Class Reference
Tests.LegacyConversionTest

Server: **epictetus.atomicobject.local**
Instance: **ENSEMBLE**
User: **UnknownUser**

DocBook | Search

[DEVCON_GEN] > [Tests] > [LegacyConversionTest]   ☐ **Private** ☐ **Storage**

class **Tests.LegacyConversionTest** extends UnitTest.ProductionTestHelper

▶ **Inventory**

▶ **Summary**

▶ **Parameters**

▼ **Methods**

• method **ExpectedLineOutput(id As %String)** as %String

Generates expected line output for a person

```
Set request = ##class(DataModel.Person).%OpenId(id)
Quit:request="" ""
Quit request.FirstName_","_request.LastName_","_request.Address_","_request.State_","_request.(
```

• method **PeopleFilePath()** as %String

Retrieves path to 'People.csv' file

```
Do ..GetSetting(..#PRODUCTION,"PersonFileOperation","FileName",.value)
Quit ##class(%SYS.System).TempDirectory()_"/"_value
```

• method **TestLegacyConversion()**

A full system test of the production.

1. Populate two random entries of 'legacy' data
2. Wait for the call interval for the 'LegacyPersonService'
3. Assert that the message goes from the 'LegacyPersonService' to the 'PersonFileOperation'
4. Verify that 'People.csv' has been written out
5. Load 'People.csv' into a stream
6. Generate expected line output from first and second legacy records and check them against the file
7. Verify that stream has reached the end of the file

```
Do ##class(External.LegacyData).Populate(2)

Do ..WaitForCallInterval("LegacyPersonService")

Do ..AssertMessageTrace("LegacyPersonService", "PersonFileOperation")

Do ..AssertFileExists(..PeopleFilePath())

Do ..LoadFileIntoStream(..PeopleFilePath(), .stream)

Do ..AssertLineInStream(..ExpectedLineOutput(1), .stream)

Do ..AssertLineInStream(..ExpectedLineOutput(2), .stream)

Do $$$AssertTrue(stream.AtEnd)
```