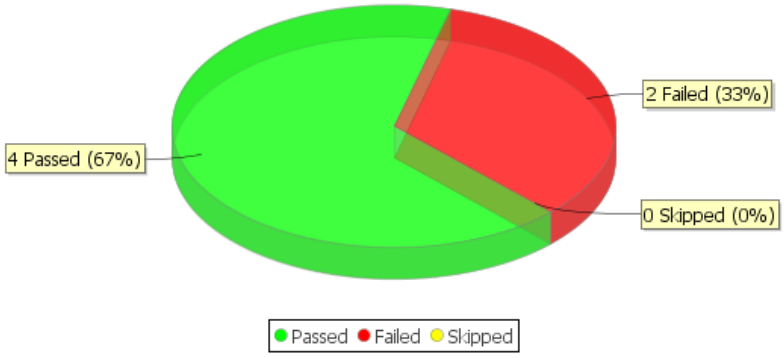
Test Run Report

Test Run Overview

Started	03 Jul 2019, 16:40:49
Ended	03 Jul 2019, 17:12:40
Duration (mm:ss)	31:50.903

Test Results Summary

Passed	4
Failed	2
Skipped	0



Test Case Summary

- Jamie spec Student Loan Project for all tax years — failed
  - Optimised Student loans1 Priority 1 Other — failed
    - A_StudentLoan1Priorty1Other_OctoberPayPeriod.testcase — failed

Failing Test Step because the Loop Limit has been reached.

One or more Sub-Test Steps failed.

The required connection name isn't present. This should come from the containing With Screen API.

The required connection name isn't present. This should come from the containing With Screen API.

    - B_StudentLoan1Priorty1Other_NovemberPayPeriod.testcase — failed

The row locator did not match any rows.

One or more Sub-Test Steps failed.

A Salesforce target is required. Target: com.provar.core.model.base.java.UiTargetValueImpl@3d15d74a

A Salesforce target is required. Target: com.provar.core.model.base.java.UiTargetValueImpl@3d15d74a

Error performing Ui Assert. Connection Name: Admin, cause: [NoSuchElementException: No elements matched the field locator. Field name: benefit For documentation on this error, please visit: http://seleniumhq.org/exceptions/no_such_element.html Build info: version: 'unknown', revision: 'unknown', time: 'unknown' System info: host: 'XCD_92WLKP2', ip: '192.168.1.103', os.name: 'Windows 10', os.arch: 'amd64', os.version: '10.0', java.version: '1.8.0_191' Driver info: driver.version: ProvarSeleniumDriver]

org.openqa.selenium.NoSuchElementException: No elements matched the field locator. Field name: benefit

For documentation on this error, please visit: http://seleniumhq.org/exceptions/no_such_element.html

Build info: version: 'unknown', revision: 'unknown', time: 'unknown'

System info: host: 'XCD_92WLKP2', ip: '192.168.1.103', os.name: 'Windows 10', os.arch: 'amd64', os.version: '10.0', java.version: '1.8.0_191'

Driver info: driver.version: ProvarSeleniumDriver

at com.provar.plugins.forcedotcom.core.ui.ProvarFindByElementLocator.findElement(ProvarFindByElementLocator.java:90)

at org.openqa.selenium.support.pagefactory.internal.LocatingElementHandler.invoke(LocatingElementHandler.java:38)

at com.sun.proxy.$Proxy104.getWrappedElement(Unknown Source)

at org.openqa.selenium.remote.internal.WebElementToJsonConverter.apply(WebElementToJsonConverter.java:50)

at com.google.common.collect.Iterators$7.transform(Iterators.java:750)

at com.google.common.collect.TransformedIterator.next(TransformedIterator.java:47)

at com.google.common.collect.Iterators.addAll(Iterators.java:367)

at com.google.common.collect.Lists.newArrayList(Lists.java:165)

at com.google.common.collect.Lists.newArrayList(Lists.java:150)

at org.openqa.selenium.remote.RemoteWebDriver.executeScript(RemoteWebDriver.java:571)

at com.provar.core.testapi.ProvarSeleniumDriver.scrollIntoView(ProvarSeleniumDriver.java:1778)

at com.provar.core.testapi.ProvarSeleniumDriver.scrollIntoView(ProvarSeleniumDriver.java:1763)

at com.provar.plugins.forcedotcom.core.ui.SeleniumUiHandler.assertFieldAttributes(SeleniumUiHandler.java:6260)

at com.provar.plugins.forcedotcom.core.ui.SeleniumUiHandler.performAssert(SeleniumUiHandler.java:6042)

at com.provar.plugins.forcedotcom.core.ui.SeleniumUiHandler.perform(SeleniumUiHandler.java:382)

at com.provar.plugins.forcedotcom.core.connection.SfConnectionHandler.delegateToUiHandler(SfConnectionHandler.java:356)

at com.provar.plugins.forcedotcom.core.connection.SfConnectionHandler.perform(SfConnectionHandler.java:345)

at com.provar.core.testexecution.common.java.TestExecutionContextImpl.doConnectionOperation(TestExecutionContextImpl.java:813)

at com.provar.plugins.forcedotcom.core.ui.UiAssert.execute(UiAssert.java:166)

at sun.reflect.GeneratedMethodAccessor111.invoke(Unknown Source)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)

at java.lang.reflect.Method.invoke(Method.java:498)

at com.provar.core.testexecution.container.java.TestExecutionContainerImpl.invokeTestApi(TestExecutionContainerImpl.java:688)

at com.provar.core.testexecution.container.java.LocalTestExecutionContainerRunner.doInvokeTestApi(LocalTestExecutionContainerRunner.java:627)

at com.provar.core.testexecution.container.java.LocalTestExecutionContainerRunner.access$10(LocalTestExecutionContainerRunner.java:573)

at com.provar.core.testexecution.container.java.LocalTestExecutionContainerRunner
$MessageProcessor.run(LocalTestExecutionContainerRunner.java:1180)

One or more Sub-Test Steps failed.

One or more Sub-Test Steps failed.

Optimized Student loans	successful
A_StudentLoan_OctoberPayPeriod.testcase	successful
B_StudentLoan_NovemberPayPeriod.testcase	successful
Optimized StudentLoansCTAEO	successful
A_StudentLoanAndCTAEO_OctoberPayPeriod.testcase	successful
B_StudentLoanAndCTAEO_NovemberPayPeriod.testcase	successful

A_StudentLoan1Priorty1Other_OctoberPayPeriod.testcase

Output

- Salesforce Connect: Admin (Test)
  Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test
- Click On Payroll Link
  - Sleep for {15} seconds
    Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
  - On PayrollTab page
    - Click the Payroll Tab
  - Process Payroll ID
    Loaded 1 rows from Parameter Value Source.
    - On SF xcdhcm__Payroll page with pageobject Xcdhcm__ Payroll
      - UI Assert
    - On SF Payroll Screen
      - Sleep for {10} seconds
        Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
      - Set the Search field to : StdntLoanPriorty1Other1_Payroll
      - On Payroll page
        - With PayrollTable row
          - Click the Pay Roll
- On SF xcdhcm__PayrollDetailView page with pageobject Xcdhcm__ Payroll Detail View
  - Sleep for {10} seconds
    Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
  - Click the Go To List button
    Background activity wait successful. Actual duration(seconds): 0.347
  - Navigate: Refresh
  - Sleep for {10} seconds
    Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
- On SF Payroll (UK Net Payroll) Pay Runs related list
  - With xcdhcm__XCD_PAY_Periods__r rows {7}
    Populated the row elements.
    The row locator matched 1 row.
    - Click the Payroll period field
- On SF xcdhcm__PayrollPeriod page with pageobject Xcdhcm__ Payroll Period
  - Navigate: Refresh
  - Sleep for {15} seconds
    Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
  - Uncheck the Payroll Set Up Un Select field
  - For Each: {ExcelData}=>payrun
    Loaded 1 rows from Parameter Value Source.
    - Sub Steps
      - With Employees rows {employeeName = $payrun.EmpName}
        - Check the Employee Namechk field
  - Click the Generate Draft button
  - UI Assert
  - Wait For: {DraftStatus.fields.draftStatus.value = "Draft"}=>{30}
    - Sub Steps
      - UI Assert
    - Sub Steps
      - UI Assert
    - Failing Test Step because the Loop Limit has been reached.
  - One or more Sub-Test Steps failed.
- If: {DraftStatus.fields.draftStatus.value = "Draft"}
  - Else
    - Navigate: Refresh

The required connection name isn't present. This should come from the containing With Screen API.

The required connection name isn't present. This should come from the containing With Screen API.

- Sleep for {20} seconds
  Timed sleep successful. Requested duration(seconds): 20, Actual duration(seconds): 20
- On SF xcdhcm__PayrollPeriod page with pageobject Xcdhcm__ Payroll Period
  - UI Assert
  - Wait For: {DraftStatus.fields.draftStatus.value = "Draft"}=>{30}
- If: {DraftStatus.fields.draftStatus.value = "Draft"}
  - Else
    - On SF xcdhcm__PayrollPeriod page with pageobject Xcdhcm__ Payroll Period
      - Click the Generate Draft button
      - UI Assert
      - Wait For: {DraftStatus.fields.draftStatus.value = "Draft"}=>{30}
    - If: {DraftStatus.fields.draftStatus.value = "Draft"}

B_StudentLoan1Priorty1Other_NovemberPayPeriod.testcase

Output

- Salesforce Connect: Admin (Test)
  Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test
- Click On Payroll Link
  - Sleep for {15} seconds
    Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
  - On PayrollTab page
    - Click the Payroll Tab
  - Process Payroll ID
    Loaded 1 rows from Parameter Value Source.
    - On SF xcdhcm__Payroll page with pageobject Xcdhcm__ Payroll
      - UI Assert
    - On SF Payroll Screen
      - Sleep for {10} seconds
        Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
      - Set the Search field to : StdntLoanPriorty1Other1_Payroll
      - On Payroll page
        - With PayrollTable row
          - Click the Pay Roll
- On SF xcdhcm__PayrollDetailView page with pageobject Xcdhcm__ Payroll Detail View
  - Sleep for {10} seconds
    Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
  - Click the Go To List button
    Background activity wait successful. Actual duration(seconds): 0.400
  - Navigate: Refresh
  - Sleep for {10} seconds
    Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
- On SF Payroll (UK Net Payroll) Pay Runs related list
  - With xcdhcm__XCD_PAY_Periods__r rows {8}
    Populated the row elements.

    The row locator matched 1 row.
    - Click the Payroll period field
- On SF xcdhcm__PayrollPeriod page with pageobject Xcdhcm__ Payroll Period
  - Navigate: Refresh
  - Sleep for {15} seconds
    Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
  - Uncheck the Payroll Set Up Un Select field
  - For Each: {ExcelData}=>payrun
    Loaded 1 rows from Parameter Value Source.
    - Sub Steps

- With Employees rows {employeeName = $payrun.EmpName}
  - Check the Employee Namechk field
- Click the Generate Draft button
- UI Assert
- Wait For: {DraftStatus.fields.draftStatus.value = "Draft"}=>{30}
- If: {DraftStatus.fields.draftStatus.value = "Draft"}
  - Then
    - Call Reports Script

      Loaded 1 rows from Parameter Value Source.

      - Call 4_StudentLoan1Priorty1Other_NovemberPayPeriod__ReportOne
        - 4_StudentLoan1Priorty1Other_NovemberPayPeriod__ReportOne.testcase
          - Salesforce Connect: Admin (Test)

            Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test

          - Reports Tab for all fields excluding TYPE Fields

            Loaded 1 rows from Parameter Value Source.

            - On MyPageObjectReportPage page
              - Click the Reports Tab

                Background activity wait successful. Actual duration(seconds): 0.281

              - Click the All Reports

                Background activity wait successful. Actual duration(seconds): 0.844

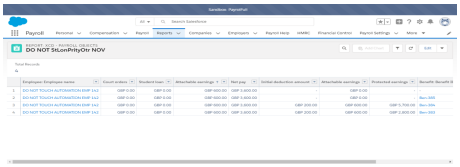
              - Set the Search All Reports to DO NOT StLonPrityOtr NOV

                Background activity wait successful. Actual duration(seconds): 0.270

              - Click the Report Selection

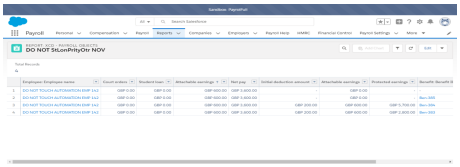
                Background activity wait successful. Actual duration(seconds): 1.257

            - On Report View Screen
              - For Each: {ReportData}=>rep
                - Sub Steps
                  - With DoNotTouchPayrollReport row
                    - Reading Actual Value
                      - Read Actual CourtOrder Value



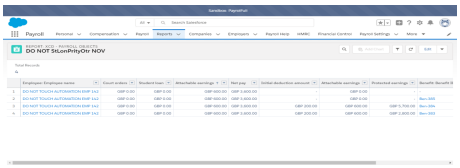
After Screenshot : 2019/07/03 16:51:26

                      - Read Actual StudentLoan Value



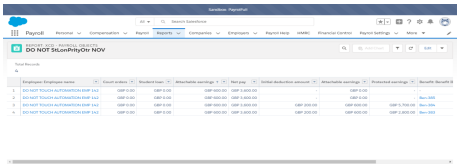
After Screenshot : 2019/07/03 16:51:27

                      - Read Actual Attachable Earnings One Value



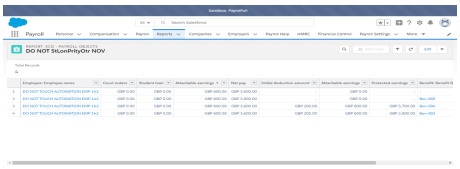
After Screenshot : 2019/07/03 16:51:28

                      - Read Actual InitialDeductionAmtOne Value



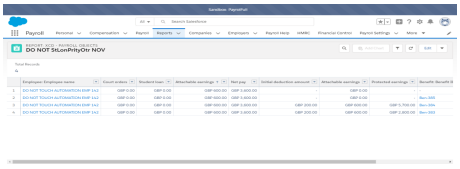
After Screenshot : 2019/07/03 16:51:29

                      - Read Actual attachableEarningsTwo Value



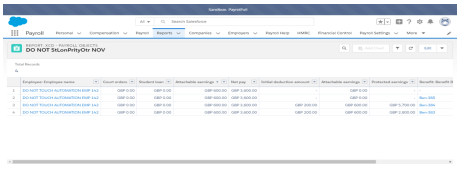
After Screenshot : 2019/07/03 16:51:30

- Read Actual ProtectedEarningsOne Value



After Screenshot : 2019/07/03 16:51:31

- Read Actual BenefitTwo Value

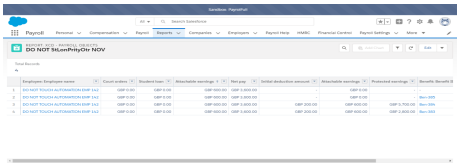


After Screenshot : 2019/07/03 16:51:32

- Setting Actual values for ExcelWrite
  - Set Values: empName
  - Set Values: O.CourtOrders
  - Set Values: O.StudentLoan
  - Set Values: O.AttachableEarnings1
  - Set Values: O.InitialDeductionAmount1
  - Set Values: O.AttachableEarnings2
  - Set Values: O.ProtectedEarnings1
  - Set Values: O.Benefit2
- Write: Actual Values
  Completed writing data to excel sheet: StudentLoans1Priority1Other
- ExpectedValueFromSheet-Oct
  Loaded 1 rows from Parameter Value Source.
  - Comparing actual and expected vlaues
    - If: {ExcelDataExpectedVal.CourtOrders = CourtOrderValue.fields.courtOrder.value}
      - Else
        - Set Values: O.CourtOrders
    - If: {ExcelDataExpectedVal.StudentLoan = studentLoanValue.fields.studentLoan.value}
      - Else
        - Set Values: O.StudentLoan
    - If: {ExcelDataExpectedVal.AttachableEarnings1 = attachableEarningsOneValue.fields.attachableEarnings1.value}
      - Else
        - Set Values: O.AttachableEarnings1
    - If: {ExcelDataExpectedVal.InitialDeductionAmount1 = InitialDeductionAmtOneValue.fields.initialDeductionAmount1.value}
      - Else
        - Set Values: O.InitialDeductionAmount1
    - If: {ExcelDataExpectedVal.AttachableEarnings2 = attachableEarningsTwoValue.fields.attachableEarnings2.value}
      - Else
        - Set Values: O.AttachableEarnings2
    - If: {ExcelDataExpectedVal.ProtectedEarnings1 = protectedEarningsOneValue.fields.protectedEarnings1.value}
      - Else
        - Set Values: O.ProtectedEarnings1
    - If: {ExcelDataExpectedVal.Benefit2 = benefitTwoValue.fields.benefit.value}
      - Then
        - Set Values: O.Benefit2
    - Write: Test Remarks

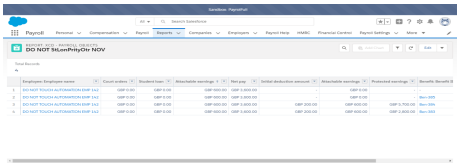
Completed writing data to excel sheet: StudentLoans1Priority1Other

- With DoNotTouchPayrollReport row
  - Reading Actual Value
    - Read Actual InitialDeductionAmtOne Value



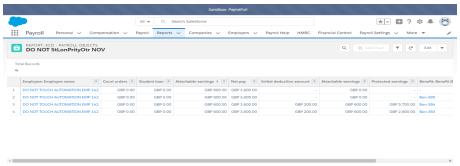
After Screenshot : 2019/07/03 16:51:36

    - Read Actual attachableEarningsTwo Value



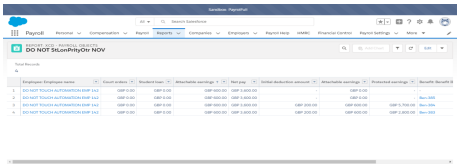
After Screenshot : 2019/07/03 16:51:37

    - Read Actual ProtectedEarningsOne Value



After Screenshot : 2019/07/03 16:51:38

    - Read Actual BenefitOne Value



After Screenshot : 2019/07/03 16:51:39

  - Setting Actual values for ExcelWrite
    - Set Values: empName
    - Set Values: N.InitialDeductionAmount2
    - Set Values: N.AttachableEarnings3
    - Set Values: N.ProtectedEarnings2
    - Set Values: N.Benefit1
  - Write: Actual Values

    Completed writing data to excel sheet: StudentLoans1Priority1Other

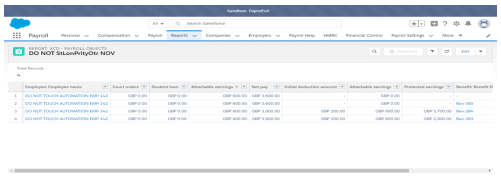
  - ExpectedValueFromSheet-Oct

    Loaded 1 rows from Parameter Value Source.

    - Comparing actual and expected vlaues
      - If: {ExcelDataExpectedVal.InitialDeductionAmount2 = InitialDeductionAmtTwoValue.fields.initialDeductionAmount1.value}
        - Then
          - Set Values: N.InitialDeductionAmount2
      - If: {ExcelDataExpectedVal.AttachableEarnings3 = attachableEarningsThreeValue.fields.attachableEarnings2.value}
        - Else
          - Set Values: N.AttachableEarnings3
      - If: {ExcelDataExpectedVal.ProtectedEarnings2 = protectedEarningsTwoValue.fields.protectedEarnings1.value}
        - Then
          - Set Values: N.ProtectedEarnings2
      - If: {ExcelDataExpectedVal.Benefit1 = benefitOneValue.fields.benefit.value}
        - Then
          - Set Values: N.Benefit1
      - Write: Test Remarks

        Completed writing data to excel sheet: StudentLoans1Priority1Other

- With DoNotTouchPayrollReport row

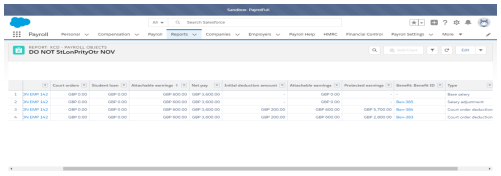


Error Screenshot

- The row locator did not match any rows.

- One or more Sub-Test Steps failed.

- On Report View Screen
  - UI Assert
    - A Salesforce target is required. Target: com.provar.core.model.base.java.UiTargetValueImpl@3d15d74a
    - A Salesforce target is required. Target: com.provar.core.model.base.java.UiTargetValueImpl@3d15d74a
  - Set Values: rowcount
  - Extracting Expected Value
    Loaded 1 rows from Parameter Value Source.
  - While: {true}
    - Sub Steps
      - With DoNotTouchPayrollReport row
        - Extracting current column values



Error Screenshot

- Error performing Ui Assert. Connection Name: Admin, cause: [NoSuchElementException: No elements matched the field locator. Field name: benefit For documentation on this error, please visit: http://seleniumhq.org/exceptions/no_such_element.html Build info: version: 'unknown', revision: 'unknown', time: 'unknown' System info: host: 'XCD_92WLKP2', ip: '192.168.1.103', os.name: 'Windows 10', os.arch: 'amd64', os.version: '10.0', java.version: '1.8.0_191' Driver info: driver.version: ProvarSeleniumDriver]

- org.openqa.selenium.NoSuchElementException: No elements matched the field locator. Field name: benefit

- For documentation on this error, please visit: http://seleniumhq.org/exceptions/no_such_element.html

- Build info: version: 'unknown', revision: 'unknown', time: 'unknown'

- System info: host: 'XCD_92WLKP2', ip: '192.168.1.103', os.name: 'Windows 10', os.arch: 'amd64', os.version: '10.0', java.version: '1.8.0_191'

- Driver info: driver.version: ProvarSeleniumDriver

- at com.provar.plugins.forcedotcom.core.ui.ProvarFindByElementLocator.findElement(ProvarFindByElementLocator.java:90)

- at org.openqa.selenium.support.pagefactory.internal.LocatingElementHandler.invoke(LocatingElementHandler.java:38)

- at com.sun.proxy.$Proxy104.getWrappedElement(Unknown Source)

- at org.openqa.selenium.remote.internal.WebElementToJsonConverter.apply(WebElementToJsonConverter.java:50)

- at com.google.common.collect.Iterators$7.transform(Iterators.java:750)

- at com.google.common.collect.TransformedIterator.next(TransformedIterator.java:47)

- at com.google.common.collect.Iterators.addAll(Iterators.java:367)

- at com.google.common.collect.Lists.newArrayList(Lists.java:165)

- at com.google.common.collect.Lists.newArrayList(Lists.java:150)

- at org.openqa.selenium.remote.RemoteWebDriver.executeScript(RemoteWebDriver.java:571)

- at com.provar.core.testapi.ProvarSeleniumDriver.scrollIntoView(ProvarSeleniumDriver.java:1778)

- at com.provar.core.testapi.ProvarSeleniumDriver.scrollIntoView(ProvarSeleniumDriver.java:1763)

- at com.provar.plugins.forcedotcom.core.ui.SeleniumUiHandler.assertFieldAttributes(SeleniumUiHandler.java:6260)

at com.provar.plugins.forcedotcom.core.ui.SeleniumUiHandler.performAssert(SeleniumUiHandler.java:6042)

at com.provar.plugins.forcedotcom.core.ui.SeleniumUiHandler.perform(SeleniumUiHandler.java:382)

at com.provar.plugins.forcedotcom.core.connection.SfConnectionHandler.delegateToUiHandler(SfConnectionHandler.java:356)

at com.provar.plugins.forcedotcom.core.connection.SfConnectionHandler.perform(SfConnectionHandler.java:345)

at com.provar.core.testexecution.common.java.TestExecutionContextImpl.doConnectionOperation(TestExecutionContextImpl.java:

at com.provar.plugins.forcedotcom.core.ui.UiAssert.execute(UiAssert.java:166)

at sun.reflect.GeneratedMethodAccessor111.invoke(Unknown Source)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)

at java.lang.reflect.Method.invoke(Method.java:498)

at com.provar.core.testexecution.container.java.TestExecutionContainerImpl.invokeTestApi(TestExecutionContainerImpl.java:688)

at com.provar.core.testexecution.container.java.LocalTestExecutionContainerRunner.doInvokeTestApi(LocalTestExecutionContain

at com.provar.core.testexecution.container.java.LocalTestExecutionContainerRunner.access$10(LocalTestExecutionContainerRunner.java:573)

at com.provar.core.testexecution.container.java.LocalTestExecutionContainerRunner$MessageProcessor.run(LocalTestExecutionContainerRunner.java:1180)

If: {colValues.fields.employeeName.value = ExcelData.AutomationEmployeeNames AND colValues.fields.type.value = "Court order payment to third party" OR colValues.fields.type.value = "Court order deduction"}

Else

If: {colValues.fields.type.value = "Student loan"}

One or more Sub-Test Steps failed.

Sub Steps

With DoNotTouchPayrollReport row

Extracting current column values

If: {colValues.fields.employeeName.value = ExcelData.AutomationEmployeeNames AND colValues.fields.type.value = "Court order payment to third party" OR colValues.fields.type.value = "Court order deduction"}

Else

If: {colValues.fields.type.value = "Student loan"}

Sub Steps

With DoNotTouchPayrollReport row

Extracting current column values

If: {colValues.fields.employeeName.value = ExcelData.AutomationEmployeeNames AND colValues.fields.type.value = "Court order payment to third party" OR colValues.fields.type.value = "Court order deduction"}

Then

Setting Actual values for ExcelWrite

If: {colValues.fields.benefit.value = "Ben-384"}

Then

If: {colValues.fields.type.value = "Court order payment to third party"}

If: {colValues.fields.type.value = "Court order deduction"}

Then

Set Values: L.TypeCODeductionAmount1

Write: Actual Values

Completed writing data to excel sheet: StudentLoans1Priority1Other

ExpectedValueFromSheet-Nov payroll period

Loaded 1 rows from Parameter Value Source.

Comparing actual and expected vlaues

If: {colValues.fields.type.value = "Court order payment to third party"}

If: {colValues.fields.type.value = "Court order deduction"}

Then

If: {ExcelDataExpectedVal.TypeCODeductionAmount1 = EmpName.columns.amount1.columnValues[LoopCounter]}
Else
Set Values: M.TypeCODeductionAmount1
Write: Test Remarks
Completed writing data to excel sheet: StudentLoans1Priority1Other

Sub Steps
With DoNotTouchPayrollReport row
Extracting current column values
If: {colValues.fields.employeeName.value = ExcelData.AutomationEmployeeNames AND colValues.fields.type.value = "Court order payment to third party" OR colValues.fields.type.value = "Court order deduction"}
Then
Setting Actual values for ExcelWrite
If: {colValues.fields.benefit.value = "Ben-384"}
Else
If: {colValues.fields.type.value = "Court order payment to third party"}
If: {colValues.fields.type.value = "Court order deduction"}
Then
Set Values: L.TypeCODeductionAmount2
Write: Actual Values
Completed writing data to excel sheet: StudentLoans1Priority1Other

ExpectedValueFromSheet-Nov payroll period
Loaded 1 rows from Parameter Value Source.

Comparing actual and expected vlaues
If: {colValues.fields.type.value = "Court order payment to third party"}
If: {colValues.fields.type.value = "Court order deduction"}
Then
If: {ExcelDataExpectedVal.TypeCODeductionAmount2 = EmpName.columns.amount1.columnValues[LoopCounter]}
Else
Set Values: M.TypeCODeductionAmount2
Write: Test Remarks
Completed writing data to excel sheet: StudentLoans1Priority1Other

One or more Sub-Test Steps failed.

Salesforce Connection closed successfully. Name: Admin

A_StudentLoan_OctoberPayPeriod.testcase

Output

Salesforce Connect: Admin (Test)
Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test

Click On Payroll Link
Sleep for {15} seconds
Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15

On PayrollTab page
Click the Payroll Tab
Process Payroll ID
Loaded 1 rows from Parameter Value Source.

On SF xcdhcm__Payroll page with pageobject Xcdhcm__ Payroll
UI Assert
On SF Payroll Screen
Sleep for {10} seconds
Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10

Set the Search field to : StudentLoan_Payroll

- On Payroll page
  - With PayrollTable row
    - Click the Pay Roll
- On SF xcdhcm__PayrollDetailView page with pageobject Xcdhcm__ Payroll Detail View
  - Sleep for {10} seconds
    Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
  - Click the Go To List button
    Background activity wait successful. Actual duration(seconds): 0.396
  - Navigate: Refresh
  - Sleep for {10} seconds
    Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
- On SF Payroll (UK Net Payroll) Pay Runs related list
  - With xcdhcm__XCD_PAY_Periods__r rows {7}
    Populated the row elements.

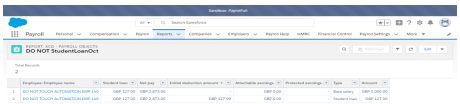
    The row locator matched 1 row.

    - Click the Payroll period field
- On SF xcdhcm__PayrollPeriod page with pageobject Xcdhcm__ Payroll Period
  - Navigate: Refresh
  - Sleep for {15} seconds
    Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
  - Uncheck the Payroll Set Up Un Select field
  - For Each: {ExcelData}=>payrun
    Loaded 1 rows from Parameter Value Source.

    - Sub Steps
      - With Employees rows {employeeName = $payrun.EmpName}
        - Check the Employee Namechk field
  - Click the Generate Draft button
  - UI Assert
  - Wait For: {DraftStatus.fields.draftStatus.value = "Draft"}=>{30}
- If: {DraftStatus.fields.draftStatus.value = "Draft"}
  - Then
    - Call Reports Script
      Loaded 1 rows from Parameter Value Source.

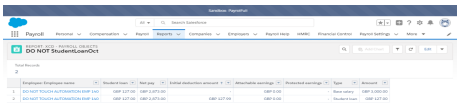
      - Call 2_StudentLoan_OctoberPayPeriod__ReportOne
        - 2_StudentLoan_OctoberPayPeriod__ReportOne.testcase
          - Salesforce Connect: Admin (Test)
            Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test
          - Reports Tab for all fields excluding TYPE Fields
            Loaded 1 rows from Parameter Value Source.

            - On MyPageObjectReportPage page
              - Click the Reports Tab
                Background activity wait successful. Actual duration(seconds): 0.310
              - Click the All Reports
                Background activity wait successful. Actual duration(seconds): 0.870
              - Set the Search All Reports to DO NOT StudentLoanOct
                Background activity wait successful. Actual duration(seconds): 0.269
              - Click the Report Selection
                Background activity wait successful. Actual duration(seconds): 1.282
            - On Report View Screen
              - For Each: {ReportData}=>rep
                - Sub Steps
                  - With DoNotTouchPayrollReport row
                    - Reading Actual Value
                      - Read Actual StudentLoan Value



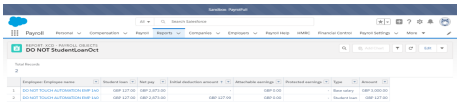
After Screenshot : 2019/07/03 16:56:40

- Read Actual Net Pay Value



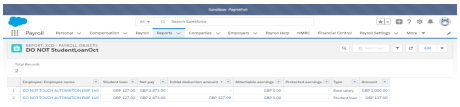
After Screenshot : 2019/07/03 16:56:41

- Read Actual InitialDeduction Value



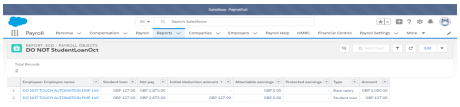
After Screenshot : 2019/07/03 16:56:42

- Read Actual attachableEarningsTwo Value



After Screenshot : 2019/07/03 16:56:42

- Read Actual ProtectedEarnings Value



After Screenshot : 2019/07/03 16:56:43

- Setting Actual values for ExcelWrite
  - Set Values: empName
  - Set Values: O.StudentLoan
  - Set Values: O.NetPay
  - Set Values: O.InitialDeductionAmount
  - Set Values: O.AttachableEarnings
  - Set Values: O.ProtectedEarnings
- Write: Actual Values
  Completed writing data to excel sheet: StudentLoans
- ExpectedValueFromSheet-Oct
  Loaded 1 rows from Parameter Value Source.
  - Comparing actual and expected vlaues
    - If: {ExcelDataExpectedVal.StudentLoan = studentloanValue.fields.studentLoan.value}
      - Then
        - Set Values: O.StudentLoan
    - If: {ExcelDataExpectedVal.NetPay = NePayValue.fields.netPay.value}
      - Then
        - Set Values: O.NetPay
    - If: {ExcelDataExpectedVal.InitialDeductionAmount = InitialDeductionValue.fields.intitialDeduction.value}
      - Then
        - Set Values: O.InitialDeductionAmount
    - If: {ExcelDataExpectedVal.AttachableEarnings = attachableEarningsValue.fields.attachableEarnings.value}
      - Else
        - Set Values: O.AttachableEarnings
    - If: {ExcelDataExpectedVal.ProtectedEarnings = protectedEarningsValue.fields.protectedEarnings.value}
      - Then
        - Set Values: O.ProtectedEarnings
    - Write: Test Remarks
      Completed writing data to excel sheet: StudentLoans

- Reports for only TYPE
  Loaded 1 rows from Parameter Value Source.
  - On Report View Screen
    - UI Assert
      - A Salesforce target is required. Target: com.provar.core.model.base.java.UiTargetValueImpl@19f286db
      - A Salesforce target is required. Target: com.provar.core.model.base.java.UiTargetValueImpl@19f286db
    - Set Values: rowcount
    - Extracting Expected Value
      Loaded 1 rows from Parameter Value Source.
    - While: {true}
      - Sub Steps
        - With DoNotTouchPayrollReport row
          - Extracting current column values
          - If: {colValues.fields.employeeName.value = ExcelData.AutomationEmployeeNames AND colValues.fields.type.value = "Student loan"}
      - Sub Steps
        - With DoNotTouchPayrollReport row
          - Extracting current column values
          - If: {colValues.fields.employeeName.value = ExcelData.AutomationEmployeeNames AND colValues.fields.type.value = "Student loan"}
            - Then
              - Reading Actual Value
              - Setting Actual values for ExcelWrite
                - If: {colValues.fields.type.value = "Student loan"}
                  - Then
                    - Set Values: L.TypeStudentLoan
              - Write: Actual Values
                Completed writing data to excel sheet: StudentLoans
              - ExpectedValueFromSheet-Nov payroll period
                Loaded 1 rows from Parameter Value Source.
              - Comparing actual and expected vlaues
                - If: {colValues.fields.type.value = "Student loan"}
                  - Then
                    - If: {ExcelDataExpectedVal.TypeStudentLoan = EmpName.columns.amount1.columnValues[LoopCounter]}
                      - Then
                        - Set Values: M.TypeStudentLoan
                        - Write: Test Remarks
                          Completed writing data to excel sheet: StudentLoans

Salesforce Connection closed successfully. Name: Admin

B_StudentLoan_NovemberPayPeriod.testcase

Output

- Salesforce Connect: Admin (Test)
  Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test
- Click On Payroll Link
  - Sleep for {15} seconds
    Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
  - On PayrollTab page
    - Click the Payroll Tab
  - Process Payroll ID
    Loaded 1 rows from Parameter Value Source.
    - On SF xcdhcm__Payroll page with pageobject Xcdhcm__ Payroll
      - UI Assert
    - On SF Payroll Screen

- Sleep for {10} seconds
  Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
- Set the Search field to : StudentLoan_Payroll
- On Payroll page
  - With PayrollTable row
    - Click the Pay Roll

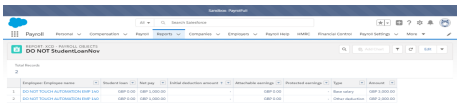
On SF xcdhcm__PayrollDetailView page with pageobject Xcdhcm__ Payroll Detail View
- Sleep for {10} seconds
  Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
- Click the Go To List button
  Background activity wait successful. Actual duration(seconds): 0.398
- Navigate: Refresh
- Sleep for {10} seconds
  Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10

On SF Payroll (UK Net Payroll) Pay Runs related list
- With xcdhcm__XCD_PAY_Periods__r rows {8}
  Populated the row elements.

  The row locator matched 1 row.
  - Click the Payroll period field

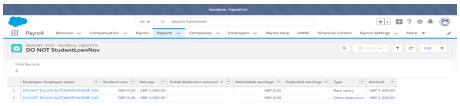
On SF xcdhcm__PayrollPeriod page with pageobject Xcdhcm__ Payroll Period
- Navigate: Refresh
- Sleep for {15} seconds
  Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
- Uncheck the Payroll Set Up Un Select field
- For Each: {ExcelData}=>payrun
  Loaded 1 rows from Parameter Value Source.
  - Sub Steps
    - With Employees rows {employeeName = $payrun.EmpName}
      - Check the Employee Namechk field
- Click the Generate Draft button
- UI Assert
- Wait For: {DraftStatus.fields.draftStatus.value = "Draft"}=>{30}
  - Sub Steps
    - UI Assert

If: {DraftStatus.fields.draftStatus.value = "Draft"}
- Then
  - Call Reports Script
    Loaded 1 rows from Parameter Value Source.
    - Call 4_StudentLoan_NovemberPayPeriod__ReportOne
      - 4_StudentLoan_NovemberPayPeriod__ReportOne.testcase
        - Salesforce Connect: Admin (Test)
          Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test
        - Reports Tab for all fields excluding TYPE Fields
          Loaded 1 rows from Parameter Value Source.
          - On MyPageObjectReportPage page
            - Click the Reports Tab
              Background activity wait successful. Actual duration(seconds): 0.320
            - Click the All Reports
              Background activity wait successful. Actual duration(seconds): 0.834
            - Set the Search All Reports to DO NOT StudentLoanNov
              Background activity wait successful. Actual duration(seconds): 0.267
            - Click the Report Selection
              Background activity wait successful. Actual duration(seconds): 3.563
          - On Report View Screen
            - For Each: {ReportData}=>rep
              - Sub Steps
                - With DoNotTouchPayrollReport row
                  - Reading Actual Value
                    - Read Actual StudentLoan Value



After Screenshot : 2019/07/03 17:02:11

- Read Actual Net Pay Value



After Screenshot : 2019/07/03 17:02:12

- Setting Actual values for ExcelWrite
  - Set Values: empName
  - Set Values: O.StudentLoan
  - Set Values: O.NetPay
- Write: Actual Values
  Completed writing data to excel sheet: StudentLoans
- ExpectedValueFromSheet-Oct
  Loaded 1 rows from Parameter Value Source.
  - Comparing actual and expected vlaues
    - If: {ExcelDataExpectedVal.StudentLoan = studentloanValue.fields.studentLoan.value}
      - Then
        - Set Values: O.StudentLoan
    - If: {ExcelDataExpectedVal.NetPay = NePayValue.fields.netPay.value}
      - Else
        - Set Values: O.NetPay
    - Write: Test Remarks
      Completed writing data to excel sheet: StudentLoans

Salesforce Connection closed successfully. Name: Admin

A_StudentLoanAndCTAEO_OctoberPayPeriod.testcase

Output

- Salesforce Connect: Admin (Test)
  Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test
- Click On Payroll Link
  - Sleep for {15} seconds
    Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
  - On PayrollTab page
    - Click the Payroll Tab
  - Process Payroll ID
    Loaded 1 rows from Parameter Value Source.
    - On SF xcdhcm__Payroll page with pageobject Xcdhcm__ Payroll
      - UI Assert
    - On SF Payroll Screen
      - Sleep for {10} seconds
        Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
      - Set the Search field to : StdntLoanNCTAEO_Payroll
      - On Payroll page
        - With PayrollTable row
          - Click the Pay Roll
- On SF xcdhcm__PayrollDetailView page with pageobject Xcdhcm__ Payroll Detail View
  - Sleep for {10} seconds
    Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
  - Click the Go To List button

Background activity wait successful. Actual duration(seconds): 0.265

- Navigate: Refresh
- Sleep for {10} seconds
  Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10

On SF Payroll (UK Net Payroll) Pay Runs related list
- With xcdhcm__XCD_PAY_Periods__r rows {7}
  Populated the row elements.

  The row locator matched 1 row.

  - Click the Payroll period field

On SF xcdhcm__PayrollPeriod page with pageobject Xcdhcm__ Payroll Period
- Navigate: Refresh
- Sleep for {15} seconds
  Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
- Uncheck the Payroll Set Up Un Select field
- For Each: {ExcelData}=>payrun
  Loaded 1 rows from Parameter Value Source.

  - Sub Steps
    - With Employees rows {employeeName = $payrun.EmpName}
      - Check the Employee Namechk field
- Click the Generate Draft button
- UI Assert
- Wait For: {DraftStatus.fields.draftStatus.value = "Draft"}=>{30}

If: {DraftStatus.fields.draftStatus.value = "Draft"}
- Then
  - Call Reports Script
    Loaded 1 rows from Parameter Value Source.

    - Call 2_StudentLoanAndCTAEO_OctoberPayPeriod__ReportOne
      - 2_StudentLoanAndCTAEO_OctoberPayPeriod__ReportOne.testcase
        - Salesforce Connect: Admin (Test)
          Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test

        - Reports Tab for all fields excluding TYPE Fields
          Loaded 1 rows from Parameter Value Source.

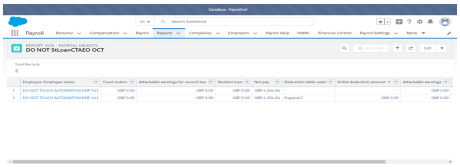
          - On MyPageObjectReportPage page
            - Click the Reports Tab
              Background activity wait successful. Actual duration(seconds): 0.279

            - Click the All Reports
              Background activity wait successful. Actual duration(seconds): 0.808

            - Set the Search All Reports to DO NOT StLoanCTAEO OCT
              Background activity wait successful. Actual duration(seconds): 0.267

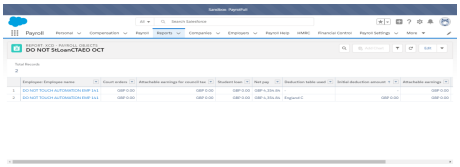
            - Click the Report Selection
              Background activity wait successful. Actual duration(seconds): 3.528

          - On Report View Screen
            - For Each: {ReportData}=>rep
              - Sub Steps
                - With DoNotTouchPayrollReport row
                  - Reading Actual Value
                    - Read Actual CourtOrder Value



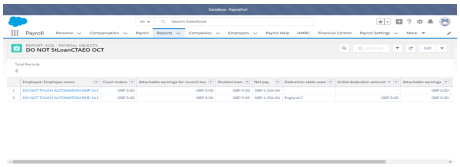
After Screenshot : 2019/07/03 17:06:57

- Read Actual AttachableEarningsForCouncilTax Value



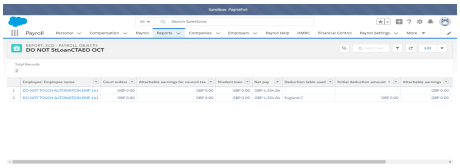
After Screenshot : 2019/07/03 17:06:58

- Read Actual StudentLoan Value



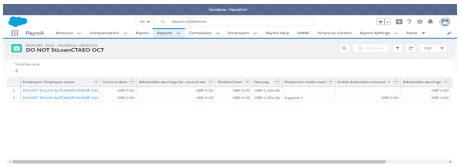
After Screenshot : 2019/07/03 17:06:58

- Read Actual Net Pay Value



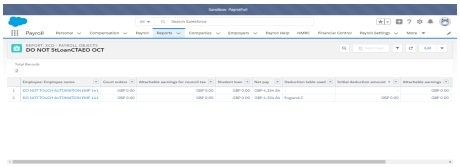
After Screenshot : 2019/07/03 17:06:59

- Read DeductionTable Value



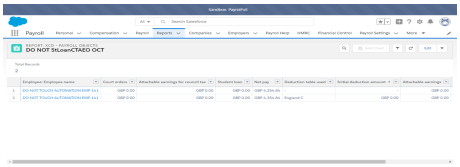
After Screenshot : 2019/07/03 17:07:00

- Read Actual InitialDeduction Value



After Screenshot : 2019/07/03 17:07:01

- Read Actual attachableEarningsTwo Value



After Screenshot : 2019/07/03 17:07:02

- Setting Actual values for ExcelWrite
  - Set Values: empName
  - Set Values: O.CourtOrders
  - Set Values: O.AttachableEarningsForCouncilTax
  - Set Values: O.StudentLoan
  - Set Values: O.NetPay
  - Set Values: O.DeductionTableUsed
  - Set Values: O.InitialDeductionAmount
  - Set Values: O.AttachableEarningsTwo
- Write: Actual Values

  Completed writing data to excel sheet: StudentLoansCTAEO
- ExpectedValueFromSheet-Oct

  Loaded 1 rows from Parameter Value Source.
  - Comparing actual and expected vlaues
    - If: {ExcelDataExpectedVal.CourtOrders = CourtOrderValue.fields.courtOrder.value}
      - Else
        - Set Values: O.CourtOrders
    - If: {ExcelDataExpectedVal.AttachableEarningsForCouncilTax = AttahableEarningsValueCT.fields.attachableEarningsForCouncilTax.value}
      - Else
        - Set Values: O.AttachableEarningsForCouncilTax
    - If: {ExcelDataExpectedVal.StudentLoan = studentloanValue.fields.studentLoan.value}
      - Then

- Set Values: O.StudentLoan
- If: {ExcelDataExpectedVal.NetPay = NePayValue.fields.netPay.value}
  - Else
    - Set Values: O.NetPay
- If: {ExcelDataExpectedVal.DeductionTableUsed = deductionTableValue.fields.deductionTableUsed.value}
  - Then
    - Set Values: O.DeductionTableUsed
- If: {ExcelDataExpectedVal.InitialDeductionAmount = InitialDeductionValue.fields.intitialDeduction.value}
  - Else
    - Set Values: O.InitialDeductionAmount
- If: {ExcelDataExpectedVal.AttachableEarningsTwo = attachableEarningsTwoValue.fields.attachableEarningsTwo.value}
  - Else
    - Set Values: O.AttachableEarningsTwo
- Write: Test Remarks
  Completed writing data to excel sheet: StudentLoansCTAEO

Reports for only TYPE
Loaded 1 rows from Parameter Value Source.

- On Report View Screen
  - UI Assert
    - A Salesforce target is required. Target: com.provar.core.model.base.java.UiTargetValueImpl@3ec7040b
    - A Salesforce target is required. Target: com.provar.core.model.base.java.UiTargetValueImpl@3ec7040b
  - Set Values: rowcount
  - Extracting Expected Value
    Loaded 1 rows from Parameter Value Source.
  - While: {true}
    - Sub Steps
      - With DoNotTouchPayrollReport row
        - Extracting current column values
        - If: {colValues.fields.employeeName.value = ExcelData.AutomationEmployeeNames AND colValues.fields.type.value = "Court order administrative cost" OR colValues.fields.type.value = "Court order payment to third party" OR colValues.fields.type.value = "Court order deduction"}
    - Sub Steps
      - With DoNotTouchPayrollReport row
        - Extracting current column values
        - If: {colValues.fields.employeeName.value = ExcelData.AutomationEmployeeNames AND colValues.fields.type.value = "Court order administrative cost" OR colValues.fields.type.value = "Court order payment to third party" OR colValues.fields.type.value = "Court order deduction"}
          - Then
            - Reading Actual Value
            - Setting Actual values for ExcelWrite
              - If: {colValues.fields.type.value = "Court order payment to third party"}
              - If: {colValues.fields.type.value = "Court order deduction"}
                - Then
                  - Set Values: L.TypeCODeductionAmount
            - Write: Actual Values
              Completed writing data to excel sheet: StudentLoansCTAEO
            - ExpectedValueFromSheet-Nov payroll period
              Loaded 1 rows from Parameter Value Source.
            - Comparing actual and expected vlaues
              - If: {colValues.fields.type.value = "Court order payment to third party"}
              - If: {colValues.fields.type.value = "Court order deduction"}
                - Then
                  - If: {ExcelDataExpectedVal.TypeCODeductionAmount = EmpName.columns.amount1.columnValues[LoopCounter]}
                    - Else
                      - Set Values: M.TypeCODeductionAmount
                      - Write: Test Remarks

Completed writing data to excel sheet: StudentLoansCTAEO

Salesforce Connection closed successfully. Name: Admin

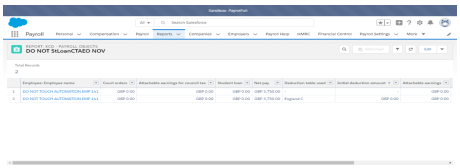
B_StudentLoanAndCTAEO_NovemberPayPeriod.testcase

Output

- Salesforce Connect: Admin (Test)
  Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test
- Click On Payroll Link
  - Sleep for {15} seconds
    Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
  - On PayrollTab page
    - Click the Payroll Tab
  - Process Payroll ID
    Loaded 1 rows from Parameter Value Source.
    - On SF xcdhcm__Payroll page with pageobject Xcdhcm__ Payroll
      - UI Assert
    - On SF Payroll Screen
      - Sleep for {10} seconds
        Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
      - Set the Search field to : StdntLoanNCTAEO_Payroll
      - On Payroll page
        - With PayrollTable row
          - Click the Pay Roll
- On SF xcdhcm__PayrollDetailView page with pageobject Xcdhcm__ Payroll Detail View
  - Sleep for {10} seconds
    Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
  - Click the Go To List button
    Background activity wait successful. Actual duration(seconds): 0.257
  - Navigate: Refresh
  - Sleep for {10} seconds
    Timed sleep successful. Requested duration(seconds): 10, Actual duration(seconds): 10
- On SF Payroll (UK Net Payroll) Pay Runs related list
  - With xcdhcm__XCD_PAY_Periods__r rows {8}
    Populated the row elements.

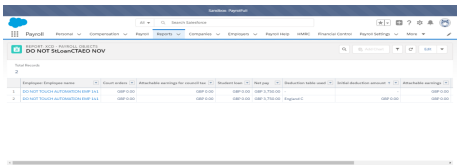
    The row locator matched 1 row.
    - Click the Payroll period field
- On SF xcdhcm__PayrollPeriod page with pageobject Xcdhcm__ Payroll Period
  - Navigate: Refresh
  - Sleep for {15} seconds
    Timed sleep successful. Requested duration(seconds): 15, Actual duration(seconds): 15
  - Uncheck the Payroll Set Up Un Select field
  - For Each: {ExcelData}=>payrun
    Loaded 1 rows from Parameter Value Source.
    - Sub Steps
      - With Employees rows {employeeName = $payrun.EmpName}
        - Check the Employee Namechk field
  - Click the Generate Draft button
  - UI Assert
  - Wait For: {DraftStatus.fields.draftStatus.value = "Draft"}=>{30}
    - Sub Steps
      - UI Assert
- If: {DraftStatus.fields.draftStatus.value = "Draft"}
  - Then
    - Call Reports Script
      Loaded 1 rows from Parameter Value Source.
      - Call 4_StudentLoanAndCTAEO_NovemberPayPeriod__ReportOne

- 4_StudentLoanAndCTAEO_NovemberPayPeriod__ReportOne.testcase
  - Salesforce Connect: Admin (Test)
    Salesforce connection opened successfully. Connection name: Admin, ResultName: Admin, resultScope: Test
  - Reports Tab for all fields excluding TYPE Fields
    Loaded 1 rows from Parameter Value Source.
    - On MyPageObjectReportPage page
      - Click the Reports Tab
        Background activity wait successful. Actual duration(seconds): 0.274
      - Click the All Reports
        Background activity wait successful. Actual duration(seconds): 0.887
      - Set the Search All Reports to DO NOT StLoanCTAEO NOV
        Background activity wait successful. Actual duration(seconds): 0.268
      - Click the Report Selection
        Background activity wait successful. Actual duration(seconds): 0.257
    - On Report View Screen
      - For Each: {ReportData}=>rep
        - Sub Steps
          - With DoNotTouchPayrollReport row
            - Reading Actual Value
              - Read Actual CourtOrder Value



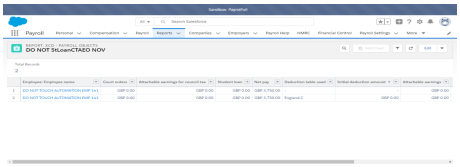
After Screenshot : 2019/07/03 17:12:21

              - Read Actual AttachableEarningsForCouncilTax Value



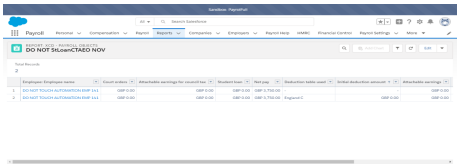
After Screenshot : 2019/07/03 17:12:22

              - Read Actual StudentLoan Value



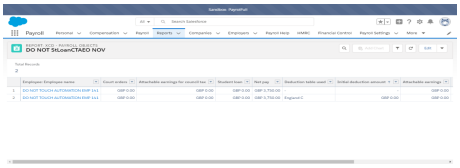
After Screenshot : 2019/07/03 17:12:22

              - Read Actual Net Pay Value



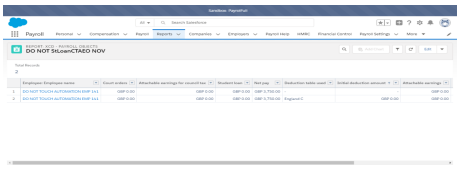
After Screenshot : 2019/07/03 17:12:24

              - Read Actual InitialDeduction Value



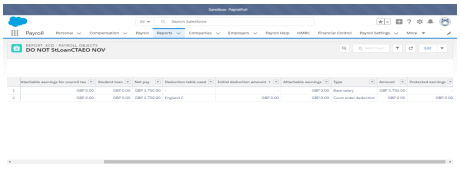
After Screenshot : 2019/07/03 17:12:25

              - Read Actual attachableEarningsTwo Value



After Screenshot : 2019/07/03 17:12:25

- Read Actual ProtectedEarnings Value



After Screenshot : 2019/07/03 17:12:26

- Setting Actual values for ExcelWrite
  - Set Values: empName
  - Set Values: O.CourtOrders
  - Set Values: O.AttachableEarningsForCouncilTax
  - Set Values: O.StudentLoan
  - Set Values: O.NetPay
  - Set Values: O.InitialDeductionAmount
  - Set Values: O.AttachableEarningsTwo
  - Set Values: O.ProtectedEarnings
- Write: Actual Values
  Completed writing data to excel sheet: StudentLoansCTAEO
- ExpectedValueFromSheet-Oct
  Loaded 1 rows from Parameter Value Source.
  - Comparing actual and expected vlaues
    - If: {ExcelDataExpectedVal.CourtOrders = CourtOrderValue.fields.courtOrder.value}
      - Then
        - Set Values: O.CourtOrders
    - If: {ExcelDataExpectedVal.AttachableEarningsForCouncilTax = AttahableEarningsValueCT.fields.attachableEarningsForCouncilTax.value}
      - Else
        - Set Values: O.AttachableEarningsForCouncilTax
    - If: {ExcelDataExpectedVal.StudentLoan = studentloanValue.fields.studentLoan.value}
      - Else
        - Set Values: O.StudentLoan
    - If: {ExcelDataExpectedVal.NetPay = NePayValue.fields.netPay.value}
      - Else
        - Set Values: O.NetPay
    - If: {ExcelDataExpectedVal.InitialDeductionAmount = InitialDeductionValue.fields.intitialDeduction.value}
      - Else
        - Set Values: O.InitialDeductionAmount
    - If: {ExcelDataExpectedVal.AttachableEarningsTwo = attachableEarningsTwoValue.fields.attachableEarningsTwo.value}
      - Else
        - Set Values: O.AttachableEarningsTwo
    - If: {ExcelDataExpectedVal.ProtectedEarnings = protectedEarningsValue.fields.protectedEarnings.value}
      - Then
        - Set Values: O.ProtectedEarnings
    - Write: Test Remarks
      Completed writing data to excel sheet: StudentLoansCTAEO
- Reports for only TYPE
  Loaded 1 rows from Parameter Value Source.
  - On Report View Screen
    - UI Assert
      - A Salesforce target is required. Target: com.provar.core.model.base.java.UiTargetValueImpl@3969288e

A Salesforce target is required. Target: com.provar.core.model.base.java.UiTargetValueImpl@3969288e

- Set Values: rowcount
- Extracting Expected Value
  Loaded 1 rows from Parameter Value Source.
- While: {true}
  - Sub Steps
    - With DoNotTouchPayrollReport row
      - Extracting current column values
      - If: {colValues.fields.type.value = "Student loan"}
  - Sub Steps
    - With DoNotTouchPayrollReport row
      - Extracting current column values
      - If: {colValues.fields.type.value = "Student loan"}

Salesforce Connection closed successfully. Name: Admin