



PDF text search in UWP

This guide explains how to search a PDF for text, how to handle the results, and how the search can be controlled with various options.

First, create a `TextSearcher` object and register a result handler:

```
1 private TextSearcher TextSearcher { get; set; }
2
3 public void MyPageConstructor()
4 {
5     TextSearcher = new TextSearcher();
6
7     pdfView.InitializationCompletedHandler += delegate
8     {
9         // We register the search result handler as soon as the `PdfView` is i
10        TextSearcher.SearchResultHandler += APIOnSearchResultHandler;
11    };
12 }
```



The search result handler will be called on a non-UI thread, which allows you to do any long-running processing if needed. However, take care to dispatch any UI updates on the UI thread as demonstrated in this code, which was taken from the Catalog example provided with the SDK:

```
1 private async void APIOnSearchResultHandler(TextSearcher sender, IPageResu
2 {
3     // Dispatch on the UI thread. Otherwise, updating the `SearchResu
4     await CoreApplication.MainView.CoreWindow.Dispatcher.RunAsync(Cor
5     {
6         foreach (var result in pageResults.Results) SearchResults.Add(new Search
```



```
7     });  
8 }
```

To initiate a search inside a document, create an instance of `Query`. This can be done through one of its static factory methods by passing in text, a regular expression, or one of the search presets included in Nutrient UWP SDK:

```
var searchQuery = Query.FromText("pspdfkit");
```



```
var searchQuery = Query.FromSearchPattern(SearchPattern.CreditCardNumber);
```



Using the `TextSearcher` object, call the static method `SearchDocument`:

```
TextSearcher.SearchDocument(pdfView.Document, searchQuery);
```



The search result handler will be called once for each page searched. If no results are found, an empty results list will be passed to the handler.

Configuration

The `Query` object offers several properties to configure the search.

PROPERTY	DESCRIPTION
<code>CompareOptions</code>	Sets various <u>compare options</u> .
<code>SearchType</code>	Sets the <code>SearchType</code> for the query. The static factory methods will initiate
<code>GenerateTextPreviews</code>	Set to generate a text preview. Default is <code>true</code> .
<code>SearchAnnotations</code>	Set to search the content of annotations.
<code>MaximumSearchResults</code>	Specifies the maximum search results returned.
<code>SearchAllPages</code>	Set to specify whether all pages should be searched.

PROPERTY	DESCRIPTION
<code>ReturnEmptyResults</code>	Set to <code>true</code> to return an empty result set for pages with no matches.
<code>PriorityPages</code>	Search the pages in this list of <code>IRange</code> s first. Each <code>IRange</code> specifies a
<code>PreviewRange</code>	Sets the range of the preview that should be generated.

Was this helpful?

☒ YES

☐ NO

Questions? [Contact us](#)

