



Adobe Sign

Creating Forms with Text Tags Reference Guide

Table of Contents

Overview	4
Text Tag Version	4
Text Tags Quick Start Guide	5
Text Tag Examples	5
Signature and Initials Fields.....	5
Recipient Role Fields	5
Prefill Fields	6
Text Fields.....	6
Read-only Fields	6
Checkboxes.....	6
Radio Buttons	6
Field Validation.....	6
Conditional Rules.....	6
Calculated Fields.....	7
Conversion of Text Tags to Form Fields	7
Field Length and Formatting	7
Interacting with Form Fields During Signing	7
Form Fields in the Signed document.....	8
Long Text Tags	8
Text Tags Deep Dive	9
Text Tag Syntax	9
Controlling Field Interactions During Signing	10
Assigning Form Field to Specific Signers	10
Sender Prefills Data	10
Unassigned Fields	11
Basic Adobe Sign Field Types	11
Signature and Initials Fields.....	11
Special Adobe Sign Fields	11
Special Read-only Fields	12
Signature Block.....	12
Text Fields.....	12
Defining Required / Optional Fields	13
Editable / Read-Only Fields	13
Advanced Document Processing with Text Tags	14
Additional Field Types	14
Checkboxes.....	14
Radio Buttons	15
Multi-line Text Fields	16
Drop-down Lists	16

Supporting Documents (File Attachments)	17
Hyperlinks	18
Participant Stamps and Transaction IDs.....	18
Digital Signatures.....	19
Specifying Default Value.....	19
Specifying Tooltip.....	19
Specifying Font Appearance	19
Specifying Field Alignment	20
Specifying Repeat Fields	20
Specifying Data Validation Rules	21
Supported Field Validation Rules	21
Masking Sensitive Data	23
Specifying Conditional Rules for Fields	23
Specifying Calculation Formulas For Fields.....	25
Shortening Text Tags	26
Removing Pages From Output.....	27
Advanced Text Tags Examples	28
Checkboxes.....	28
Radio Buttons	28
Drop-down List	28
Field Validation.....	28
Shortening Text Tags	29
Conditional Rules.....	29
Calculated Fields.....	30

Overview

Adobe Sign text tags are specially formatted text that can be placed anywhere within the content of your document specifying the location, size, type of fields such as signature and initial fields, check boxes, radio buttons and form fields; and advanced optional field processing rules. Text tags can also be used when creating PDFs with form fields. See the [Creating Forms within Acrobat Reference Guide](#).

Text tags are converted into Adobe Sign form fields when the document is sent for signature or uploaded. Adobe Sign text tags can be placed in any document type such as PDF, Microsoft Word, PowerPoint, Excel and text files (.txt, .rtf).

For example:

<code>{{Sig_es_ :signer1:signature}}</code>	A signature field assigned to the recipient identified as signer1.
<code>{{Cmpy_es_ :signer2:company}}</code>	A field for the company name of the recipient identified as signer2.
<code>{{address_es_ :signer1}}</code>	A field for the recipient identified as signer1 to enter his or her address.
<code>{{*city_es_ :signer1}}</code>	A mandatory field for the recipient identified as signer1 to enter his or her city.
<code>{{!price}}</code>	A read-only field. Recipients cannot change the value of this field.

Adobe Sign text tags offer a powerful mechanism for setting up document templates that allow positioning signature and initial fields, collecting data from multiple parties within an agreement, defining validation rules for the collected data, and adding qualifying conditions. Once a document is correctly set up with text tags it can be used as a template when sending documents for signatures ensuring that the data collected for agreements is consistent and valid throughout the organization.

Note: Where applicable, features and functions specific to the [Adobe Sign – Business](#), [Adobe Sign – Enterprise](#), or both plans are noted.

Text Tag Version

This document describes the current, 2.0 syntax of Adobe Sign text tags. This syntax should be used for all new documents and revisions to existing documents. As of July 1, 2016, support will be discontinued for documents created using the older version of the text tag syntax.

Text Tags Quick Start Guide

Text Tag Examples

All the examples below assume tags are used within document content. When using tags as PDF form field names, omit the opening '{{' and the closing '}}'.

Signature and Initials Fields

<code>{{Sig_es_:signer1:signature}}</code>	A signature field assigned to the recipient identified as signer1.
<code>{{Int_es_:signer1:initials}}</code>	An initials field assigned to the recipient identified as signer1.
<code>{{SigB_es_:signer1:signatureblock}}</code>	A signature block assigned to the recipient identified as signer1.
<code>{{OSig_es_:signer1:optsignature}}</code>	An optional signature field assigned to the recipient identified as signer1.
<code>{{OInt_es_:signer1:optinitials}}</code>	An optional initials field assigned to the recipient identified as signer1.

Recipient Role Fields

The signer and sender fields are used to specify the recipient roles for an agreement.

Signer Field

The "signer" role specifies the first recipient (signer or approver) who is not the "sender". The "signer1" role identifies the first recipient in all cases. The recipient identified as "signer1" could be the sender if the sender is signing or approving the agreement first.

<code>{{Cmpy_es_:signer1:company}}</code>	A field for the recipient identified as signer1 to enter his or her company name.
<code>{{N_es_:signer2:fullname}}</code>	A read-only field for the recipient identified as signer2 to enter his or her full name.
<code>{{Dte_es_:signer1:date}}</code>	A read-only field containing the date of signature.
<code>{{Em_es_:signer1:email}}</code>	A read-only field containing the signer's email address.

Note: Instead of using "sender" and "signer" roles, another approach is to only use roles such as "signer1", "signer2", "signer3" and so forth. Please ensure that these participants are added in the correct order when sending the document.

Adobe Sign

Sender Field

The "sender" role can only be used when the "I need to sign First/Last" option is specified during the Send process. The "sender" role always specifies a field assigned to the "sender" regardless of whether he or she signs first or last.

<code>{{*Ttl_es_:sender:title}}</code>	A mandatory field for the sender to enter his or her title.
--	---

Note: Instead of using "sender" and "signer" roles, another approach is to only use roles such as "signer1", "signer2", "signer3" and so forth. Please ensure that these recipients are added in the correct order when sending the document.

Prefill Fields

<code>{{price_es_:prefill}}</code>	A field the sender fills in before sending the document.
------------------------------------	--

Text Fields

<code>{{address_es_:signer1}}</code>	An optional field for the recipient identified as signer1 to enter his or her address.
<code>{{*city_es_:signer1}}</code>	A mandatory field for the recipient identified as signer1 to enter his or her city.

Read-only Fields

<code>{{!price}}</code>	A read-only field. Signers cannot change the value of this
-------------------------	--

Checkboxes

<code>{{[]}}</code>	A checkbox.
<code>{{[x]}</code>	A checkbox, checked by default.

Radio Buttons

<code>{{(Red)Color}}</code>	A radio button that has the value of Red when selected.
-----------------------------	---

Field Validation

<code>{{*Mobile_es_:signer1:phone}}</code>	A mandatory phone field allowing US phone number
<code>{{d_es_:signer1:num(>0,<60)}}</code>	An optional number field allowing values between 0 and

Conditional Rules

<code>{{field_es_:showif(price>60)}}</code>	A field that shows if the "price" has a value greater than 60.
--	--

Calculated Fields

<code>{{TV_es_:calc([SubTotal]+[Tax])}}</code>	A field whose value is automatically calculated as the sum of the "SubTotal" and "Tax" fields.
--	--

Conversion of Text Tags to Form Fields

Field Length and Formatting

When using text tags in an editable document, Adobe Sign overlays a rectangle covering the entire span of the text tag (from the beginning "{" to the ending "}") and converts the text tag into its corresponding Adobe Sign form field.

The length of the form field can be increased by adding white space characters at the beginning of the text tag immediately after the "{" or at the end of the text tag just before the "}". Decreasing the size of the form field can be achieved by reducing the font size of the text tag to get the appropriate length.

The form field formatting (font size, type, color, etc.) is determined by the format of the first '{'. To ensure correct processing of text tags by Adobe Sign, tag definitions should be specified in commonly occurring fonts within the document (Helvetica, Times, Arial, Verdana or Courier). Text tag definitions are case sensitive and must be specified in lowercase text.

The figure below shows text tags placed within a Microsoft Word Document.

This Agreement is made effective as of {{(.....signerDate_es_:signer1:date)}}, between My company ('Company'){{(init_es_:signer1:initials)}} and {{(contractor_es_:signer1:company.....)}}('Contractor')

The entire span of the text tag, from the beginning curly brace to the closing curly brace, **must** fit within a single line. If a text tag spans multiple lines, either in the source document or in the processed PDF, the form field will not be correctly recognized by Adobe Sign and may cause the underlying text to be visible to the signer when signing the document.

When used within PDF Forms, the curly braces should not be included in the name of the PDF form fields and the size of the field is determined by the size of the PDF form field and formatting of the data entered into the field is determined by the PDF form field properties.

This Agreement is made effective as of [sigDate_es_:signer1:date], between My company ('Company') [init_es_:signer1:initials] and [contractor_es_:signer1:company] ('Contractor')

A single document cannot include both PDF form fields and fields defined using text tags.

Interacting with Form Fields During Signing

The above text tags are converted to the following fillable fields during signing experience:

This Agreement is made effective as of Apr 20, 2016, between My company ('Company') * and *Enter your company name ('Contractor')

Form Fields in the Signed document

After data is merged into the signed document, the following would appear in the signed PDF:

This Agreement is made effective as of Apr 21, 2016 and Acme Marketing, Inc.	, between My company ('Company') (('Contractor'))	
---	--	---

Long Text Tags

When used in editable documents, text tags containing many directives may become very long, making it difficult to specify small fields on the document. Adobe Sign provides a simple mechanism allowing you to specify shorter tags in the document.

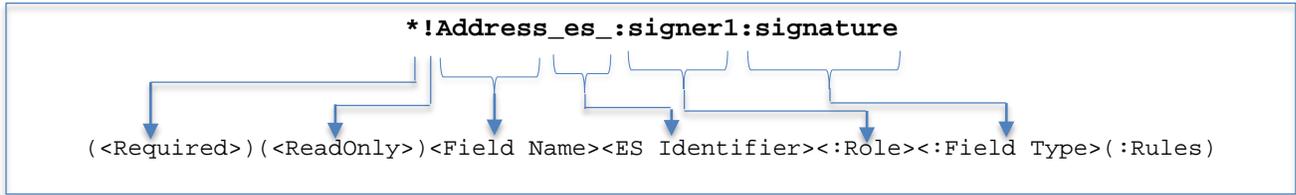
For example, if you need to have a mandatory phone field assigned to the sender, instead of putting the full `{{*Mobile_es_:signer1:phone}}` tag, you may use a shorter tag `{{ $mob }}` in the appropriate location in the document with whitespace to adjust the size of the field. The complete definition of the text tag `{{#mob=*Mobile_es_:signer1:phone}}` can be placed at another location in the document.

To learn more about text tag shortening, see [Shortening Text Tags](#).

Text Tags Deep Dive

Text Tag Syntax

An Adobe Sign text tag is defined as:



Where the components are defined as follows:

- **Required**—An optional flag that specifies whether the particular field is required and must be filled by the signer during signing. If the flag is not present, the form field is not mandatory and the signer can (but is not required to) fill in a value during signing.
- **ReadOnly**—An optional flag that indicates that the particular field is a read-only field. Marking a field as read-only ensures that the data in the field cannot be modified by anyone during signing. Data can be merged into read-only fields using the Adobe Sign APIs or MegaSign via CSV before the document is sent for signature.
- **Field Name**—A unique name to identify this specific field within the document. The field name is used to identify the values that are provided during signing when the data gets exported from the service.

Note 1: If a document has more than one text tag with the same name but different directives, or those directives are in a different order amongst the same-name text tags, Adobe Sign automatically adds "efield" + unique number + "_" to the beginning of the name. For example if two fields called "Address" appear in the document, Adobe Sign changes the name of one of the fields to "efield2_Address".

Note 2: If there are two text tags with the same field name and the same directives and order of directives within a document, these fields are treated as cloned fields. Data entered in one field is automatically cloned to the other field location within the document.

- **ES Identifier**—A specific text string "_es_" which **must** be included after the field name in the text tag in order to enable Adobe Sign specific features. If the Adobe Sign identifier is not included after the field name, a form field is added to the document without any directives.
- **Directive**—A list of one or more definitions beginning with a colon (":") that specify the role, the field type, and additional field processing rules defining the type of input allowed in the field. Directives can be defined in any order.
 - **Role**—The participant this field is associated with (e.g., sender, signer).
 - **Field Type**—The type of Adobe Sign form field (e.g., signature, initials, check box or a text field).
 - **Rules**—Advanced processing rules such as data validation, logic to show/hide fields based on conditions, etc.

Controlling Field Interactions During Signing

Form fields in a document can be assigned to a specific signer during the signature workflow. Adobe Sign ensures that a signer can only interact with the fields that have been assigned to him/her during signing. Form fields can be assigned to a specific signer by adding a specific directive in the text tag.

Assigning Form Field to Specific Signers

Adobe Sign supports the following directives for assigning form field to specific signers.

<code>{{Field2_es_:signer1}}</code>	A field in the agreement that is assigned to the first who is not the sender of the agreement.
<code>{{Field3_es_:signer1}}</code> <code>{{Field4_es_:signer2}}</code>	If an agreement requires more than one signer and each signer needs to provide data during signing, fields in the agreement can be assigned to specific signers using <code>signer<x></code> directive where <code><x></code> is order in which the signers sign the agreement. Fields marked with <code>signer1</code> directive are assigned to the first signer, fields marked with <code>signer2</code> directive are assigned to the second signer, and so on. Adobe Sign ensures that a signer can only interact with the fields that are assigned to her. If the sender of the agreement signs first during signing, any fields marked as "signer1", in addition to the field marked as "sender", is presented to the sender.

Sender Prefills Data

In some cases, the sender is not required to sign the agreement but is required to enter data before the agreement is sent out for signature. This is accomplished by adding a "prefill" directive to the relevant form fields within the agreement. Adobe Sign allows the sender to complete those fields before sending the agreement for signature.

<code>{{Field5_es_:prefill}}</code>	A field in the agreement that needs to be filled by the sender before the agreement is sent for signature.
-------------------------------------	--

Note: Prefill is enabled by default for documents uploaded to your Adobe Sign Document Library. Prefill can also be enabled by enabling the "Preview & Add Signature Fields" option when sending documents. To enable "prefill" for all agreements sent from the Send page, regardless of whether the "Preview" option is enabled, contact Adobe Support.

Unassigned Fields

During signing, any signer can fill a field that does not have a directive assigning it to a specific recipient. After the field has been filled, it is locked to prevent subsequent signers from changing its value.

<code>{{Address}}</code>	A field in an agreement that can be filled by any signer.
<code>{{zip_es_:zip4}}</code>	A field in an agreement for any signer to enter in a zip code with validation.

Basic Adobe Sign Field Types

A variety of field types can be created in a document using specific directives within text tags. Adobe Sign interprets the tags and converts them to the appropriate form fields during signing. All the field types that can be added to a document through the Adobe Sign web application can also be specified through text tags.

Signature and Initials Fields

You may position Adobe Sign signature or initials fields at specific locations within the document by adding a directive to designate the type of the signature field. By default, signature and initials fields are mandatory. However, some documents require that the signer optionally initial or sign at specific additional locations within the document. Optional signature or initials fields can also be defined within a document using text tags.

Note: Adobe Sign requires that a document contain at least one signature within the document for each signer of the document. Approvers of a document are not required to have a signature field on the document during signing.

<code>{{Sig1_es_:sender:signature}}</code>	A required signature field assigned to the sender.
<code>{{Int1_es_:signer1:initials}}</code>	A required initials field assigned to the recipient identified as signer1.
<code>{{OSig1_es_:signer2:optsignature}}</code>	An optional signature field assigned to the recipient identified as signer2.
<code>{{OInt1_es_:signer1:optinitials}}</code>	An optional initials field assigned to the recipient identified as signer1.

Special Adobe Sign Fields

Adobe Sign provides special fields that can be positioned within the document to collect information from the signer such as signer's job title or company name. If the information for these fields is already available, Adobe Sign automatically populates these fields when presenting to the signer during signing.

<code>{{Ttl1_es_:title}}</code>	A field to specify the signer's job title.
<code>{{Cmpy1_es_:company}}</code>	A field to specify the signer's company.

Special Read-only Fields

In addition, Adobe Sign also supports collecting the date of signature, the email address, and the name of the signer within the agreement. These fields are automatically populated by Adobe Sign and cannot be modified by the signer when signing the agreement.

<code>{{Dte1_es_:date}}</code>	A field to automatically capture the date of signature.
<code>{{Em1_es_:email}}</code>	A field to automatically capture the signer's email address. When used in Adobe Sign Widgets, the email address field is editable by the signer when he/she signs the Widget. Depending on your account configuration, Adobe Sign verifies the email address provided by the signer before completing the Widget signing process.
<code>{{Name1_es_:fullname}}</code>	A field to specify the signer's full name.

Signature Block

If no signature field is placed in the document for a specific signer, Adobe Sign automatically places a signature block at the bottom of the document for that particular signer. By default, a signature block includes signature and email address fields. However, if your Account or Group Admin has enabled either the 'Require signers to provide their job title when e-signing' or the 'Require signers to provide their company name when e-signing' setting or both (in Signature Preferences), then title and/or company fields will be included in the signature block. The graphic below shows a signature block that includes all four fields.



The image shows a signature block form with a light pink background and a blue border. It contains four rows of input fields, each with a label on the left: 'Signature:', 'Email:', 'Title:', and 'Company:'. The 'Signature:' field contains the text 'Signer Name (Date)' in a smaller, lighter font. The other three fields are empty.

You may position the entire Adobe Sign signature block in your document using the "signatureblock" directive.

<code>{{SBik1_es_:signatureblock}}</code>	Positions a signature block within the document at the specified location.
---	--

Text Fields

All fields defined within a document that do not contain any additional field type definition directives are treated as fillable text fields within a document.

<code>{{propertyName_es_:signer}}</code>	Defines an editable text field called "propertyName" within a document.
--	---

Defining Required / Optional Fields

For most fields, you may specify if they are required or optional. Adobe Sign ensures that a signer cannot sign an agreement before all mandatory fields are filled.

By default, all fields are optional with the following exceptions:

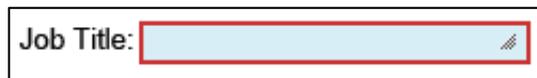
- Signature and Initials fields are mandatory by default unless you define these fields using the "optsignature" or "optinitials" text tags. See [Signature and Initials Fields](#) for more information on these text tags.
- Title and Company fields are mandatory if your account settings require them to be part of the signature block. See [Signature Block](#) for information on the specific account settings.

A mandatory field is marked with an asterisk (*) prefix to its name or by adding the required directive in the text tag.

For example:

<code>{{*phone_es_:signer1}}</code> <code>{{phone_es_:signer1:required}}</code> <code>{{phone_es_:prefill:required}}</code>	Defines a mandatory text field called "phone" to be filled by the signer or during the prefill process.
<code>{{address_es_:signer1}}</code>	Defines an optional field called "address" to be filled by the recipient identified as signer1.

Required fields are outlined in red when signing.



Editable / Read-Only Fields

By default, all fields that are not read-only are editable. Exceptions include some special signer fields (email, signer name, and signature date) that are read-only by default. See [Special Read-only Fields](#) for more information.

Read-only fields are useful when automatically merging data into a document using the Adobe Sign API or the MegaSign data merge function. Read-only fields ensure the signer cannot change the merged data when signing.

Adding an exclamation mark (!) prefix to the field name or using the read-only directive marks it as a read-only field.

For example:

<code>{{!price}}</code> <code>{{price_es_:readonly}}</code>	Defines a read-only text field to be filled in by merging data before the document is sent out for signature.
--	---

Advanced Document Processing with Text Tags

Additional Field Types

You can create the following types of fields using their unique text tags:

- [Checkboxes](#)
- [Radio buttons](#)
- [Multi-line text fields](#)
- [Drop-down lists](#)
- [Supporting documents \(attachments\)](#)
- [Hyperlinks](#)
- [Participant stamps and transaction IDs](#)

Checkboxes

You can create a checkbox using text tags by placing the Checkbox tag in your document or by using the checkbox directive.

<code>{{[]}}{{CB_es_:checkbox}}</code>	Positions an unnamed checkbox within a document.
<code>{{[x]}}{{CB1_es_:checkbox(checked)}}</code>	Positions an unnamed checkbox that is checked by default within a document.
<code>{{[[]Comm_es_:signer1]}}{{Comm_es_:checkbox:signer1}}</code>	Positions a checkbox called "Comm" assigned to the recipient identified as signer1.
<code>{{[*]}}{{CB2_es_:required:checkbox}}</code>	An unnamed mandatory checkbox that must be filled in by the recipient identified as signer1. See Defining Required / Optional Fields for more information about defining mandatory fields.
<code>{{[!]}{{CB3_es_:readonly:checkbox}}</code>	An unnamed read-only checkbox. See Defining Required / Optional Fields for more information about defining mandatory fields.
<code>{{CB4_es_:checkbox:label("I agree.")}}</code>	Defines a checkbox with a label for a user-friendly description of the checkbox. The label is displayed to the signer during signing and in this case displays: "I agree."

Radio Buttons

Text tags can be used to define and position **Radio Buttons** within a document. Radio buttons must have at least two options (choices) within a radio button group, a group identified by the same field name within the document. Each option (choice) of the radio button must have a different selection value. Radio buttons can be defined by using the following syntax.

<code>{{(option_value)fieldname_es_}}</code>	Positions a radio button within a document.
<code>{{(Red)Color_es_:signer1}}</code> <code>{{(Blue)Color_es_:signer1}}</code> <code>{{(Green)Color_es_:signer1}}</code>	Defines three radio buttons belonging to the same group called "Color" assigned to the recipient identified as signer1 . The options have values of Red, Blue, and Green.
<code>{{Color_es_:signer1:radio(Red)}}</code> <code>{{Color_es_:signer1:radio(Blue)}}</code> <code>{{Color_es_:signer1:radio(Green)}}</code>	An alternative way to define the Color radio button group using the radio directive.
<code>{{(Red)Color_es_:signer1:label("Red")}}</code> <code>{{(Blue)Color_es_:signer1:label("Blue")}}</code> <code>{{(Green)Color_es_:signer1:label("Green")}}</code>	Defines a radio button group and gives each radio button a user-friendly label. The label is displayed to the signer during signing.

Specifying Radio Button Style

Adobe Sign supports creating radio buttons in different styles. The specified radio button style is displayed in the PDF. However, during signing the radio buttons options are always presented as circles.

Text Tag Directive	Permitted values		Example
:style	circle – Default style if none is specified.		:style(circle) :style(diamond) :style(star) :style(check) :style(cross) :style(square)
	diamond		
	star		
	check		
	cross		
	square		

Multi-line Text Fields

Text tags can be used to define text fields that can span multiple lines. These fields can be used within documents to collect multiple lines of data during signing.

<code>{{largeField_es_:signer1:multiline}}</code>	Defines a field, assigned to the recipient identified as signer1, called "largeField" that allows the signer to enter multiple lines of data in the field during signing.
<code>{{largeField2_es_:signer1:multiline(4)}}</code>	Defines a field, assigned to the recipient identified as signer1, called "largeField2" that allows the signer to enter four lines of data during signing. If the number of lines parameter is missing, the field support two lines of data.

Drop-down Lists

Text tags can be used to define and position drop-down lists within a document. Drop-down lists can contain multiple options; the user can choose one of these options when signing the document. A drop-down list definition has two components: a unique set of options from which the recipient can select one choice and optionally a set of values that are returned when exporting the data from the agreement.

Drop-down options must be unique although drop-down values do not need to be unique (i.e., multiple drop-down options can have the same value exported from them). Drop-down values are optional. If no drop-down values are defined, the drop-down options are returned when the data gets exported from the agreement. When drop-down values are defined, the number of values defined must match the number of options. If the number of options and values specified do not match, this results in an error and the field is not treated as a drop-down list.

When defining a default selection for a drop-down list or defining conditions for another field based on the recipient's selection of one of the options from the drop-down list, the export values (not the options) must be used. If there are no export values defined, then use the drop-down options.

<code>{{*Color_es_:signer1:dropdown(options="Red,Green,Blue")}}</code>	Positions a required drop-down list within a document. During signing, the list allows the recipient identified as signer1 to select from one of the three available choices. This drop-down list does not specify any export values, as a result the option selected is exported from the Manage page, through the API or from the Reports page.
--	---

<pre> {{Color_es_:signer1:dropdown(options="Red,Green,Blue", values="R,G,B")}} </pre>	<p>Positions a drop-down list within a document. During signing, the list allows the recipient identified as signer1 to select from one of the three available options. In this case, the drop-down list also specifies possible export values (R,G,B). Depending on the option selected by the recipient during signing, the corresponding value gets exported from the Manage page, through the API or from the Reports page. For example if the recipient selects "Red" the value exported is "R".</p>
<pre> {{Color_es_:signer1:dropdown(options="Red,Green,Blue", values="R,,B")}} </pre>	<p>In this example, the drop-down list options are the same as in the previous example but one of the values is empty. If the recipient identified as signer1 selects "Green" from the drop-down during signing, no value gets exported for this field.</p>
<pre> {{Color_es_:signer1:dropdown(options="Red,,Blue", values="R,G,B")}} </pre>	<p>In this example, the drop-down presents three options to the recipient identified as signer1 during signing, with the second option being blank. If the recipient selects the blank option, the value exported is "G".</p>

Additional directives can be added to the drop-down fields definition as needed, for example, by making a drop-down list required, applying conditional logic to drop-down lists, or specifying that the option selected by the recipient should be masked in the PDF.

Supporting Documents (File Attachments)

Adobe Sign can be used to collect supporting documents from signers during signing. During the signing experience, Adobe Sign converts the text tag into a field that the signer can select to upload a file. The uploaded document(s) are included as part of the signed document and are converted to PDF and attached at the end of the signed agreement after the signing process. Form fields for collecting files from the signer are defined by using the "attachment" directive.

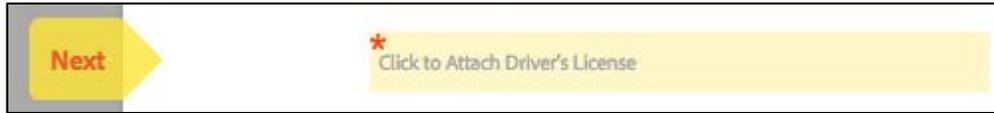
<pre> {{*DriversLicense_es_:signer1:attachment}} </pre>	<p>Defines a mandatory field called "DriversLicense" assigned to the recipient identified as signer1. The recipient is required to use this field to upload a file during signing.</p>
<pre> {{photo_es_:signer1:attachment:label("Photo")}} </pre>	<p>Defines an optional field called "photo" assigned to the recipient identified as signer1. The recipient can use this field to upload an image during signing. The directive label can be used to specify a user-friendly description for the field. The label is displayed to the signer during signing.</p>

Adobe Sign

Defining a file attachment field:

```
{{*DriversLicense_es_:_signer1:attachment:label("Driver's License")}}
```

File Attachment field presented during signing:



Hyperlinks

Adobe Sign can be used to specify hyperlinks in documents. **Hyperlinks** are clickable during signing and can navigate the recipient to a location within the document or to an external URL. Labels can also be specified for hyperlinks so that the document displays the text label instead of the direct URL.

Note: Hyperlinks are only supported for *Adobe Sign – Business* and *Adobe Sign – Enterprise* plans.

<pre>{{companywebsite_es_:_link(http://www.adobe.com):label(Adobe)}}</pre>	Defines a hyperlink that links to an external URL. Includes a label "Adobe" that is displayed to the recipient during signing.
<pre>{{doclink_es_:_link:page(10):label(View this section)}}</pre>	Defines a hyperlink that links to page 10 in the current document. Includes a label "View this section" that is displayed to the recipient during signing.

Participant Stamps and Transaction IDs

Participant Stamps and **Transaction IDs** can be added to the signed PDFs for signers. The specific participant's stamp appears after that person has signed or approved the document. The participant stamp includes the participant's name, signing/approval event time, and email. The transaction IDs and stamps appear in the final signed PDF.

Note: Participant Stamps and Transaction IDs are only supported for *Adobe Sign – Business* and *Adobe Sign – Enterprise* plans.

<pre>{{userstamp1_es_:_signer1:stamp}}</pre>	Defines a participant stamp for the recipient identified as signer1. This stamp appears in the final signed PDF.
<pre>{{userstamp2_es_:_signer2:stamp}}</pre>	Defines a participant stamp for the recipient identified as signer2. This stamp appears in the final signed PDF.
<pre>{{transstamp2_es_:_transactionid}}</pre>	Defines a transaction ID stamp that appears in the final signed PDF.
<pre>{{userstamp_es_:_signer1:stamp:repeat}} {{transstamp1_es_:_transactionid:repeat}}</pre>	Defines a stamp that appears in the same location on all pages in the final signed PDF.

Digital Signatures

Digital Signatures can be added to documents. There can only be one digital signature per signer in each document. If you add more than one Digital Signature per signer (e.g., `{{digsig1_:_signer1: digitalsignature}}` and `{{digsig2_:_signer1: digitalsignature}}`), only the first is preserved and the others are automatically removed when the document is sent for signature, unless manually removed in the preview and authoring environment before being sent.

Note: Digital Signatures are only supported for the *Adobe Sign – Enterprise* plan.

<code>{{digsig1_es_:_signer1:digitalsignature}}</code>	Defines a digital signature for the recipient identified as signer1.
--	--

Specifying Default Value

Adobe Sign supports specifying default value for a field using text tags. The specified default value is shown to the participants during signing.

Text Tag Directive	Example
:default	:default("Default value specified in double quotes") :default('Default value specified in single quotes')

If the default value contains a single quote or apostrophe, the default value must be enclosed within double quotes. For example: `{{location_es_:_signer1:default("Signer's Business")}}`. If double quotes are required within the default value, the default value must be placed within single quotes.

Specifying Tooltip

Text tags also allow specifying **Tooltips** for Adobe Sign form fields. Tooltips are displayed to the recipient during signing.

Text Tag Directive	Example
:tooltip	:tooltip("Please provide a copy of your Driver's License") :tooltip('Choose a color')

Specifying Font Appearance

Text tags can be used to change the font appearance of the field so that it overrides the font appearance used in the text tag for font name, color, and size. To specify font name, use the font name of any font in the existing document or either of the following fonts: Helvetica, Times, Arial, Verdana, or Courier. To specify color, use the RGB hex value or any of the following valid color values: white, lightgray, gray, darkgray, black, red, pink, orange, yellow, green, magenta, cyan, or blue. By default, the font name and size are the font name and size of the first bracket "{" at the beginning of the text tag.

Text Tag	Example
----------	---------

Directive	
:font	:font(name=Times, color=#FF0000, size=12) – Specifies the field appearance to be Times font in red text and size 12 point font :font(color=green, size=14) – Specifies the field appearance to be in Green text and size 14 point font

Specifying Field Alignment

Text tags can be used to align the values entered in the form fields. By default the values in the form fields are left aligned. However using directives on the text tags, the form fields can be aligned to be center aligned or right aligned.

Text Tag Directive	Example
:align(alignment)	:align(left) – Aligns the text in the field to the left. This is the default behavior for form fields and the directive does not need to be satisfied. :align(right) – Aligns the data in the field to the right. :align(center) – Center aligns the data entered in the form field.

Specifying Repeat Fields

Text tags can be used to repeat the same fields on each page in the document without having to copy the text tag on each page. For example, this could be particularly useful for signature fields or participant stamps on lengthy documents.

Text Tag Directive	Example
:repeat	:stamp:repeat – Copies the participant stamp to the same location on each of the other pages in the document. Make sure to define a signer role for the stamp. (e.g., signer1, signer2). :signature:repeat – Copies the signature field to the same location on each of the other pages in the document. Make sure to define a signer role for the stamp. (e.g., signer1, signer2). :repeat(even) – Copies the field to the same location on the even pages in the document. :repeat(odd) – Copies the field to the same location on the odd pages in the document. :repeat(after) – Copies the field to the same location on all the pages after the page where the field is added. :repeat(before) – Copies the field to the same location on all the pages before the page where the field is added. :repeat(2,10), :repeat(6-9), :repeat(2,6-9) – Copies the field to the same location on the specified page or page ranges in the document.

Specifying Data Validation Rules

Adobe Sign supports a variety of field validation types specified by additional text tag directives. Specified data validation rules are applied when the signer is filling in the data during signing. Adobe Sign will not allow the recipient to complete the signing process if the data entered does not match the specified validation rule.

For example:

<code>{{*Mobile_es_:signer1:phone}}</code>	Defines a required field called "Mobile" assigned to the recipient identified as signer1. The recipient must provide a valid phone number in this field during signing.
<code>{{Birthday_es_:signer2:isdate}}</code>	Defines a field called "Birthday" assigned to the recipient identified as signer2. The recipient needs to enter a date value during signing.

Some field validation types may have additional parameters that narrow down the allowed values in the field.

<code>{{duration_es_:signer1:num(>=0,<=60)}}</code>	Defines an optional field called "duration" where the value of the field must be between 0 and 60.
<code>{{tenNum_es_:signer1:string(char=num,maxlen=10)}}</code>	Defines an optional field called "tenNum" which allows entering up to a maximum of 10 digits.

Supported Field Validation Rules

Validation Type	Text Tag Directive	Optional Parameters & Values		Samples
String	:string	char	alpha – Letters only	:string :string(char=alpha) :string(char=num) :string(char=alphanumeric) :string(maxlen=10) :string(char=alpha,maxlen=10)
			num – Numbers only	
			alphanumeric – Letters & numbers	
		maxlen	Specific maximum length	
Number	:num	>=	Greater than or equal	:num :num(<=1000) :num(>=50,<=500)
		<=	Less than or equal	
Currency	:curr	country	us – Default	:curr :curr(country=us) :curr(country=uk) :curr(<=500) :curr(country=us,<=100)
			uk - United Kingdom	
		>=	Greater than or equal	
		<=	Less than or equal	
Percentage	:pct	>=	Greater than or equal	:pct :pct(>=50) :pct(>=0,<=100)
		<=	Less than or equal	

Adobe Sign

Validation Type	Text Tag Directive	Optional Parameters & Values		Samples
Date	:isdate	format	mm/dd/yy (Default)	:isdate
			mm/dd/yyyy	:isdate(format=dd/mm/yy)
			dd/mm/yy	
			dd/mm/yyyy	
			mm/yy	
Time	:time			:time
Email	:isemail			:isemail
Zip	:zip	country	us – Default US Zip Code format	:zip
			uk - United Kingdom Zip Code format	:zip(country=uk)
Zip + 4	:zip4			:zip4
Phone	:phone	country	us – Default US phone # format	:phone
			uk - United Kingdom phone # format	:phone(country=uk)
Social Security Number	:ssn			:ssn
Using regular expression for validating data in a text tag	:custom	regexp	A regular expression expressed in the form of a string. In this syntax, the backslash char that is part of regexp constructs such as "\w", must always be escaped with a backslash., i.e., "\\w". And in order to search for a backslash, one must use \\\\. The string begin and end character double-quotes ("). If a quote is expected to occur within the data that the recipient enters, to provide a regexp validation string to match the quote they must be escaped as \" in the expression. A good reference for the syntax of regular expressions used by Adobe Sign can be found here .	:custom(regexp="^5[1-5][0-9]{14}\$") :custom(regexp="^.{3}\$",msg="Not a match for three chars in a row!")
		msg	Message to be displayed to the recipient when the data does not match the specified expression.	

A single field can only include one directive for validation. If multiple validation rules are specified in a single text tag, only the first validation rule gets applied. All subsequent validation rules are ignored.

Masking Sensitive Data

Adobe Sign supports masking sensitive data collected during signing by adding a specific directive to the text tag. During signing, the data entered by the signer in such a field is converted into asterisks (*) or to a user-defined masking character as soon as the signer exits the field. The signer-entered data is also not presented to any subsequent signers or included in the completed PDF. The signer-entered data can be exported from Adobe Sign or through the Adobe Sign API.

Text tag definition:

```
CC Number: {{{*CC_es_:_signer1:mask}}}
```

During signing:

```
CC Number: ******
```

In the final PDF:

```
CC Number: *****
```

<code>{{*CC_es_:_signer1:mask}}</code>	Defines a required field called "CC" assigned to the recipient identified as signer1, which gets masked during signing.
<code>{{*CC_es_:_signer1:mask(char=-)}}</code>	Defines a required field called "CC" assigned to the recipient identified as signer1, which gets masked during signing with "-". If a character is not specified in the text tag, asterisk (*) is used as the default masking character.

Specifying Conditional Rules for Fields

Adobe Sign supports applying conditional rules for fields in a document. Conditional rules can be used to define when a particular field within a document is shown to or hidden from the signer during signing. Conditions applied to a particular field are based on the values of one or more other fields within the document. Conditional rules applied to a particular field can also specify that a particular field be shown or hidden when all the conditions defined for a particular field are satisfied or when any one of the conditions defined for a particular field is satisfied.

<code>{{_es_:_signer1:showif(condition)}}</code> <code>{{_es_:_signer1:showifany(condition)}}</code>	Show the field to the recipient identified as signer1 if any one of the conditions defined is satisfied.
<code>{{_es_:_signer1:showifall(condition)}}</code>	Show the field to the recipient identified as signer1 only if all the defined conditions are satisfied.
<code>{{_es_:_signer1:hideif(condition)}}</code> <code>{{_es_:_signer1:hideifany(condition)}}</code>	Hide the field from the recipient identified as signer1 if any one of the conditions defined is satisfied.

Adobe Sign

<code>{{_es_:signer1:hideifall(condition)}}</code>	Hide the field from the recipient identified as signer1 only if all the defined conditions are satisfied.
<code>{{_es_:signer1:enableif(condition)}}</code> <code>{{_es_:signer1:enableifany(condition)}}</code>	Enable the field for the recipient identified as signer1 if any one of the conditions defined is satisfied. This is similar to the 'showif' condition except that if the condition is not satisfied, the field appearance during signing is visible but grayed out and not editable. The grayed out field displays the field's default value, if applicable. The value also appears in the signed PDF.
<code>{{_es_:signer1:enableifall(condition)}}</code>	Enable the field for the recipient identified as signer1 only if all the defined conditions are satisfied.
<code>{{_es_:signer1:disableif(condition)}}</code> <code>{{_es_:signer1:disableifany(condition)}}</code>	Disable the field for the recipient identified as signer1 if any one of the conditions defined is satisfied. The disabled field appears as a grayed out field during signing. If there is a default field value, the value of a disabled field still appears in the signed PDF.
<code>{{_es_:signer1:disableifall(condition)}}</code>	Disable the field for the recipient identified as signer1 only if all the defined conditions are satisfied.

A condition is defined in the following way: **fieldname operator value** where the **fieldname** is the name of the form field within the document on which the condition is based. The supported operators and valid values for the various supported form fields are shown in the tables below.

Conditional Field Operators

Operator	Meaning	Value	Example
=, ==, eq	Equals	Single value	myCheck=checked
!=, <>, neq	Not Equals	Single value	signature!=signed
<, lt	Less Than	Single value, numeric	myNumber < 10
>, gt	Greater Than	Single value, numeric	myNumber > 40
<=, lte	Less Than or Equals	Single value, numeric	myNumber <= 15
>=, gte	Greater Than or Equals	Single value, numeric	myNumber >= 0
?=, in	In	Comma delimited list, must be quoted or within parentheses	option ?="1,2,3"
?!=,!in, not in, notin	Not In	Comma delimited list, must be quoted or within parentheses	option ?!="1,2,3"
~=, contains	Contains	Single value	Title ~= Manager
~!=, !contains, not contains, notcontains	Does Not Contain	Single value	Title ~!= Junior

Multiple conditions for a single field can be separated using a comma. For example:

`{{notRed_es_:showifall(checkbox=checked,Color!=Red)}}` defines a field called "notRed" which is shown to the signer when both the conditions are satisfied.

Conditional Field Value for Supported Text Tag Fields

Field Type	Valid Values	Example
Signature	signed, unsigned	signHere=signed
Initials	signed, unsigned	initHere!=signed
Checkbox	checked, unchecked	myCheck=checked
Radio Button	Value of the radio button options	Color=Red
Text Fields	Value of the text field	Title!=Senior Title~=Chief Price>100

Conditions can be assigned to any field, including required fields like initials, signatures, and signature blocks. When a condition is defined on a required field it is only treated as required if the condition makes the field visible to the signer during signing. In other words, if a signature field has a condition on it that hides it, it is not visible to the signer and therefore does not require the signer to provide a signature in this particular field. The "required field" count at the bottom of the signing page gets dynamically updated, based on the defined conditions, to accurately reflect the number of fields that the signer is required to fill before completing signing the document.

Every signer must have one signature field that does not have a condition assigned to it. Because of this, if all signature fields for a given signer have a condition, a new signature block is added for that signer at the end of the document.

Specifying Calculation Formulas For Fields

Adobe Sign supports dynamically populating field values into a document during signing based on other field values/choices made by the customer during signing; these types of fields are known as **Calculated Fields**. Calculation formulas can be defined similarly to other fields on the document using text tags or the Adobe Sign drag-and-drop authoring environment. A calculation formula can reference one or more existing fields in the document and can also be based on another calculated field within the document. Calculations can be applied to numeric fields, date fields and text fields.

In addition to calculating the value of a field in the document, calculated fields can also include formatting information to appropriately style the output of the calculated value in the document. Additional rules such as Conditional Rules, Field Masking, and so forth can also be applied to calculated fields through text tags.

<code>f1_es_ :signer1:calc(formula)</code>	A calculated field assigned to the recipient identified as signer1. Its value for is automatically calculated based the defined formula.
<code>f2_es_ :signer1:calc(formula):format(formatType, format)</code>	A calculated field assigned to the recipient identified as signer1. Its value is calculated based on the defined formula and the resulting output value is of a specific type and is formatted per the format specified.

Example: A calculated field can be used to dynamically calculate the total tax owed on an order.

```
{{Tax_es_ :signer1:calc([subtotal] * [taxrate]):format(number, "$,.00")}}
```

The tax calculation is the based on the total cost of the items ordered by the customer and the tax rate for the state of residency of the signer. The resulting calculated tax value is formatted with a dollar (\$) symbol before the value and includes two decimal points in the resulting value. For example, if the subtotal is 1250 and tax rate is 6.25%, the resulting calculated value displayed in the Tax field in the document would be \$1,328.13.

For more information about calculated fields and using them in your documents, please refer to the [Calculated Fields Reference Guide](#).

Shortening Text Tags

Text tags provide a very powerful mechanism for defining different types of fields within a document, assigning those fields to specific signers, specifying validation rules on the fields and/or applying conditional rules to those fields. Applying all of this to a single form field can make the field definition very long and can cause the document structure to be changed as a result of the field definition. Furthermore, since each field definition must fit on one line, the use of multiple complex directives can make document layout very difficult. To ensure consistency of the structure of the document, Adobe Sign supports a mechanism for defining shortened Text Tags within the document. A shortened text tag consists of two parts:

- **Reference Tag**—A shortened tag definition placed at the appropriate location within the document. The reference tag controls the size and location of the actual form field in the document. A reference tag has the following syntax: `{{$shortened_name}}`. For example, `{{r}}` defines a reference tag where "r" is the short name of the tag. The size of the field can be adjusted using whitespaces either immediately after the last open curly brace or immediately preceding the first closed curly brace.
- **Definition Tag**—This refers to the complete definition for a specific tag and can be defined anywhere in the document. The size and location of the definition tag does not determine the size and location of the actual form field within the document. A definition tag has the following syntax: `{{#shortened_name=Standard Tag Syntax}}`. For example, `{{#r=[]ReceiveCommunication_es_ :signer1}}` defines a complete tag definition for the reference tag named "r". In this example, the tag defines a check box assigned to the recipient identified as signer1.

For example, the figure below show two reference tags placed within a document.

```
>{{$a}} I would like to receive e-mail communications.  
E-mail: {{{$e}}}
```

The complete definition of the two reference tags is defined elsewhere in the document as shown in the figure below.

```
{{#a=(email) TypeOfCommunication_es_signer1:showif(ReceiveCommunication="checked")}}  
{{#e=*emailAddress_es_signer1:isemail:showif(TypeOfCommunication="email")}}
```

The complete definition for the text tags is very long and without the ability to define shortened tags in the appropriate location within the document it would not have been possible to maintain the structure of the original document.

Removing Pages From Output

By separating form field locations from the definition of the field, it is possible that all the definitions for tags within a document are placed at the end of the document or on a completely separate page within the document. When the document is sent through Adobe Sign, all the text tags are processed and a white box is placed on top of the text tag definition so the text is no longer visible to the signer. This may result in entirely blank pages within the document after being processed by Adobe Sign, pages that contain only text tags. To avoid additional unwanted blank pages on the document Adobe Sign supports a special text tag definition to remove a particular page from the processed output document generated by Adobe Sign before sending the document for signature.

{{#REMOVE_PAGE_FROM_OUTPUT}}	Special directive that instructs Adobe Sign to remove the page on which this directive appears from the processed output for the document. The tag can appear anywhere on the page. A document sent to Adobe Sign must contain at least one page. If this directive appears on all the pages of a document sent to Adobe Sign, the first page would not be deleted.
-------------------------------------	---

Advanced Text Tags Examples

Below are a few examples of text tags with description for each tag.

Checkboxes

<code>{{[]_es_:signer1}}</code>	A check box, assigned to the recipient identified as signer1.
<code>{{[*]}</code>	A mandatory checkbox.
<code>{{[!]}</code>	A read-only checkbox.

Radio Buttons

<code>{{(Daily)Rate_es_:signer1}}</code>	A radio button assigned to signer1 with the value of "Daily"
--	--

Drop-down List

<code>{{*Color_es_:signer1:dropdown(options="Red,Green,Blue")}}</code>	A required drop-down list with three possible options. The list is assigned to recipient identified as signer1.
<code>{{Color_es_:signer1:dropdown(options="Red,Green,Blue"):default("Red")}}</code>	A drop-down list with three possible options and the option "Red" is selected by default. The list is assigned to recipient identified as signer1. If the value is specified in the definition of the drop-down list, the value must be used to specify the default otherwise the option is used.
<code>{{Color_es_:dropdown(options=Red,Green,Blue" , values="R,G,B"):default("R"):mask}}</code>	A drop-down list, not assigned to any recipient, with three possible options corresponding to three export values. The value "R" corresponding to the recipient choice of "Red" is selected by default and the resulting selected option gets masked (displayed as "*") in the PDF.
<code>{{Color_es_:dropdown(options="Red,Green,Blue"):showif(Rate="Daily")}}</code>	A drop-down list, not assigned to any recipient, with three possible options. The drop-down list is only shown if the value of the radio button called Rate is = "Daily".

Field Validation

<code>{{(Birthday_es_:signer2:isdate}}</code>	An optional date field for the recipient identified as signer2 to enter dates in US date format.
<code>{{*Zip2_es_:signer2:zip}}</code>	A required field for the recipient identified as signer2 to provide a valid zip code. The field validates the value entered by the recipient.

Shortening Text Tags

<code>{{\$r}}</code>	A reference tag specifying a location of a field.
<code>{{#r=[]Spouse_es_:prefill}}</code>	A definition for a check box to be prefilled by the sender.

Text tag shortening can also be used as functions and not just to replace an entire field.

For example, let's use the following tag:

```
{{field1_es_:signer1:showif(fieldA='Yes')}}
```

This above tag could also be shortened using the following:

```
{{field_es_: $A:$B}}
```

```
#A=signer
```

```
#B=showif(fieldA='Yes')
```

Text tag shortening also can behave like methods for calculations.

For example, let's use the following tag:

```
{{field1_es_:calc(a * b / c)}}
```

This above tag could also be shortened using the following:

```
Field1_es_:$1(a,b,c)
```

```
#1(option1, option2, option3) = calc([option1] * [option2] / [option3])
```

Conditional Rules

<code>{{f_es_:hideifall(sig=unsigned,initials=unsigned)}}</code>	A field that shows only if sig and initials are both signed.
<code>{{*emailAddress_es_:signer1:isemail:showif (TypeOfComm="receiveEmail")}}</code>	A required field for the recipient identified as signer1 to enter a valid email address. The field is only displayed when another field called "TypeOfComm" has a value of "receiveEmail". The field validates the value entered by the recipient.
<code>{{f_es_:hideifany(Color="Red",Color="Green")}}</code>	A field that is hidden if the recipient chooses either the color Red or the color Green in the drop-down list called "Color".

Calculated Fields

<pre> {{shipState_es_:signer1:calc([billingState]) :showif(sameAs=unchecked)}} </pre>	<p>A field in the document assigned to the first signer that is only displayed when another field (a checkbox) called "sameAs" is unchecked. The value of the field is automatically populated with the value of the field "billingState" that the signer has entered in the document.</p>
<pre> {{total_es_:signer1:calc(q1*pr1) :format(Number,"\$,0.00"):align(right)}} </pre>	<p>A field in the document assigned to the recipient identified as signer1. Its value gets automatically calculated as the product of the values in fields "q1" and "pr1". The resulting value is formatted as currency and displayed with a dollar (\$) symbol in front of the resulting value and containing two decimal points. The resulting value is also aligned right aligned within the field.</p>
<pre> {{price_es_:signer1:calc(if(pn1="W1",9.99, if(pn1="W2",5.99,if(pn1="W3",24.99,0))))}} </pre>	<p>A field in the document called "price" which gets calculated based on the value selected by the signer in another fields "pn1" in the document. In this example if field "pn1" equals W1 the value displayed in the "price" field is 9.99, if field "pn1" equals W2 the value displayed in the "price" field is 5.99 and so on. If none of the values match the value of the field "pn1" the value displayed in "price" is 0. Using this mechanism, calculated fields can be used to automatically populate the price of an item based on the selection that a user has made during signing.</p>

For more information about calculated fields see the [Calculated Fields Reference Guide](#).