

# Upload Data in Web Dynpro Table from a Text File with Delimiters



## Applies to:

SAP NetWeaver Web Dynpro. For more information, visit the [Portal and Collaboration homepage](#).

## Summary

This article uploads the data from a text file with delimiters to a Web Dynpro table UI element.

**Author:** Nikhil Tapkir

**Company:** L & T Infotech

**Created on:** 09 November 2009

## Author Bio

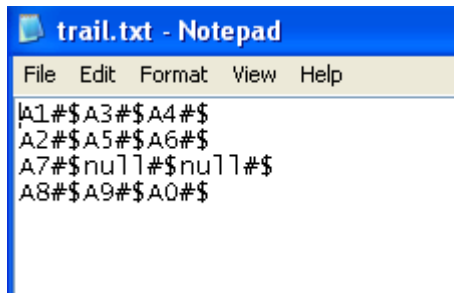
Nikhil Tapkir works as an NetWeaver Consultant in L & T Infotech and has an experience of two years.

## Table of Contents

Application Details and Requirements:.....	3
Steps to Create the Application .....	3
Application Results .....	9
Related Content.....	10
Disclaimer and Liability Notice.....	11

## Application Details and Requirements:

This application uploads the data from a flat file i.e. a text file and uploads the data into the webdynpro Table UI element. In short the application will read data from the text file. The text file should have the following format the columns values should be separated by delimiters and each line should contain a single record if the data is to be uploaded into a table. Below the screen shot has been attached for the flat file.



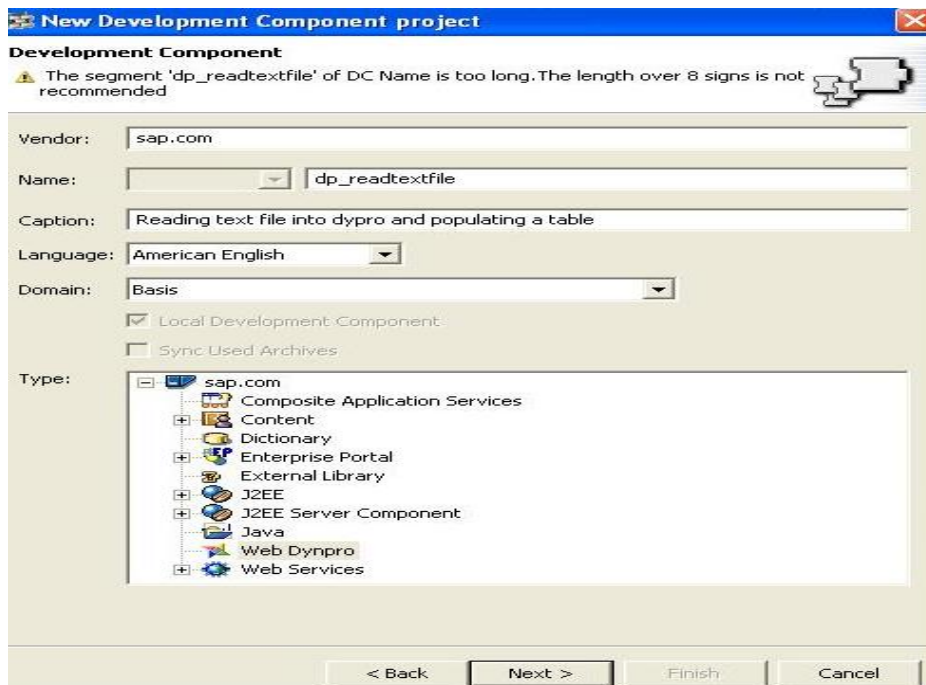
```

A1#$A3#$A4#$
A2#$A5#$A6#$
A7#$null#$null#$
A8#$A9#$A0#$

```

## Steps to Create the Application

1. Create a Web Dynpro application.



Right Click on Dc Create an application .

**New Application**

**Application properties**  
Enter the properties for the new application

Name: Appintextile  
Package: com.sap  
 Authentication  
Source Folder: src/packages

< Back   **Next >**   Finish   Cancel

Click on next .Finally on Finish.

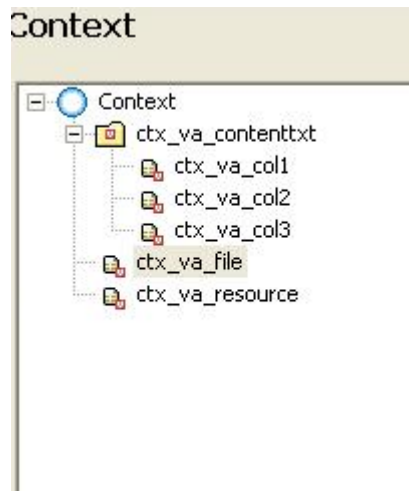
**New Web Dynpro Component**

Enter name and package for the new Web Dynpro component and default window.  
Please note that package entries will be converted to lower case.

Component Name: Appintextile  
Component Package: com.sap  
Source Folder: src/packages  
Window Name: Appintextile  
Window Package: com.sap  
 Embed new View  
View Name: AppintextileView  
View Package: com.sap

< Back   Next >   **Finish**   Cancel

2. Create a context Structure in the ApplintextileView as shown below.



The Properties for the node `ctx_va_contenttxt`

Properties	
Property	Value
[-] Mapping	
mappingRef	Applintextile.ctx_va_contenttxt
mappingType	collection_and_selection
[-] Misc	
cardinality	0..n
collectionType	list
initializeLeadSelection	true
name	ctx_va_contenttxt
selection	0..1
singleton	true
structure	
technicalDocumentation	
typedAccessRequired	true

`Ctx_va_col1`, `Ctx_vacol2`, `ctx_va_col3` and `ctx_va_file` are of the type String.

`Ctx_va_resource` of the type given below.

[-] Mapping	
mappingRef	Applintextile.ctx_va_resource
[-] Misc	
name	ctx_va_resource
readOnly	false
structureElement	
type	com.sap.ide.webdynpro.uelementdefinitions.Resource

3. Add the file upload UI element and using apply template create a Table UI element using the ctx\_va\_contenttxt node

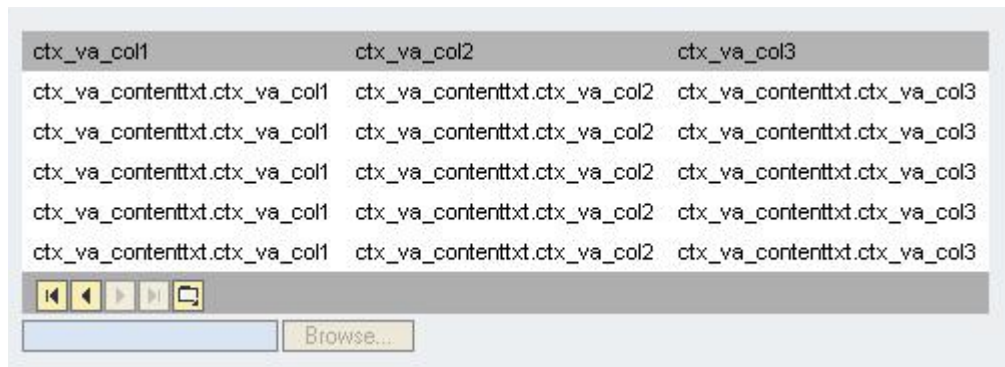
Properties of the table UI element.



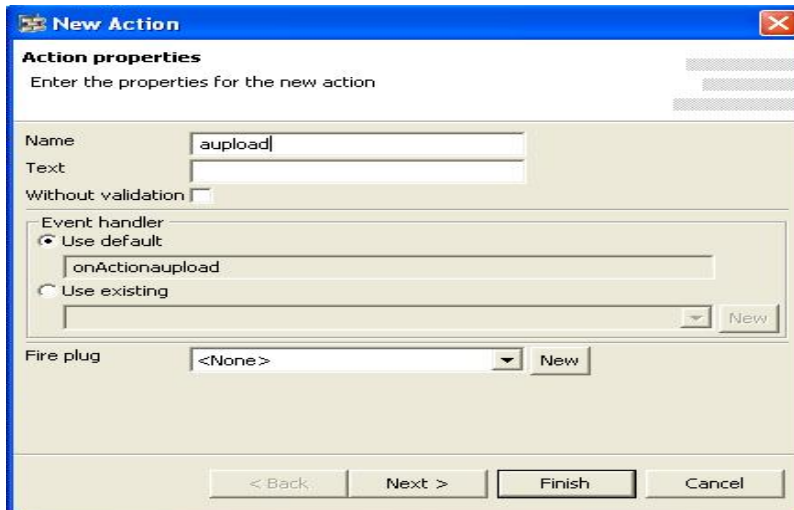
Properties	
Property	Value
Element Properties [Table]	
accessibilityDescription	
compatibilityMode	auto
dataSource	ctx_va_contenttxt
design	standard
emptyTableText	

Properties of the File Upload UI element.

Properties	
Property	Value
Element Properties [UIElement]	
data	ctx_va_file
enabled	true
fileName	
id	FileUpload
resource	ctx_va_resource
textDirection	inherit
tooltip	<>
visible	visible
width	



4. Add a Button UI element and create an action upload.



5. Go to the implementation tab of the View and copy the following code in the method generated on creating the action upload.

```
public void onActionaupload(
    com.sap.tc.Web_Dynpro.progmodel.api.IWDCustomEvent wdEvent) {
    //@begin onActionaupload(ServerEvent)
    InputStream text = null;
    int temp = 0;
    try {

        File file =
            new File(
                wdContext
                    .currentContextElement()
                    .getCtx_va_resource()
                    .getResourceName()
                    .toString());
        FileOutputStream op = new FileOutputStream(file);

        if (wdContext.currentContextElement().getCtx_va_resource()
            != null) {

            text =
                wdContext
                    .currentContextElement()
                    .getCtx_va_resource()
                    .read(
                        false);
            while ((temp = text.read()) != -1) {
                op.write(temp);
            }
            op.flush();
            op.close();
            String filePath = file.getAbsolutePath();

            BufferedReader input =
                new BufferedReader(new FileReader(filePath));
            String line = null;

            while ((line = input.readLine()) != null) {
                StringTokenizer st = new StringTokenizer(line,
                    "#$", false);
```

```

        while (st.hasMoreTokens()) {
IPrivateAppIntextileView.ICtx_va_contenttxtElement ele =
wdContext.createCtx_va_contenttxtElement();
        String col1 = st.nextToken();
        String col2 = st.nextToken();
        String col3 = st.nextToken();

        if (!col1.equalsIgnoreCase("null")) {
            ele.setCtx_va_col1(col1);
        } else {
            ele.setCtx_va_col1(null);
        }
        if (!col2.equalsIgnoreCase("null")) {
            ele.setCtx_va_col2(col2);
        } else {
            ele.setCtx_va_col2(null);
        }
        if (!col3.equalsIgnoreCase("null")) {
            ele.setCtx_va_col3(col3);
        } else {
            ele.setCtx_va_col3(null);
        }

wdContext.nodeCtx_va_contenttxt().addElement(ele);
        }

    }
} catch (Exception e) {
    e.getMessage();
}
//@@end
}

```

**Note:** The delimiter in above case is "#\$".

6. Save the application right click on the dc build the dc and deploy.
7. Run the application.



## Application Results

1. Click on browse and upload the correct flat file i.e. the file in which the data is to be uploaded.
2. After selection of the file click on upload button. The data from the text file is uploaded into the table.

	ctx_va_col1	ctx_va_col2	ctx_va_col3
	A1	A3	A4
	A2	A5	A6
	A7		
	A8	A9	A0

Row 1 of 4

## Related Content

[Office Control in Web Dynpro II](#)

[Uploading Excel File With Example in Web Dynpro Java Application](#)

For more information, visit the [User Interface Technology homepage](#).

## Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.