DRTE

Developer Documentation: Generate PDF and Upload to Files (Object Agnostic)

DRTE

1. Functional Summary

Unlike the "Directly on Server" action which targets a temporary Preview record, this method generatePdfAndUploadToFilesDocument (Generate PDF and Upload to Files, return Content Version Id) is designed to be **object-agnostic**. It can target any field (typically a Rich Text Area) on any Salesforce object.

It retrieves the content from the specified field, renders it as a PDF with optional branding (headers/footers), and creates a ContentVersion record linked directly to the source record.

2. Input Parameters

Parameter	API Name	Type	Required	Description
Record ID	recordId	Text	**Yes**	The ID of the record where the content lives and where the file will be attached.
Object API Name	objectApiName	Text	**Yes**	The API name of the target object (e.g., Account, Contract).
Source Field API Name	fileApiName	Text	**Yes**	The API name of the field containing the HTML/Rich Text content (e.g., Rich_Content__c).
File Name	fileName	Text	No	The title of the generated file (omit ".pdf"). Defaults to the Token value.
Envelope Token	token	Text	No	A unique identifier used for the filename fallback and PDF metadata.
Footer Text	footerText	Text	No	Custom text to render in the footer of the PDF.

Header Image URL	headerImageURL	Text	No	A public URL for a branding logo to appear in the PDF header.

3. Flow Implementation Guide

Scenario: Converting a "Final Comments" field to a PDF Archive

1. **Trigger:** Create a Record-Triggered Flow that runs when a Case is marked "Closed."
2. **Add Action:** Search for **"Generate PDF and Upload to Files, return Content Version Id"**.
3. **Map Variables:**
   - Record ID: {!$Record.Id}
   - Object API Name: Case
   - Source Field API Name: Resolution_Summary__c (a custom Rich Text field).
   - File Name: Closure_Summary_{!$Record.CaseNumber}
4. **Result:** Upon Case closure, a PDF of the resolution summary is automatically generated and attached to the Case.

---

4. Developer Constraints & Best Practices

- **Transaction Limits:** Because getContentAsPDF() is treated as a callout, you cannot perform any DML (like updating the record) *before* calling this action in the same transaction.
  - *Best Practice:* Place this action at the very beginning of your Flow logic or use a "Pause" element to separate transactions.
- **Security Error Handling:** If the objectApiName is invalid or the user lacks permissions, the method throws an AuraHandledException. In a Flow context, ensure you use **Fault Paths** to handle these exceptions gracefully.