

Test Cases

## Test Case ID:

### Test Case Title:

#### Test Case Description:

#### Test Steps:

----------------------------------------

## Test Case ID: \*\* TC-1

### Test Case Title:

\*\* Valid Data - Basic URL and Filename Cleaning

#### Test Case Description:

\*\*  
This test case verifies that the dbt model correctly cleans a sample URL and filename containing the ".URL" extension and "&KEY&" string. This represents the happy path where the input data adheres to the expected format.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with valid placeholder data including a filename with the ".URL" extension and a URL containing "&KEY&".
* Example data: `document\_id = '123', filename = 'Document1.URL', xstring = 'https://example.com/document&KEY&'`
* Step 2: Execute the dbt model.
* Step 3: Query the output table created by the dbt model.
* Step 4: Assert that the `filename` column contains 'Document1' (the ".URL" extension has been removed).
* Step 5: Assert that the `url` column contains 'https://example.com/document' (the "&KEY&" string has been removed).
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-2

### Test Case Title:

\*\* Filename Cleaning - No ".URL" Extension

#### Test Case Description:

\*\*  
This test case checks the scenario where the input filename does not contain the ".URL" extension. The model should not modify the filename in this case.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with a filename that does \*not\* have the ".URL" extension.
* Example data: `document\_id = '456', filename = 'Document2', xstring = 'https://example.com/document&KEY&'`
* Step 2: Execute the dbt model.
* Step 3: Query the output table created by the dbt model.
* Step 4: Assert that the `filename` column still contains 'Document2' (no change occurred).
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-3

### Test Case Title:

\*\* URL Cleaning - No "&KEY&" String

#### Test Case Description:

\*\*  
This test case verifies that the dbt model handles URLs that do not contain the "&KEY&" string correctly. The model should not modify the URL in this case.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with a URL that does \*not\* contain the "&KEY&" string.
* Example data: `document\_id = '789', filename = 'Document3.URL', xstring = 'https://example.com/document'`
* Step 2: Execute the dbt model.
* Step 3: Query the output table created by the dbt model.
* Step 4: Assert that the `url` column still contains 'https://example.com/document' (no change occurred).
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-4

### Test Case Title:

\*\* Valid Data - Multiple Occurrences of "&KEY&" in URL

#### Test Case Description:

\*\*  
This test case ensures that the dbt model removes \*all\* occurrences of the "&KEY&" string from the URL, not just the first one.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with a URL containing multiple instances of "&KEY&".
* Example data: `document\_id = '101', filename = 'Document4.URL', xstring = 'https://example.com/document&KEY&more&KEY&'`
* Step 2: Execute the dbt model.
* Step 3: Query the output table created by the dbt model.
* Step 4: Assert that the `url` column contains 'https://example.com/documentmore' (all "&KEY&" strings have been removed).
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-5

### Test Case Title:

\*\* Case Sensitivity - ".url" Extension (lowercase)

#### Test Case Description:

\*\*  
This test case tests whether the model is case-sensitive when removing the ".URL" extension. It uses a lowercase ".url" extension. This is an important consideration for consistent data cleaning.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with a filename containing a lowercase ".url" extension.
* Example data: `document\_id = '202', filename = 'Document5.url', xstring = 'https://example.com/document&KEY&'`
* Step 2: Execute the dbt model.
* Step 3: Query the output table created by the dbt model.
* Step 4: Assert that the `filename` column contains 'Document5' (the ".url" extension has been removed, confirming case-insensitivity, if that's desired. Otherwise, test should assert that the filename contains 'Document5.url').
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-6

### Test Case Title:

\*\* Empty Filename

#### Test Case Description:

\*\*  
This test case verifies the model's behavior when the input filename is empty.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with an empty filename.
* Example data: `document\_id = '303', filename = '', xstring = 'https://example.com/document&KEY&'`
* Step 2: Execute the dbt model.
* Step 3: Query the output table created by the dbt model.
* Step 4: Assert that the `filename` column is also empty (or NULL, depending on the desired behavior).
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-7

### Test Case Title:

\*\* Empty URL

#### Test Case Description:

\*\*  
This test case checks the model's behavior when the input URL is empty.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with an empty URL.
* Example data: `document\_id = '404', filename = 'Document6.URL', xstring = ''`
* Step 2: Execute the dbt model.
* Step 3: Query the output table created by the dbt model.
* Step 4: Assert that the `url` column is also empty (or NULL, depending on the desired behavior).
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-8

### Test Case Title:

\*\* Filename with Multiple Periods Before .URL

#### Test Case Description:

\*\*  
This test case aims to capture edge cases in filenames that might complicate the substring removal logic.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with a filename containing multiple periods before '.URL'.
* Example data: `document\_id = '505', filename = 'Document.Part1.Part2.URL', xstring = 'https://example.com/document&KEY&'`
* Step 2: Execute the dbt model.
* Step 3: Query the output table created by the dbt model.
* Step 4: Assert that the `filename` column contains 'Document.Part1.Part2'.
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-9

### Test Case Title:

\*\* Very Long URL (Exceeding Typical Lengths)

#### Test Case Description:

\*\*  
This test case evaluates model handling of long URLs, ensuring that performance or size limits do not affect the URL cleaning process.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with a URL exceeding typical length limits. The URL should contain '&KEY&'.
* Example data: `document\_id = '606', filename = 'Document7.URL', xstring = 'https://example.com/' + 'a' \* 2000 + '&KEY&'` (This creates a URL of 2000 "a"s)
* Step 2: Execute the dbt model.
* Step 3: Query the output table created by the dbt model.
* Step 4: Assert that the `url` column contains the long URL with the '&KEY&' string removed. Verify the length and beginning of the URL are correct.
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-10

### Test Case Title:

\*\* Filename Starting with '.URL'

#### Test Case Description:

\*\*  
This test case tests the scenario when the input filename is just ".URL".  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with filename equal to ".URL".
* Example data: `document\_id = '707', filename = '.URL', xstring = 'https://example.com/document&KEY&'`
* Step 2: Execute the dbt model.
* Step 3: Query the output table.
* Step 4: Assert that the `filename` is empty (or NULL), depending on desired behavior.
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-11

### Test Case Title:

\*\* URL consisting only of "&KEY&"

#### Test Case Description:

\*\*  
This tests the scenario where the entire URL is just the "&KEY&" string.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define `source\_sap\_documents` CTE where xstring = "&KEY&".
* Example data: `document\_id = '808', filename = 'Document8.URL', xstring = '&KEY&'`
* Step 2: Execute the dbt model
* Step 3: Query the output table.
* Step 4: Assert the `url` column is empty (or NULL), depending on desired behavior.
* \*\*

----------------------------------------

## Test Case ID: \*\* TC-12

### Test Case Title:

\*\* Null Values in Filename and Xstring (URL)

#### Test Case Description:

\*\* This test verifies how the model handles scenarios where the input data contains NULL values for the filename and the Xstring (URL). Proper NULL handling is crucial for data integrity.  
\*\*

#### Test Steps:

* \*\*
* Step 1: Define a `source\_sap\_documents` CTE with `filename` and `xstring` both set to NULL. Include a `document\_id` for reference.
* Example data: `document\_id = '909', filename = NULL, xstring = NULL`
* Step 2: Execute the dbt model.
* Step 3: Query the output table created by the dbt model.
* Step 4: Assert that the resulting `filename` column is NULL.
* Step 5: Assert that the resulting `url` column is NULL.

----------------------------------------