

**Social Security Administration**



## **Section 508**

**Accessibility of Electronic  
and Information Technology  
for People with Disabilities**

[www.socialsecurity.gov](http://www.socialsecurity.gov)

**Guide to Accessible  
PDF Documents and Forms**



# Table of Contents

---

<b>INTRODUCTION .....</b>	<b>2</b>
<b>ACCESSIBLE PDF CHECKLIST .....</b>	<b>3</b>
<b>HOW TO CREATE ACCESSIBLE PDF DOCUMENTS .....</b>	<b>4</b>
Design with Accessibility in Mind .....	4
Manage Bookmarks .....	5
Run the Accessibility Full Check .....	6
Address the Problems Reported by the Accessibility Full Check.....	10
Define the Language(s) Used .....	11
Identify and Correct Reading Order Problems .....	14
Add Elements to the Tag Tree .....	15
Set the Tab Order of Pages .....	16
Use Reflow to Find Problems.....	17
Review Alternate Text.....	18
Change Tags to Artifacts.....	19
Correct Character Mappings .....	20
Fix Table Tags .....	21
Check Hyperlinks .....	24
Set Initial View Preferences .....	26
View Using High-Contrast Color Combinations .....	27
Expand Abbreviations .....	28
Check Color .....	29
Review Structure.....	30
Optimizing File Size in Acrobat .....	31
<b>HOW TO CREATE ACCESSIBLE PDF FILLABLE FORMS .....</b>	<b>32</b>
Percievable Requirements .....	33
Operable Requirements .....	35
Unambiguous Requirements.....	38
Robust Requirements .....	40
<b>AUTOMATED PDF TEST &amp; REMEDIATION TOOLS.....</b>	<b>41</b>
<b>SSA DESK GUIDE TO ACCESSIBLE PDF DOCUMENTS .....</b>	<b>42</b>
<b>SSA DESK GUIDE TO ACCESSIBLE PDF FILLABLE FORMS .....</b>	<b>43</b>

# Introduction

---

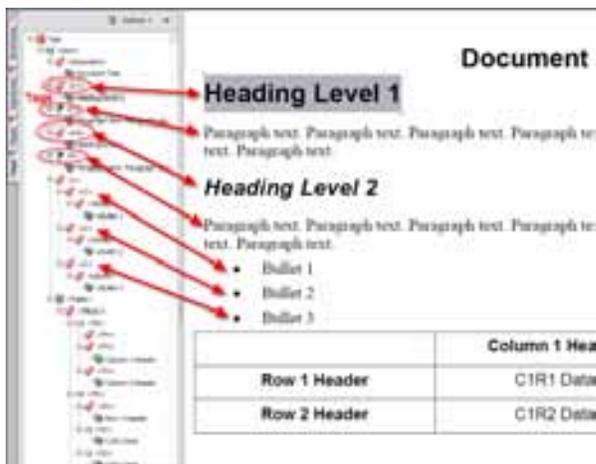
Most PDF documents today are simply converted from other formats and posted online or sent out as a document for printing purposes. While this method is fast and effective, it also creates problems for people with disabilities that want to access content of PDF document using assistive technology (screen reader). **The process of creating accessible PDF Documents is not automatic and requires time and effort to implement.** This document was created to help technical and non-technical staff create accessible Acrobat PDF documents. What follows is the guide to testing, repairing, adding and improving Accessibility features in Acrobat.

## **What is a PDF?**

Portable Document Format (PDF) was developed by Adobe for the distribution of electronic documents in a format that preserves the exact layout, font attributes, and formatting of the document from which it was created, ensuring that the electronic version of a document retains the exact look of the source material.

## **Tagged PDF**

Accessible PDF documents use tags to indicate the structural elements of the document's content, such as which page elements are headings, graphics, paragraphs, tables and so on, and how these elements relate to each other.



A properly tagged PDF will also integrate with a range of assistive technologies including Braille devices and screen readers.

## **Adobe Acrobat Help**

Many of the tools and techniques discussed are described in detail in the Complete Acrobat Help file, which can be accessed by pressing the F1 key while Acrobat is open and active.

# Accessible PDF Checklist

All or most of the responses to these checklist items should be "Yes" or "N/A" (not applicable). If a "No" response is used, the item should be addressed in an alternate fashion which renders the document accessible.

Test Condition	Yes/No
1. Bookmarks link to the appropriate destination.	
2. Bookmarks are nested properly.	
3. Bookmarks are labeled properly.	
4. All comments have been deleted.	
5. Document language has been defined.	
6. Foreign words or phrases have language definitions.	
7. Tab order is logical.	
8. There are no character mapping problems; or a workaround, such as alt text or artifacts, has been applied.	
9. All elements are contained within the tag tree.	
10. Read order is logical.	
11. Reflow is logical.	
12. Alternate text is used appropriately.	
13. Elements which should be ignored by a screen reader have been marked as artifacts.	
14. Empty tags have been deleted.	
15. Tables are marked up properly, using TH for column and row headings; or a workaround, such as descriptive text, has been applied.	
16. Hyperlinks are active.	
17. Hyperlinks link to the appropriate destination.	
18. Hyperlink text is descriptive of its destination.	
19. The initial view has been set to open the Bookmarks tab.	
20. The user's magnification settings have not been overridden, unless there is a strong reason to do so.	
21. All text is readable in each of Acrobat's high-contrast color combinations.	
22. Abbreviations have alternate text applied, if necessary.	
23. Information conveyed solely with color is available in another way, such as alternate text.	
24. Adequate structure has been applied to the document.	
25. The Accessibility Full Check returns no warnings.	
26. A screen reader has been used to validate that the document makes sense when read aloud.	

# HOW TO CREATE ACCESSIBLE PDF DOCUMENTS

---

## Design with Accessibility in Mind

### Design Method

Much of the work in creating an accessible PDF document happens in the creation of the document itself, not in the conversion to PDF. The key is to design the document in its source application, such as Microsoft Word, using techniques designed to enhance the accessibility of the final PDF document.

<b>To learn how to use Acrobat's:</b>	<b>Read this section:</b>
TouchUp Reading Order Tool	Editing the Structure of Tagged Adobe PDF Documents
Tags Tab	Advanced Tools for Correcting Tagging Errors
Bookmarks Tab	Adding Navigability to Adobe PDF Documents
Read Out Loud Feature	Customizing Adobe Acrobat for Accessibility > Using Read Out Loud feature

### Note

Whenever an issue is encountered within the PDF document, instructions have been given to correct the problem within Acrobat. However, it is at all times recommended that changes be made in the original Word document (if possible) and that it then be converted again to PDF.

Save the PDF file frequently, appending different version numbers to the filename. Some Acrobat tools do not include an Undo command, and Acrobat will sometimes crash. (An occasional **Save As** will reduce file size.)

## Manage Bookmarks

### **Test Method**

Review the bookmarks and determine if the bookmarks:

- link to the appropriate destination,
- are nested properly, and
- are labeled correctly.

Note: If Word's styles and Adobe PDFMaker's bookmarks feature were used properly, the PDF bookmarks should be correct. If they are not, they can be corrected in Word, or they can be modified in the Bookmarks tab within Acrobat.

If bookmarks are modified in the Bookmarks tab within Acrobat, those same modifications will need to be re-applied every time the document is converted from Word. It is recommended that bookmarks be corrected in Word.

### **Remediation Techniques**

#### ***To modify bookmarks in Acrobat:***

1. Open the **Bookmarks** tab by selecting **View > Navigation Tabs > Bookmarks** from the menubar.
2. Drag bookmarks to where they belong, create/delete bookmarks and/or apply formatting using the **Options** menu.

Adobe Acrobat Help: See *Adding Navigability to Adobe PDF Documents > Using Bookmarks* in the Complete Acrobat Help file (F1) for more detailed instructions.)

## Run the Accessibility Full Check

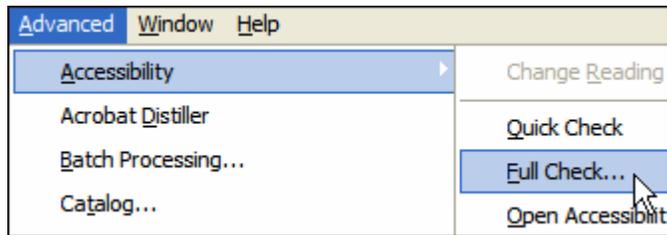
### Test Method

Run the Accessibility Full Check to find specific accessibility issues which may need attention.

### Remediation Techniques

***Check the accessibility of a PDF document using Full Check:***

1. Choose **Advanced > Accessibility > Full Check**.



2. Select **Create Accessibility Report**, and then click **Browse** to save a copy to the desired location.

**Accessibility Full Check**

**Report and Comment Options**

- Create Accessibility Report
- 
- Folder: C:\Documents and Settings\875408\My Documents
- Include repair hints in Accessibility Report
- Create comments in document

**Page Range**

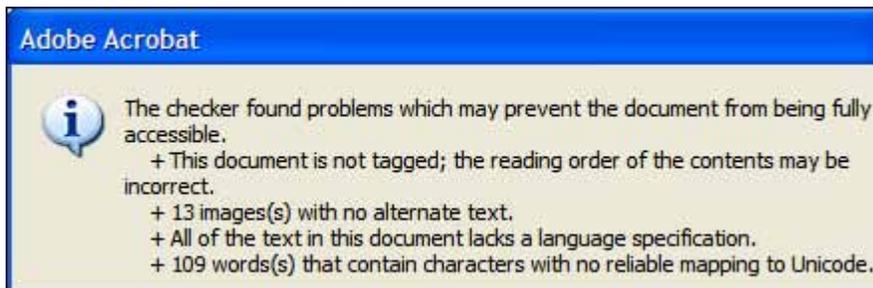
- All pages in document
- Only those pages selected in Pages panel
- Pages from  to

**Checking Options**

- Alternative descriptions are provided
- Text language is specified
- Reliable character encoding is provided
- All content is contained in the document structure
- All form fields have descriptions
- List and table structure is correct
- Tab order is consistent with the structure order

3. Specify the pages to be included in the accessibility check.
4. To view the report as comments in the PDF document, select **Create Comments In Document**.
5. Select all of the checkboxes for the desired **Checking Options**.
6. Click **Start Checking**.

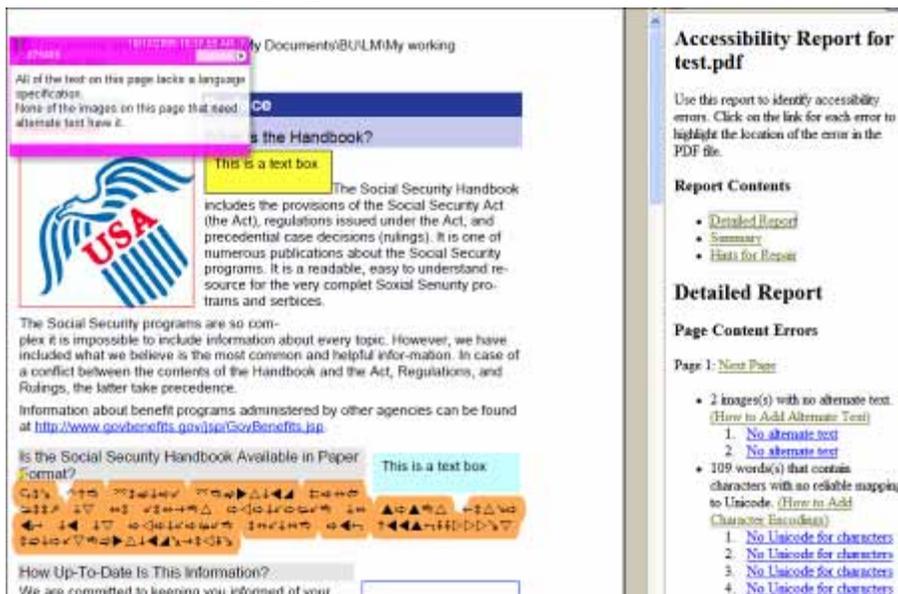
A dialog opens reporting the problems found in the document.



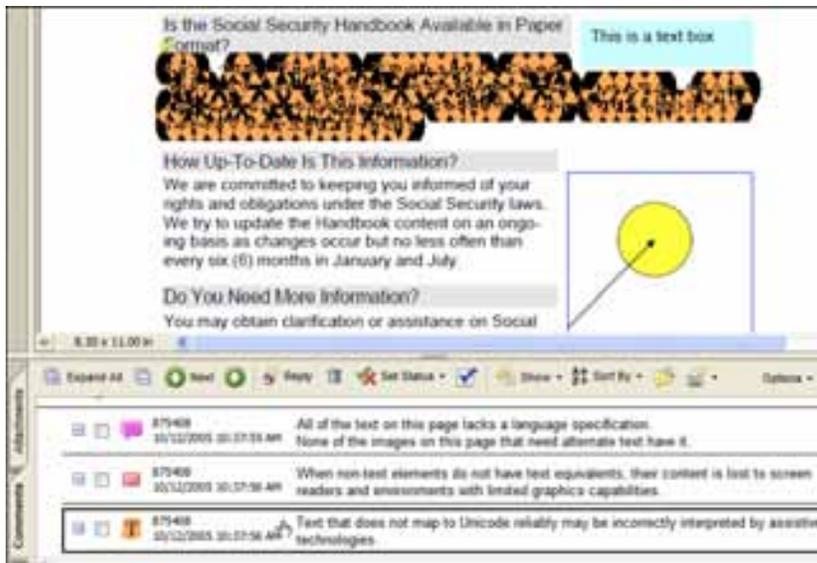
Note the problems found and dismiss the dialog by selecting **OK**.

A report opens in the How To window with a list of problems and links to specific areas of the document.

Comments and highlights within the document indicate potential problems.



To view and highlight problem individually, open the **Comment** tab and select individual comments within the **Comments** pane. The corresponding element in the document will be highlighted.



After addressing each problem noted by a comment, delete the comment (otherwise it will appear as an untagged element the next time the Full Check is run).

To delete a comment, select the comment to be deleted within the **Comments** tab, and then press **Delete**.

## Address the Problems Reported by the Accessibility Full Check

### **Warnings**

After running the Accessibility Full Check, a dialog opens reporting the problems found in the document.

For each problem reported, see the topics noted to eliminate the problem.

<b>Accessibility Full Check Message</b>	<b>See Topic</b>
All of the text in this document lacks a language specification.	<a href="#">Define the Language(s) Used</a>
[Number of] elements that are not contained within the structure tree.	<a href="#">Adding Elements to the Tag Tree</a>
[Number of] pages with tab order that may be inconsistent with the structure order.	<a href="#">Set the Tab Order of Pages</a>
[Number of] images and/or Figure elements with no alternate text.	<a href="#">Alternate Text</a> and <a href="#">Artifacts</a>
[Number of] words that contain characters with no reliable mapping to Unicode.	<a href="#">Character Mappings</a>

## Define the Language(s) Used

### Test Method

Ensure the Default Document Language is set to the appropriate language. Listen to the document using a screen reader. Are all non-English words pronounced correctly? If they are not, the language property for individual words and phrases may need to be set so that the proper pronunciation dictionary is used. (In theory, screen readers should change the pronunciation dictionary in order to pronounce words according to the language in which they are written. This is not always the case with today's screen readers.)

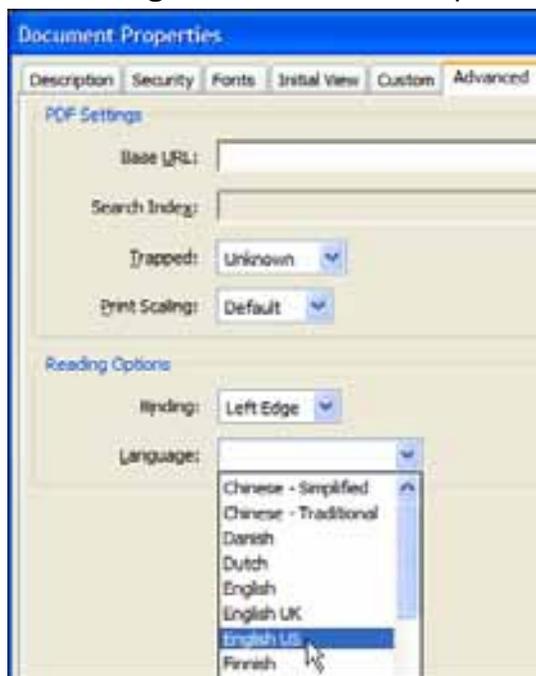
### Note

The document and text language properties can only be applied within Acrobat. If changes are made in the source document, these properties will need to be set again after conversion.

### Remediation Techniques

#### ***Set the default document language.***

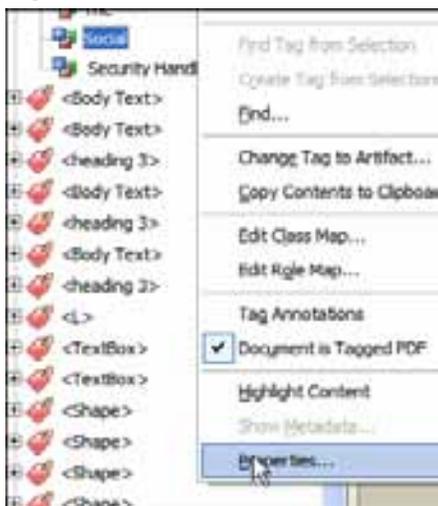
1. From the menubar, select **File > Document Properties > Advanced tab > Reading Options > Language**
2. Select **English US** from the dropdown list.



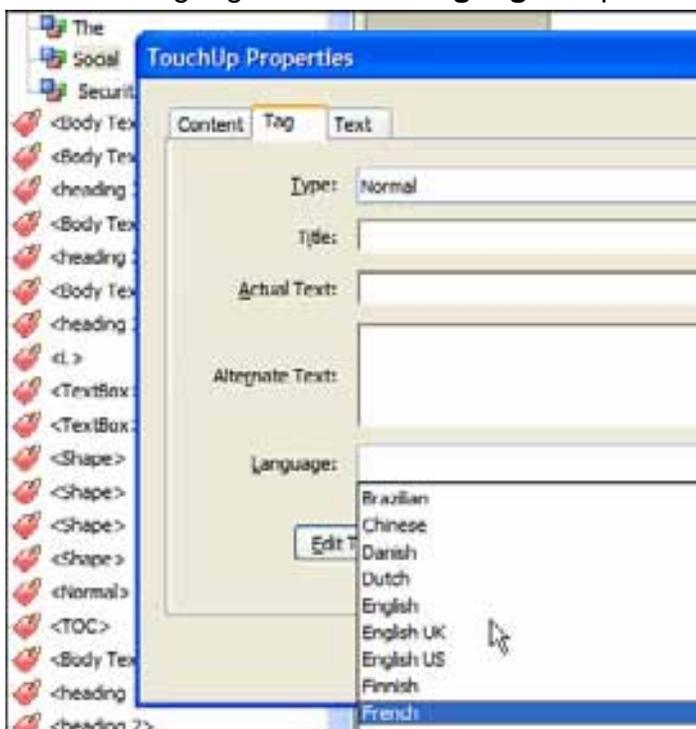
3. Select **OK**.

**Set the language for portions of the text that use a different language from the document's default language.**

1. Open the **Tags** tab by selecting **Navigation Tabs > Tags** from the menubar.
2. In the **Tags** tab, expand the tag tree as needed to see the elements.
3. Select the appropriate text element, and choose **Properties** from the **Options** menu.



4. In the **TouchUp Properties** dialog box, select the **Tag** tab.
5. Select a language from the **Language** dropdown list menu.



6. Select **Close**.

**Note:**

The language specified for an element also applies to all elements nested under it in the logical structure tree.

(See *Editing the Structure of Tagged Adobe PDF Documents* and *Advanced Tools for Correcting Tagging Errors* in the Complete Acrobat Help file (F1) for more detailed instructions.)

## Identify and Correct Reading Order Problems

### **Test Method**

To determine if a document has a logical read order, it will be necessary to listen to it using a screen reader, outputting it to an alternate file format, viewing it with the TouchUp Reading Order tool, and/or reflowing it.

### **Remediation Techniques**

***Check read order using any or all of the following techniques:***

- Listen to the document using a screen reader.
- Output the document to a text file and read it.
- Reflow the document.
- View the document using the TouchUp Reading Order tool.

***Repair read order problems using the Tags tab or the TouchUp Reading Order tool.***

(See Editing the Structure of Tagged Adobe PDF Documents and Advanced Tools for Correcting Tagging Errors in the Complete Acrobat Help file (F1) for more detailed instructions.)

## Add Elements to the Tag Tree

### **Test Method**

Determine if there are elements in the document which are not represented within the logical structure tree.

### **Remediation Techniques**

***Add elements to the tag tree using the Tags tab or the TouchUp Reading Order tool.***

(See Editing the Structure of Tagged Adobe PDF Documents and Advanced Tools for Correcting Tagging Errors in the Complete Acrobat Help file (F1) for more detailed instructions.)

## Set the Tab Order of Pages

### Test Method

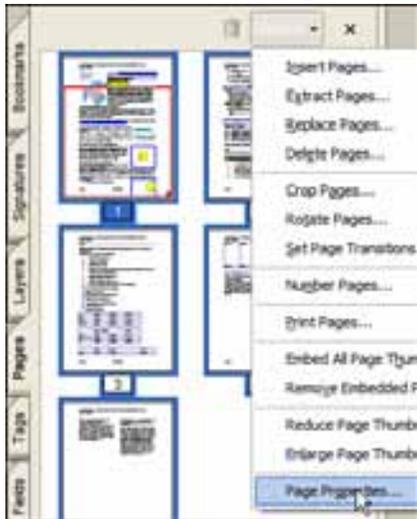
Listen to the document using a screen reader to determine if the document content is read in an order that makes sense.

The tab order often, but not always, passes through from the Word document. Explicitly setting the default tab order of all pages ensures that the tab order of pages is in a logical order.

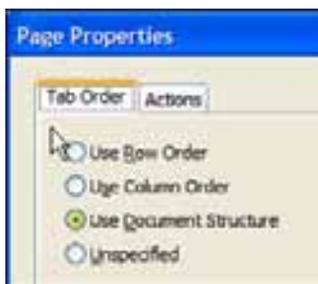
### Remediation Techniques

#### ***Set the tab order of all pages.***

1. Open the **Pages** tab by selecting **Navigation Tabs > Pages** from the menubar.
2. In the **Pages** tab, select all pages
3. Select **Options > Page Properties > Tab Order** tab.



4. Select the radio button for **Use Document Structure**.



## Use Reflow to Find Problems

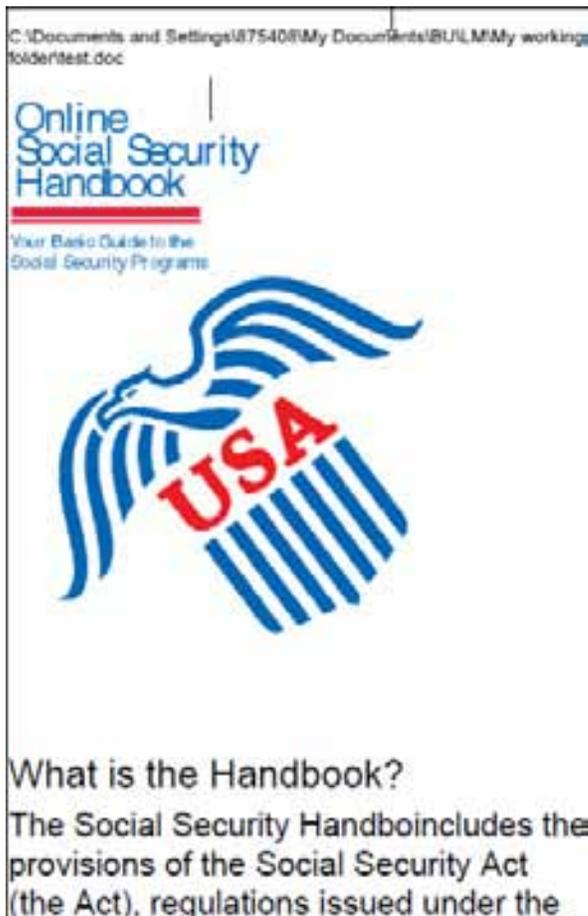
### **Test Method**

Reflow the document. If the tagged PDF document does not reflow properly, the content order of the PDF file may contain inconsistencies, or the tagging process itself may be the cause.

### **Remediation Techniques**

***Reflow the document.***

**View > Reflow.**



***Repair problems using the Tags tab or the TouchUp Reading Order tool.***

(See Editing the Structure of Tagged Adobe PDF Documents and Advanced Tools for Correcting Tagging Errors in the Complete Acrobat Help file (F1) for more detailed instructions.)

## Review Alternate Text

### Test Method

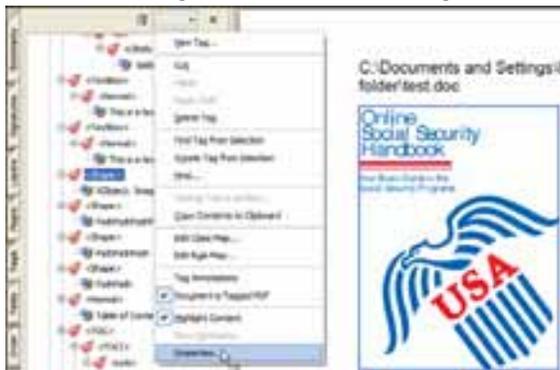
Ensure that alternate text makes sense within the context of the document when using a screen reader.

### Remediation Techniques

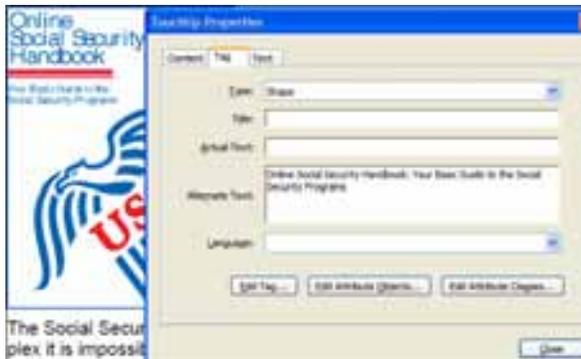
***Return to the Source Document and add alternate text to all elements which require it.***

Alternately, to **add alternate text** within Acrobat:

1. Choose **View > Navigation Tabs > Tags**.
2. Expand the logical structure tree to find and select the **<Figure>** tag element for the image.
3. Choose **Highlight Content** from the **Options** menu on the **Tags** tab to see a highlighted area in the document that corresponds to the tag.
4. Choose **Properties** from the **Options** menu on the **Tags** tab.



5. In the **TouchUp Properties** dialog box, click the **Tag** tab.
6. For **Alternate Text**, type text that describes the figure.



## Change Tags to Artifacts

### Test Method

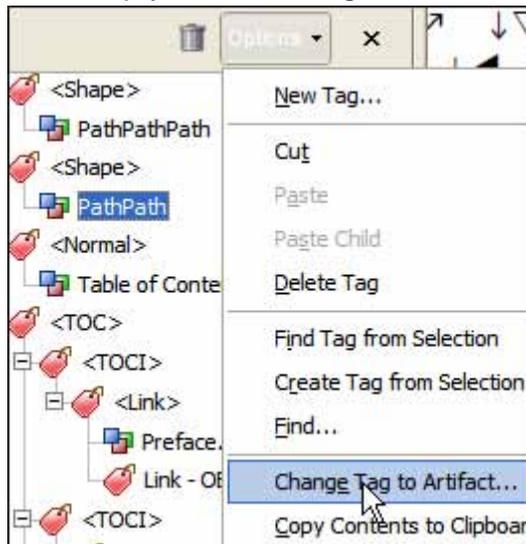
Ensure that elements which should not be read aloud by the screen reader are not. To hide an element from the screen reader, it should be marked as an artifact.

### Remediation Techniques

***Return to the Source Document and use a carriage return as the alternate text for all elements which require it.***

Alternately, to ***change an element to an artifact*** within Acrobat:

1. In the **Tags** tab, expand the section of the logical structure that needs to be changed.
2. Select an element.
3. Choose **Change Tag to Artifact** from the **Options** menu.
4. If an empty container tag is left behind, delete it.



## Correct Character Mappings

### **Test Method**

The Accessibility Full Check is the best indicator of character mapping issues.

### **Remediation Techniques**

***Return to the Source document and use only characters which do not cause a character mapping error in Acrobat.***

***Alternately, treat these elements as images and add alternate text, or if the character is decorative (such as a bullet), change it to an artifact.***

Keep in mind that "correcting" the problem in this way will affect the extractability of text content, which may have repercussions if the document needs to be repurposed.

## Fix Table Tags

### **Test Method**

Review the table tags to determine whether columns, rows, and cells have been correctly identified.

### **Note**

Tables that lack well-defined borders and rules are often tagged incorrectly or contain adjacent page elements. Poorly tagged tables can be corrected by selecting and redefining them or by splitting combined cells by creating a tag for each cell.

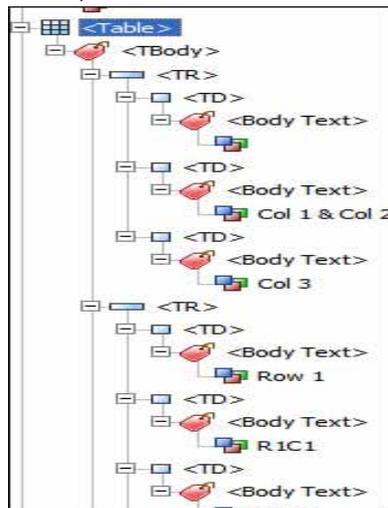
Tables pose a special challenge for screen readers because they present textual or numerical data to be easily referenced visually. Content within table cells can be complex and might contain lists, paragraphs of text, form fields, or another table.

### **Remediation Techniques**

#### ***Check Tab Elements***

In the Tags tab, expand the tags root to view a table tag.

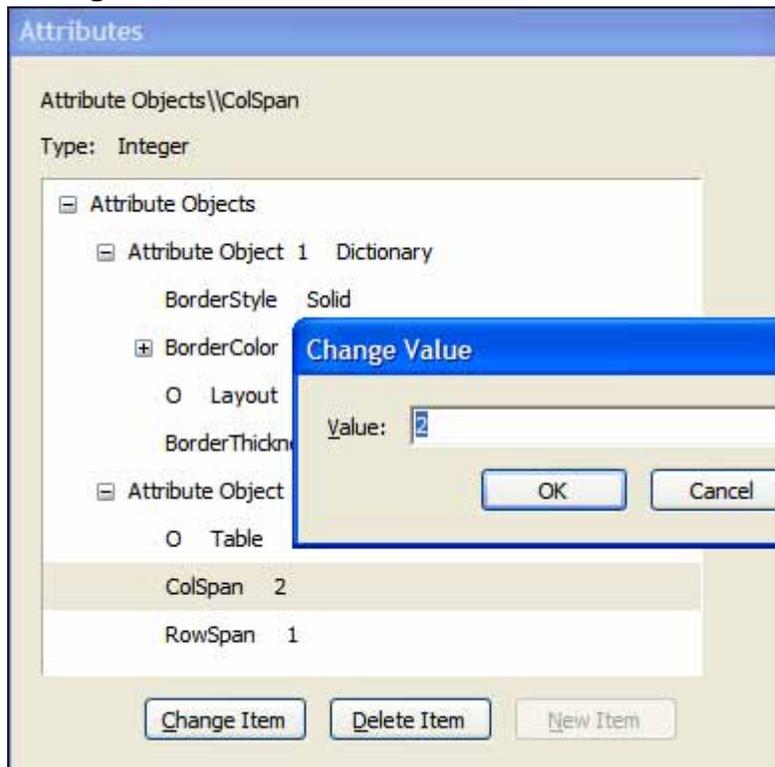
1. Select the table tag `<Table>` and verify that it contains one of the following elements:
  - Table Rows `<TR>`, each of which contains a Table Header `<TH>` or Table Data `<TD>` cells.
  - `<THead>`, `<TBody>`, and `<TFoot>` sections, each of which contains Table Rows. (The table rows contain `<TH>` cells, `<TD>` cells, or both.)



2. Do one or more of the following:
  - If the tag for the table does not contain these elements but rows, columns, and cells appear in the table in the document pane, use the TouchUp Reading Order tool to select and define the table or individual cells.
  - If the table contains rows that span across two or more columns, set ColSpan and RowSpan attributes for these rows in the tag structure.
  - Re-create the table in the authoring application, and then convert it to a tagged PDF document.

**Set ColSpan and RowSpan attributes:**

1. In the **Tags** tab, select a <TD> or <TH> element.
2. Choose **Properties** from the **Options** menu.
3. In the **TouchUp Properties** dialog box, click the **Tags** tab, and then click **Edit Attribute Objects**.
4. Select **Attribute Objects**, and then click New Item to create a new Attribute Object Dictionary.
5. Expand the new dictionary, select the  **Layout** attribute, and then click **Change Item**.



6. Change the **Layout** value to **Table**.
7. Select the **Attribute Object Dictionary**, and click **New Item**.

8. In the **Add Key And Value** dialog box, type **ColSpan** or **RowSpan** in the **Key** box; enter the number of columns or rows spanned in the Value box; choose Integer from the **Value Type** pop-up menu; and click **OK**.
9. Close the **TouchUp Properties** dialog box.

## Check Hyperlinks

### Test Method

Review the all document hyperlinks and determine if links are:

- accurate,
- active, and
- descriptive or their destination.

### Remediation Techniques

***Review all links, and effect repairs in the original Word document.***

If there is an issue with a very long link that simply cannot be made short enough to fit on a single line, use the following technique to make it a single link in Acrobat:

1. Open the Advanced Editing toolbar:

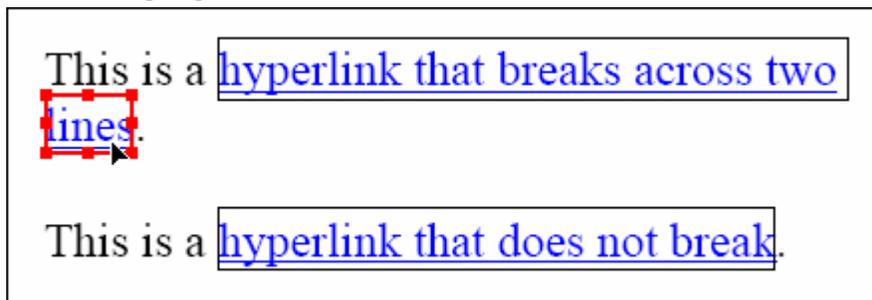
**View > Toolbars > Advanced Editing**



2. Select the link tool:

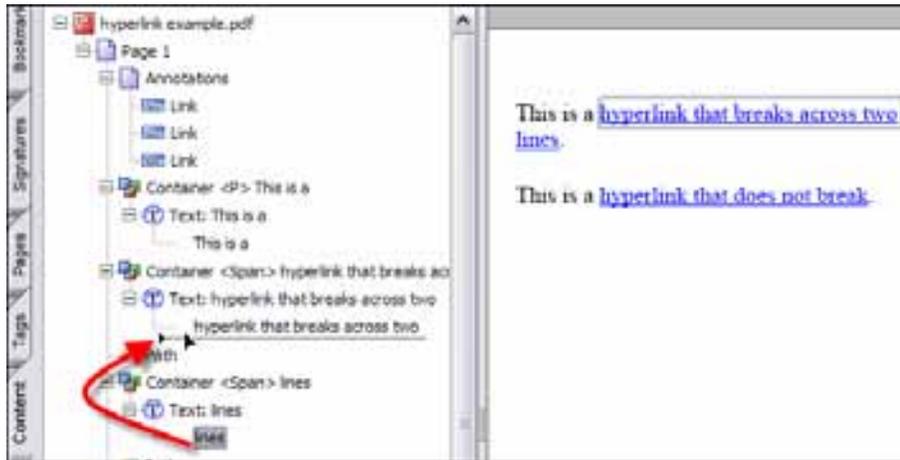


3. Links are highlighted:

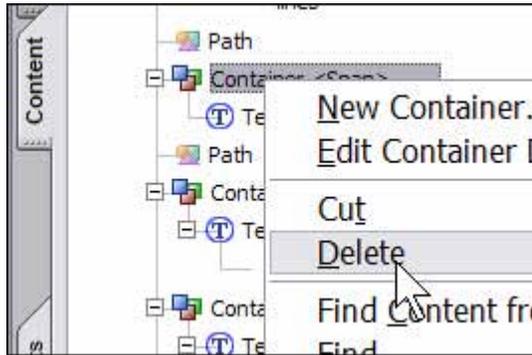


4. Select the link box of the second part of the link and delete it.

5. Open the Content tab, find the second part of the desired link text, and move the text so that it is below the first part of the link text within the same container.



6. Delete the empty container element left behind.



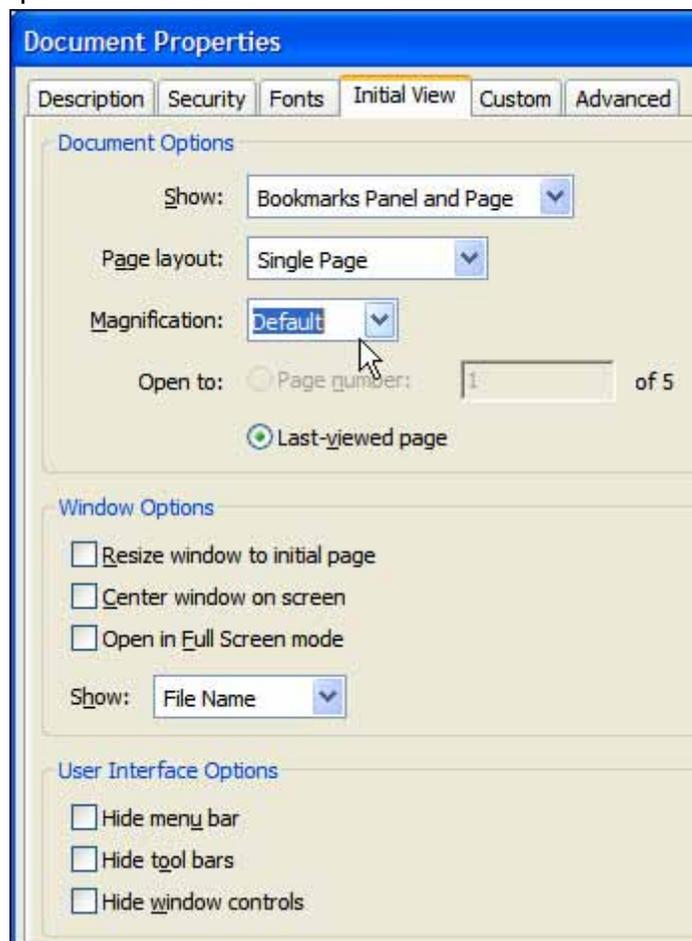
## Set Initial View Preferences

### **Test Method**

Open the document to determine if the initial view is as desired.

### **Remediation Techniques**

1. Set the desired initial view properties.
2. Select **File > Document Properties > Initial View** tab.
3. Set the **Document Options > Show** dropdown list to **Bookmarks Panel and Page** and **Page**.
4. Leave the magnification set to default unless there is a strong reason to override the user's preferences. Also, leave the Window and User Interface Options unchecked, unless there is a strong reason to override the user's preferences.



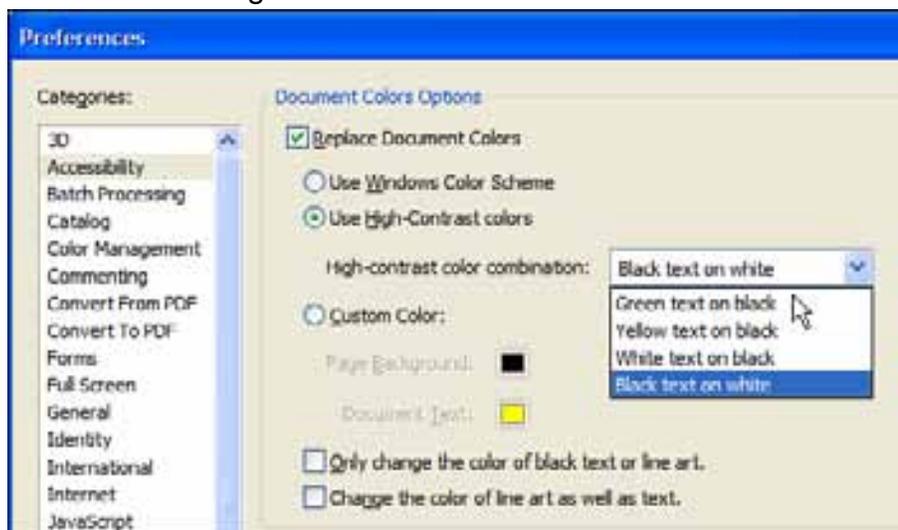
## View Using High-Contrast Color Combinations

### Test Method

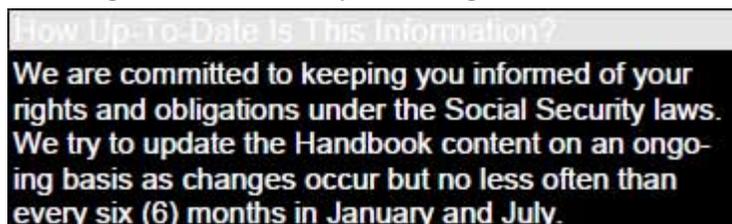
Determine if all text is still readable when the document is viewed using each of Acrobat's 4 high contrast color combinations.

### Remediation Techniques

1. View the document using Acrobat's various high contrast modes.
2. **Edit > Preferences > Accessibility**
3. Select various High-contrast color combinations and review the document.



The following example illustrates why background color should not be used with text. When viewed using high-contrast black text on white, the heading becomes nearly indistinguishable.



## Expand Abbreviations

### **Test Method**

When a screen reader is used, determine if the meaning of abbreviations is clear.

### **Remediation Techniques**

To add alternate text for an abbreviated term:

1. In the **Tags** tab, locate the abbreviated term by doing one of the following:
2. Expand the tag tree as needed to see the elements that contain the abbreviation.
3. Use the **TouchUp Text** tool or the **Select** tool to select the abbreviation in the document, and then choose **Find Tag From Selection** from the **Options** menu to locate the text in the tag tree.
4. Select the tag for that element, and choose **Properties** from the Options menu.  
**Note:** If the abbreviation includes additional text, cut the additional text and place it in a new <Span> child tag within the same <Span> parent tag.
5. In the **TouchUp Properties** dialog box, select the **Tag** tab.
6. For **Alternate Text**, type the unabbreviated version of the term.
7. Click **Close**.

## Check Color

### **Test Method**

Is color used to convey information that is not otherwise available?

When viewed in high contrast mode, can the document still be read visually?

### **Remediation Techniques**

***Return to the Source Document to correct color problems.***

## Review Structure

### Test Method

Determine if structural elements, such as H1, H2, LI, P, TABLE, etc. have been used adequately.



### Remediation Techniques

***Repair restructure problems using the Tags tab or the TouchUp Reading Order tool.***

(See Editing the Structure of Tagged Adobe PDF Documents and Advanced Tools for Correcting Tagging Errors in the Complete Acrobat Help file (F1) for more detailed instructions.)

# Optimizing File Size in Acrobat

## Why It's Important

Reducing the size of Adobe PDF files improves their performance—particularly when they're being accessed on the web.

## Optimization Techniques

**Use PDF Optimize** to optimize the PDF file:

1. **Download the accessible.optimize file provided** at the following location:  
[<http://www.xxx.xxx/accessible.optimize>]
2. **Save the accessible.optimize file to the Acrobat PDF Optimizer folder**, usually located here:  
C:\Documents and Settings\your PIN\Application Data\Adobe\Acrobat\7.0\Preferences\PDF Optimizer  
**Note:** Use the optimizer file provided until the PDF Optimizer interface has been mastered.
3. Select **Advanced > PDF Optimizer** from the menubar.



4. The **PDF Optimizer** dialog opens.
5. Select **accessible** from the **Preset** dropdown list.



6. Select **OK**.
7. Save the file using a new filename.

# HOW TO CREATE ACCESSIBLE PDF FILLABLE FORMS

---

## Introduction

To be considered accessible, the content of a PDF fillable form must be:

- perceivable,
- operable,
- unambiguous, and
- robust.

The presence of these four attributes allows a PDF to be usable by the vast majority of users, including older users and those with disabilities, and enables access using a variety of devices and assistive technologies.

In addition, the Social Security Administration has a few requirements which should be included in all SSA accessible PDF fillable forms.

Note: Section “How to Create Accessible PDF Documents” is the pre-requisite and vastly recommended before proceeding with this section on Fillable Forms.

## Definitions

### Perceivable

Content is designed to allow presentation according to the user's needs and preferences.

### Operable

User interface components in the content are designed to allow interaction according to the user's needs and preferences.

### Unambiguous

Content is linguistically and semantically explicit, and designed to be understandable by its intended audience.

### Robust

Content is designed to be compatible with current and future technologies.

## Percievable Requirements

### **Text**

#### **Text must be available to assistive technology**

- All text used in hidden text fields must be error free (spelling, punctuation, bullets, etc.)
- Text must not extend past visible limit of input box.
- All abbreviations and acronyms must be spelled out.
- Multi-line fields must indicate line number (for example: Line one of three, line two of three, and line three of three.) In addition, multi-line fields must provide question at the beginning and end lines.

### **Text Extraction**

#### **Text can be accurately and programmatically determined.**

- Content is structurally tagged.
- Fonts contain enough information to be machine readable.
- Hyphenation has been used appropriately.
- Content is searchable.

### **Text Equivalents**

#### **Alt text is provided for meaningful non-text content.**

- For all non-text content that is used to convey information, text alternatives identify the non-text content and convey the same information.
- For functional non-text content, text alternatives serve the same purpose or identify the purpose of the functional non-text content.
- Non-text content that is not functional, and does not convey information is implemented such that it can be ignored by assistive technology.

### **Multimedia Alternatives**

#### **Alternative content is provided for meaningful multimedia.**

- For multimedia, a text-alternative identifies the multimedia.
- For live audio-only or live video-only content, text alternatives at least identify the purpose of the content with a descriptive label.
- For prerecorded multimedia content, a combined transcript of captions and audio descriptions of video is available.
- Synchronized alternatives are provided for embedded multimedia.
- Captions are provided for prerecorded multimedia.
- Audio descriptions of video are provided for prerecorded multimedia.
- Real-time captions are provided for live multimedia.
- Sign language interpretation is provided for multimedia.

- Extended audio descriptions of video are provided for prerecorded multimedia.
- Audio descriptions of video are provided for live multimedia.

### **Links to Plug-ins**

**Links are provided to accessible plug-ins if they are necessary.**

- If an applet, plug-in or other application is required to be present on the client system to interpret content, a link is provided to the necessary applet, plug-in or application.
- The applet, plug-in or application necessary to interpret the content must be accessible.

### **Non-essential Color & Sufficient Contrast**

**Meaningful content can be distinguished regardless of color or contrast.**

- When information is conveyed by color, the color can be programmatically determined or the information is also conveyed through another means that does not depend on the user's ability to differentiate colors.
- Any information that is conveyed by color is visually evident when color is not available.

### **Security Set Appropriately**

**If security settings are used, screen reader access is not disabled.**

## Operable Requirements

### **Keyboard Only Operation**

**All functionality is operable via a keyboard.**

- All of the functionality of the content is operable through a keyboard interface.
- If the user can enter the content using the keyboard, then the user can exit the content using the keyboard.

### **Navigation & Orientation**

**Provide mechanisms to help users find content, orient themselves within it, and navigate through it...**

- Bookmarks are used for documents longer than 2 pages.
- Bookmarks and other links are identified through words or phrases that describe their destination.
- Repetitious content which does not add to meaning is implemented such that it can be bypassed.
- Effective metadata has been used; at a minimum a descriptive document title and the author have been identified.

### **User-directed Interactivity**

**Content is designed to allow interactivity according to the user's needs and preferences.**

- Changes of context are initiated only by user action.
- A mechanism is available to allow users to:
  - Turn off background audio that plays automatically.
  - Avoid content that could cause seizures due to photosensitivity.
  - Pause moving content.
  - Postpone or suppress non-emergency interruptions.
- Content is designed so that time-outs are not an essential part of interaction, or at least one of the following is true for each time-out that is a function of the content:
  - Time-outs can be deactivated by the user or;
  - Time-out settings can be adjusted by the user or;
  - Warnings are provide before time expires, and time-outs can be extended with a simple action (for example, "hit any key") and given at least 20 seconds to respond or;
  - The time-out is an important part of a real-time event (for example, an auction), and no alternative to the time-out is possible or;
  - The time-out is part of an activity where timing is essential (for example, competitive gaming or time-based testing), and time limits can not be extended further without invalidating the activity.
- When an authenticated session has an inactivity timeout, the user can continue the activity without loss of data after re-authenticating.

## **Functional Tooltips**

### **Use tooltips to provide information indicating what input is desired.**

- Include only enough information in the tooltip for users to know what input is expected and how to complete the field correctly.
- Indicate form inputs which are required.
- Indicate if special keystrokes are required to perform a function.
- Tooltips should not state the input type or state; this information is passed programmatically to assistive technology.

## **Form Fields & Check Boxes**

### **Use appropriate field types to collect user input.**

- Provide descriptive labels for each user interface control that accepts input from the user.
- Use checkboxes for items which require 0 to many selections. Check boxes should have more than 2 items, the first and last check boxes must speak the question associated with the check box.
- Check boxes must speak the appropriate state of the check box when selected or deselected.
- Use radio button for items which require only 1 selection.
- Avoid using radio buttons if a list box can be used instead. (Radio buttons are often misinterpreted or difficult for users with disabilities to access because web browsers and screen readers treat them inconsistently.)
- Size text fields appropriately and use character restrictions to prevent scrolling.

## **Error Prevention**

### **Help users avoid mistakes and make it easy to correct them.**

- If an input error is detected, the error is identified and provided to the user in text.
- If an input error is detected and suggestions for correction are known and can be provided without jeopardizing the security or purpose of the content, the error is identified and the suggestions are provided to the user.
- For forms that cause legal or financial transactions to occur, that modify or delete data in remote data storage systems, or that submit test responses, at least one of the following is true:
  - Actions are reversible.
  - Actions are checked for errors before going on to the next step in the process.
  - The user is able to review and confirm or correct information before submitting it.

## **Tooltips**

### **Use appropriate tooltips to provide instructions to users**

- All directions and cues must be included within the tooltip.

### **Accessible Scripting**

**Scripting is written to avoid problems with screen readers and keyboards.**

- Scripts which use mouse events also use equivalent keyboard events.
- Information provided or changed by scripts shall be programmatically identified so that it can be read by assistive technology.

### **Multi-copy**

**Reduce duplicate keying of identical information**

- Accessible multi-copy forms should automatically populate identical fields on subsequent copies.

### **Special Instructions**

**Maintain instructions on original forms**

- Instructions like “use blue or black ink only” must be retained.

### **Signatures**

**Signature fields should be non-editable**

- Signature fields should be part of the tab order, but should be read-only.

# Unambiguous Requirements

## **Defined Language**

**Content is readable and understandable.**

- The language of all text in the document is identified programmatically.
- Language of foreign words and phrases is identified programmatically.
- A mechanism is available for finding definitions of unusual words, idioms, jargon, slang, acronyms and abbreviations.

## **Logical Order**

**Document structure accurately represents content arranged in a sequence that affects its meaning.**

- A logical read order that includes all fields and buttons is identified programmatically.
- A logical tab order that includes all fields and buttons is identified programmatically.

## **Hierarchical Structure**

**Content is structurally (semantically) tagged.**

- Structural elements, such as headings, paragraphs, lists, and tables, are used to represent corresponding content.
- Structural elements indicate the relationships among and between elements.
- The document tag structure effectively indicates the correct reading order.

## **Data Table Structure**

**Tables are tagged structurally to provide information about their contents.**

- Row and column headers are identified for data tables.
- Markup is used to associate data cells and header cells for data tables that have two or more logical levels of row or column headers.
- Merged cells that span two or more associated columns or rows use the ColSpan or RowSpan attribute as appropriate.

## **Predictable Context**

**Changes of context are initiated only by user action.**

- When any component receives focus, it does not cause a change of context.
- Changing the setting of any input field does not automatically cause a change of context.
- Any change of context is implemented in a manner that can be programmatically determined.

- Changes to content, structure, selection, focus, attributes, values, state, and relationships within the content can be programmatically determined.

## **Consistency**

### **Placement and functionality of content is predictable and consistent.**

- Components that are repeated occur in the same order each time they are repeated.
- Components that have the same functionality are labeled consistently.
- Graphical components that appear on multiple pages, including graphical links, are associated with the same text equivalents wherever they appear.
- When content is navigated sequentially, elements receive focus in an order that follows relationships and sequences in the content.
- Place form labels next to their corresponding form controls.
- Labels for text boxes appear before the control
- Labels for radio buttons and checkboxes appear after the control.

## Robust Requirements

### **Compatibility**

**Content is compatible with current and future technologies.**

- Technologies are used according to published specifications.
- Accessibility conventions of the application, markup and/or programming language (API's or specific markup) are used.

# AUTOMATED PDF TEST & REMEDIATION TOOLS

## Introduction

Automated tools allow for easy and fast way to add accessible features to documents. Today, vast amount of time is spent on adding and editing accessible features such us: alternative text, tab index, etc.

NetCentric CommonLook is a plug-in application for Adobe Acrobat that can significantly reduce the time it takes to make PDF documents and fillable forms accessible. The chart below identifies the test conditions that can be automatically checked by the tool, versus the conditions that still must be tested manually. However, it is important to note that the tool covers the most time consuming test conditions.

Please note that manual interaction is required to run the NetCentric tool. It does not provide “push button” functionality, but rather productivity enhancing features that make it easy for a tester to find and repair problems. Limited copies of NetCentric CommonLook plug-in are available upon request from the Accessibility Resource Center. Please contact the SSA Section 508 Coordinator Robert Baker at robert.c.baker@ssa.gov for more information.

<b>Test Conditions</b>	<b>Manual/Automatic(NetCentric)</b>
1. Bookmarks link to the appropriate destination.	
2. Bookmarks are nested properly.	
3. Bookmarks are labeled properly.	
4. All comments have been deleted.	
5. Document language has been defined.	Supported
6. Foreign words or phrases have language definitions.	Supported
7. Tab order is logical.	Supported
8. There are no character mapping problems; or a workaround, such as alt text or artifacts, has been applied.	
9. All elements are contained within the tag tree.	
10. Read order is logical.	Supported
11. Reflow is logical.	Supported
12. Alternate text is used appropriately.	Supported
13. Elements which should be ignored by a screen reader have been marked as artifacts.	Supported
14. Empty tags have been deleted.	Supported
15. Tables are marked up properly, using TH for column and row headings; or a workaround, such as descriptive text, has been applied.	Supported
16. Hyperlinks are active.	
17. Hyperlinks link to the appropriate destination.	
18. Hyperlink text is descriptive of its destination.	
19. The initial view has been set to open the Bookmarks tab.	
20. The user's magnification settings have not been overridden, unless there is a strong reason to do so.	
21. All text is readable in each of Acrobat's high-contrast color combinations.	
22. Abbreviations have alternate text applied, if necessary.	
23. Information conveyed solely with color is available in another way, such as alternate text.	Supported
24. Adequate structure has been applied to the document.	Supported
25. The Accessibility Full Check returns no warnings.	
26. A screen reader has been used to validate that the document makes sense when read aloud.	

## SSA Desk Guide to Accessible PDF Documents

Feature	Description
<b>Perceivable</b>	<b>Content is designed to allow presentation according to the user's needs and preferences.</b>
<b>Text</b>	<ul style="list-style-type: none"> <li>All text used in hidden text fields must be error free (spelling, punctuation, bullets, etc.)</li> <li>Text must not extend past visible limit of input box.</li> <li>All abbreviations and acronyms must be spelled out for the initial occurrence in a form.</li> <li>Multi-line fields must indicate line number (for example: Line one of three, line two of three, and line three of three.) In addition, multi-line fields must provide question at the beginning and end lines.</li> </ul>
<b>Text Extraction</b>	<ul style="list-style-type: none"> <li>Text can be accurately and programmatically determined.</li> </ul>
<b>Text Equivalents</b>	<ul style="list-style-type: none"> <li>Alt text is provided for meaningful non-text content.</li> </ul>
<b>Multimedia Alternatives</b>	<ul style="list-style-type: none"> <li>Alternative content is provided for meaningful multimedia.</li> </ul>
<b>Links to Plug-ins</b>	<ul style="list-style-type: none"> <li>Links are provided to accessible plug-ins if they are necessary to render meaningful content.</li> </ul>
<b>Color &amp; Sufficient Contrast</b>	<ul style="list-style-type: none"> <li>If color is used, utilize sufficient color and contrast.</li> </ul>
<b>Security Set Appropriately</b>	<ul style="list-style-type: none"> <li>If security settings are used, screen reader access is not disabled.</li> </ul>
<b>Operable</b>	<b>User interface components in the content are designed to allow interaction according to the user's needs and preferences.</b>
<b>Well-defined Focus</b>	<ul style="list-style-type: none"> <li>Static text used for instructions or information must have a well defined focus or caret, so that JAWS speaks it, and MAGic focuses on it.</li> <li>All navigation and input in PDF must be available via JAWS screen reader.</li> </ul>
<b>Keyboard Only Operation</b>	<ul style="list-style-type: none"> <li>All functionality of the content is designed to be operated through a keyboard interface.</li> <li>Auto skip for multiple fields (like SSN) should be avoided as they can cause issues for keyboard access.</li> <li>Avoid OnChange events that refresh the screen because they interfere with keyboard access.</li> </ul>
<b>Navigation &amp; Orientation</b>	<ul style="list-style-type: none"> <li>Provide mechanisms to help users find content, orient themselves within it, and navigate through it.</li> </ul>
<b>Special Instructions</b>	<ul style="list-style-type: none"> <li>Instructions such as "use blue or black ink only" must be retained.</li> </ul>
<b>User-directed Interactivity</b>	<ul style="list-style-type: none"> <li>Content is designed to allow interactivity according to the user's needs and preferences.</li> </ul>
<b>Functional Tooltips</b>	<ul style="list-style-type: none"> <li>All directions and cues must be included within the tooltip.</li> </ul>
<b>Accessible Scripting</b>	<ul style="list-style-type: none"> <li>Scripting is written to avoid problems with screen readers and keyboards.</li> </ul>
<b>Unambiguous</b>	<b>Content is linguistically and semantically explicit, and designed to be understandable by its intended audience.</b>
<b>Defined Language</b>	<ul style="list-style-type: none"> <li>Content is readable and understandable.</li> </ul>
<b>Tab Order</b>	<ul style="list-style-type: none"> <li>Tab order must represent the logical navigation of the document.</li> <li>The last field in a form should have the ability to tab back to the first field or go on to instructional text by tabbing.</li> <li>The last form field on the page must go to the next page, if there are no more pages, go to the first field on the first page of the document.</li> <li>The last form field on the last page must indicate, "This is the last form field in this document, next tab will take you to the top of the first page in this document."</li> <li>When tables are used to collect input, first column with row header must be tab-navigable</li> </ul>
<b>Hierarchical Structure</b>	<ul style="list-style-type: none"> <li>Content is structurally (semantically) tagged.</li> </ul>
<b>Data Table Structure</b>	<ul style="list-style-type: none"> <li>Tables are tagged structurally to provide information about their contents.</li> </ul>
<b>Predictable Context</b>	<ul style="list-style-type: none"> <li>Changes of context are initiated only by user action.</li> </ul>
<b>Consistency</b>	<ul style="list-style-type: none"> <li>Placement and functionality of content is predictable and consistent.</li> </ul>
<b>Robust</b>	<b>Content is designed to be compatible with current and future technologies.</b>
<b>Compatibility</b>	<ul style="list-style-type: none"> <li>Published specifications and accessibility conventions are followed.</li> </ul>

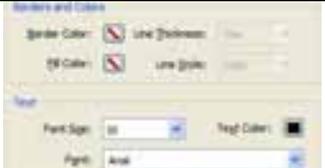
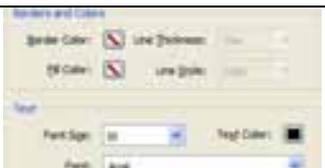


## SSA Desk Guide to Accessible PDF Fillable Forms

Feature	Description
<b>Perceivable</b>	<b>Content is designed to allow presentation according to the user's needs and preferences.</b>
<b>Text</b>	<ul style="list-style-type: none"> <li>• Text can be accurately and programmatically determined.</li> <li>• Text must not extend past visible limit of input box.</li> <li>• All abbreviations and acronyms must be spelled out for the initial occurrence in a form.</li> <li>• Multi-line fields must indicate line number (for example: Line one of three, line two of three, and line three of three.) In addition, multi-line fields must provide question at the beginning and end lines.</li> <li>• Alt text is provided for meaningful non-text content.</li> </ul>
<b>Multimedia Alternatives</b>	<ul style="list-style-type: none"> <li>• Alternative content is provided for meaningful multimedia.</li> </ul>
<b>Links to Plug-ins</b>	<ul style="list-style-type: none"> <li>• Links are provided to accessible plug-ins if they are necessary to render meaningful content.</li> </ul>
<b>Color &amp; Sufficient Contrast</b>	<ul style="list-style-type: none"> <li>• If color is used, utilize sufficient color and contrast.</li> </ul>
<b>Security Set Appropriately</b>	<ul style="list-style-type: none"> <li>• If security settings are used, screen reader access is not disabled.</li> </ul>
<b>Operable</b>	<b>User interface components in the content are designed to allow interaction according to the user's needs and preferences.</b>
<b>Well-defined Focus</b>	<ul style="list-style-type: none"> <li>• Static text used for instructions or information must have a well defined focus or caret, so that JAWS speaks it, and MAGic focuses on it.</li> <li>• All navigation and input in PDF must be available via JAWS screen reader.</li> </ul>
<b>Keyboard Only Operation</b>	<ul style="list-style-type: none"> <li>• All functionality of the content is designed to be operated through a keyboard interface.</li> <li>• Auto skip for multiple fields (like SSN) should be avoided; they can cause issues for keyboard access.</li> <li>• Avoid OnChange events that refresh the screen because they interfere with keyboard access.</li> </ul>
<b>Navigation &amp; Orientation</b>	<ul style="list-style-type: none"> <li>• Provide mechanisms to help users find content, orient themselves within it, and navigate through it.</li> </ul>
<b>Signatures</b>	<ul style="list-style-type: none"> <li>• Signature fields should be part of the tab order, but should be read-only.</li> </ul>
<b>Special Instructions</b>	<ul style="list-style-type: none"> <li>• Instructions such as "use blue or black ink only" must be retained.</li> </ul>
<b>User-directed Interactivity</b>	<ul style="list-style-type: none"> <li>• Content is designed to allow interactivity according to the user's needs and preferences.</li> </ul>
<b>Functional Tooltips</b>	<ul style="list-style-type: none"> <li>• All directions and cues must be included within the tooltip.</li> </ul>
<b>Check Box Labeling</b>	<ul style="list-style-type: none"> <li>• If check boxes have more than 2 items, the first and last check boxes must speak the question associated with the check box.</li> <li>• Check boxes must speak the appropriate state of the check box when selected or deselected.</li> </ul>
<b>Form Fields</b>	<ul style="list-style-type: none"> <li>• Use appropriate field types to collect user input.</li> </ul>
<b>Multi-copy Fill In</b>	<ul style="list-style-type: none"> <li>• Accessible multi-copy forms should automatically populate identical fields on subsequent copies.</li> </ul>
<b>Error Handling</b>	<ul style="list-style-type: none"> <li>• If error detection/validation is used, error messages shall be provided in text which can be programmatically determined.</li> <li>• If successful form submission depends on correct field input, an error message shall identify the error.</li> <li>• An error message shall describe how to correct input errors, unless doing so would invalidate the security or purpose of the form.</li> <li>• Forms which submit data shall allow users the opportunity to review/correct input prior to submission.</li> </ul>
<b>Accessible Scripting</b>	<ul style="list-style-type: none"> <li>• Scripting is written to avoid problems with screen readers and keyboards.</li> </ul>
<b>Unambiguous</b>	<b>Content is linguistically and semantically explicit, and designed to be understandable by its intended audience.</b>
<b>Defined Language</b>	<ul style="list-style-type: none"> <li>• Content is readable and understandable.</li> </ul>
<b>Tab Order</b>	<ul style="list-style-type: none"> <li>• Tab order must represent the logical navigation of the document.</li> <li>• The last field in a form should be able to tab back to the first field or tab to instructional text.</li> <li>• The last form field on the page must go to the next page, if there are no more pages, go to the first field on the first page of the document.</li> <li>• The last form field on the last page must indicate, "This is the last form field in this document, next tab will take you to the top of the first page in this document."</li> <li>• When tables are used to collect input, first column with row header must be tab-navigable</li> </ul>
<b>Hierarchical Structure</b>	<ul style="list-style-type: none"> <li>• Content is structurally (semantically) tagged.</li> </ul>
<b>Data Table Structure</b>	<ul style="list-style-type: none"> <li>• Tables are tagged structurally to provide information about their contents.</li> </ul>
<b>Predictable Context</b>	<ul style="list-style-type: none"> <li>• Changes of context are initiated only by user action.</li> </ul>
<b>Consistency</b>	<ul style="list-style-type: none"> <li>• Placement and functionality of content is predictable and consistent.</li> </ul>
<b>Robust</b>	<b>Content is designed to be compatible with current and future technologies.</b>
<b>Compatibility</b>	<ul style="list-style-type: none"> <li>• Published specifications and accessibility conventions are followed.</li> </ul>



# Acrobat Field Types and Their Usage

Field Type	Purpose	Positioning	Recommended Properties
<b>Check Box</b>	Present a list of choices from which multiple selections of boolean data (i.e., checked vs. unchecked) may be selected.	<p>Which brochure(s), if any, would you like to receive? (select all that apply)</p> <p><input checked="" type="checkbox"/> Brochure 1  <input type="checkbox"/> Brochure 2  <input checked="" type="checkbox"/> Brochure 3</p> <p>OR</p> <p>Which brochure(s), if any, would you like to receive? (select all that apply)</p> <p><input checked="" type="checkbox"/> Brochure 1  <input checked="" type="checkbox"/> Brochure 2  <input type="checkbox"/> Brochure 3</p>	 <p>It is recommended that a check button style be used for check boxes.</p>
<b>Radio Button</b>	Present a group of choices from which a single, mutually exclusive, selection of boolean data (i.e., checked vs. unchecked) may be selected.	<p>What is your gender? <input type="radio"/> Male  <input checked="" type="radio"/> Female  <input type="radio"/> Prefer not to say</p> <p>OR</p> <p>What is your gender?</p> <p><input type="radio"/> Male  <input type="radio"/> Female  <input checked="" type="radio"/> Prefer not to say</p>	
<b>Combo Box</b>	Present a list of selections in <ul style="list-style-type: none"> <li>a pop-up menu from which a user can</li> <li>select only one item, or</li> <li>enter custom text.</li> </ul>	<p>In which county do you live?</p> <p>County 1  County 2  County 3  County 4</p> <p>OR</p> <p>In which county do you live?</p> <p>County 1  County 2  County 3  County 4</p>	
<b>List Box</b>	Present a list of selections in <ul style="list-style-type: none"> <li>a scrollable window from which a user can</li> <li>select only one item, or</li> <li>select multiple items, but</li> <li>cannot enter custom text.</li> </ul>	<p>Which brochure(s), if any, would you like to receive? (use the Ctrl key to make multiple selections)</p> <p>Brochure 1  Brochure 2  Brochure 3</p> <p>OR</p> <p>Which brochure would you like to receive?</p> <p>Brochure 2  Brochure 1</p>	
<b>Text</b>	Collect text entries. Entries can be <ul style="list-style-type: none"> <li>automatically reformatted</li> <li>used in calculations</li> <li>masked (e.g., passwords)</li> </ul> <p>Read-only text fields can be used for labels and rollover help.</p>	<p>First Name: _____</p> <p>Last Name: _____</p> <p>OR</p> <p>What is your relationship to the applicant?</p> <p>_____</p> <p>(the dashed lines illustrate where the borderless boxes are positioned relative to their labels)</p>	
<b>Signature</b>	Collect an electronic signature.	<p>Signature _____</p> <p>OR</p> <p>Signature _____</p> <p>(the dashed lines illustrate where the borderless boxes are positioned relative to their labels)</p>	 <p>Signature fields should be set to read only. SSA does not accept digital signatures at this time.</p>
<b>Button</b>	Initiate an action, such as: <ul style="list-style-type: none"> <li>Submit data</li> <li>Open a file</li> <li>Perform a calculation</li> <li>Validate input</li> </ul>	<p>Submit</p>	<p>Buttons are not authorized for use in SSA forms at this time.</p>

