background

This article covers using the PdfBookmarkVisitor to easily update bookmarks in a PDF when removing pages.

When you remove pages from a PdfDocument it does not update your bookmarks, this can cause issues with bookmarks pointing to incorrect pages, or worse, an error when a bookmark points to an out of bounds page.

The PdfBookmarkVisitor class is a helper that will call your delegate with each Bookmark in a given BookmarkTree, you can change bookmarks or use it to return a list of Bookmark objects that fulfill a specific criteria. We're going to use it to update page indexes of bookmarks that will be preserved and return a list of bookmarks that point to removed pages so we can prune them.

C# example

```
dfDocument pdoc = new PdfDocument(@"Path\To\document.pdf"); /* Remove some pages */
pdoc.Pages.RemoveAt(1); pdoc.Pages.RemoveAt(100); /* Just keep track of the indexes you
removed */ List<int> removedPages = new List<int>(); removedPages.Add(1);
removedPages.Add(100); /* The PdfBookmarkVisitor is a handy class for going through all the
bookmarks easily. */ /* The delegate decides if the bookmark will be returned in the final
list. */ PdfBookmarkVisitor toBePruned = new PdfBookmarkVisitor(pdoc.BookmarkTree, true,
(mark => { foreach(PdfAction action in mark.ClickAction) //Go through all the click actions
for each bookmark. { if(action is PdfGoToViewAction) //Get the click action, if it's a
PdfGoToViewAction check it. { PdfIndexedPageReference page =
(PdfIndexedPageReference)((PdfGoToViewAction)action).Destination.Page; //Find the referenced
page. if (removedPages.Contains(page.PageIndex)) //If it's a bookmark to a removed page,
return it in the final list. return true; /*This can likely be optimized. But if we aren't
removing it and it's larger than removed pages, decrement the number of pages that got
removed under it. */ page.PageIndex -= removedPages.Count(index => index < page.PageIndex); }
} return false; //Otherwise bookmark isn't being removed, don't include it in our list. }));
foreach (var removedPage in toBePruned) //The list of removed page bookmarks. {
removedPage.ParentList.Remove(removedPage.Bookmark); //Remove them from their ParentList. }
pdoc.Save("test.pdf"); //Save output.
```

VB.NET example

```
im pdoc As New PdfDocument("Path\To\document.pdf") 'Remove some pages pdoc.Pages.RemoveAt(1)
pdoc.Pages.RemoveAt(100) 'Just keep track of the indexes you removed Dim removedPages As New
List(Of Integer)() removedPages.Add(1) removedPages.Add(100) 'The PdfBookmarkVisitor is a
handy class for going through all the bookmarks easily. 'The delegate decides if the bookmark
```

```
will be returned in the final list. Dim toBePruned As IEnumerable(Of PdfBookmarkResult) = New PdfBookmarkVisitor(pdoc.BookmarkTree, True, Function(mark As PdfBookmark) As Boolean For Each action As PdfAction In mark.ClickAction 'Go through all the click actions for each bookmark. If TypeOf action Is PdfGoToViewAction Then 'Get the click action, if it's a PdfGoToViewAction check it. Dim page As PdfIndexedPageReference = DirectCast(DirectCast(action, PdfGoToViewAction).Destination.Page, PdfIndexedPageReference) 'Find the referenced page. If removedPages.Contains(page.PageIndex) Then Return True 'If it's a bookmark to a removed page, return it in the final list. End If 'This can likely be optimized. But if we aren't removing it and it's larger than removed pages, decrement the number of pages that got removed under it. page.PageIndex = page.PageIndex - removedPages.Where(Function(index) index < page.PageIndex).Count End If Next 'Otherwise bookmark isn't being removed, don't include it in our list. Return False End Function) For Each removedPage As PdfBookmarkResult In toBePruned removedPage.ParentList.Remove(removedPage.Bookmark) Next pdoc.Save("test.pdf")
```

Original Article

Q10437 - HOWTO: Updating PDF bookmarks when removing pages using PdfDocument

Last Update

2026-11-04 - TD - (Code Fencing and Headings)

Atalasoft Knowledge Base
https://www.atalasoft.com/kb2/KB/50063/HOWTO-Update-PDF-Bookmarks-When-Remo...