

[Authentication](#)[Parameters](#)[Fonts](#)[Page breaks](#)[Header / Footer](#)[Encryption](#)

Documentation

Authentication

Authentication is done by simply passing **apiKey** parameter to the request, using GET or POST method. You will get the **apiKey** parameter to your email after the registration.

Note that your "apiKey" is private information, so don't expose it publicly!

A simple example could be like this:

```
https://api.html2pdf.app/v1/generate?html=https://example.com&apiKey=
```

Parameters

Note that parameters should be URL encoded when sending a GET request.

Parameter	Type	Description	Default
-----------	------	-------------	---------

apiKey	<code>string</code>	Your api key which you will get after registration, and can find in a dashboard	required
callBackUrl	<code>string</code>	Generates document in a background and sends result in your callback url. We will send generated pdf document (encoded with base64) to your provided callback url in this structure: <code>{"document": "..."} </code>	<code>null</code>
state	<code>string</code>	This is custom parameter provided by the user which will be sent to the <code>callBackUrl</code> when document is generated.	<code>null</code>
landscape	<code>boolean</code>	Activates landscape mode	<code>false</code>
format	<code>string</code>	Format of the document, can have one of values: <code>Letter, Legal, Tabloid, Ledger, A0, A1, A2, A3, A4, A5, A6</code>	<code>A4</code>
width	<code>integer</code>	Custom width of the pdf size (use in conjunction with height parameter)	<code>null</code>
height	<code>integer</code>	Custom height of the pdf size (use in conjunction with width parameter)	<code>null</code>
marginTop marginRight marginBottom marginLeft	<code>integer</code>	Empty spaces between the outer and the beginning of the content (integer value which is equal to pixels)	<code>0</code>
filename	<code>string</code>	Filename header will be returned	<code>null</code>
waitFor	<code>integer</code>	Parameter to wait for some time (in seconds) before generate the pdf document. It is useful if you have some javascript code running in the background after page was loaded. Available values from <code>0</code> to <code>10</code> .	<code>0</code>
media	<code>string</code>	Use value to generate pdf using print mode styles, can have one of values:	<code>screen</code>

scale	<i>number</i>	content. Scale range is from <i>0.1</i> to <i>2</i>	<i>1</i>
headerTemplate footerTemplate	<i>string</i>	Parameter to provide header / footer of the page. HTML code should be provided and used in conjunction with <i>marginTop</i> / <i>marginBottom</i> to be visible	<i>null</i>

Fonts

We support fonts listed bellow.

As an alternative way, you can use any font you want using [Google fonts](#) or [Web fonts](#).

Andale Mono	Comic Sans MS	NotoSans	Verdana
Arial	Courier New	NotoSerif	Webdings
Arial Black	Georgia	Tahoma	
Arial Narrow	Impact	Times New Roman	
Arial Unicode	Microsoft Sans Serif	Trebuchet MS	

Page breaks

A HTML code example if you want to force a page break at a certain point in an HTML document:

```
<html>
  <head>
    <meta http-equiv="content-type" content="text/html; charset=
    <title>Paginated HTML</title>
    <style>
      div.page {
```

```
</style>
<head>
<body>
  <div class="page">
    <h1>This is Page 1</h1>
  </div>
  <div class="page">
    <h1>This is Page 2</h1>
  </div>
  <div class="page">
    <h1>This is Page 3</h1>
  </div>
</body>
</html>
```

Header and footer

HTML template for the header and footer could be provided using parameters `headerTemplate` and `footerTemplate`. The value should be valid HTML markup with following classes used to inject printing values into them:

Class name	Description
<code>date</code>	Formatted print date
<code>title</code>	Document title
<code>url</code>	Document location
<code>pageNumber</code>	Current page number
<code>totalPages</code>	Total pages in the document

It's important to use `headerTemplate` in conjunction with a large enough `marginTop` parameter, or `footerTemplate` with `marginBottom` to make them visible.

Template code example

```
Date: <span class="date"></span> |  
Title: <span class="title"></span> |  
URL: <span class="url"></span> |  
Total: <span class="totalPages"></span>  
</div>
```

Adding an image

Unfortunately, there is no possibility to load an external resource in the header or footer.

So only one way to add an image to the header or footer is to add base64 encoded content to the image src attribute.

```
  
<head>  
<style>  
  html {  
    -webkit-print-color-adjust: exact; /* this is important to show  
  }  
  body {  
    font-size: 20px;  
    font-family: 'Arial Narrow', serif;  
  }  
  .footer {  
    padding: 0 12px;  
    width: 100%;
```

```
</head>
<body>
<div class="footer">Page: <span class="pageNumber"></span> of <span
  class="totalPages"></span> | Date: <span class="date"></spa
  src="data:image/png;base64, iVBORw0KGgoAAAA...kJggg=="
  style="width:30px;height:30px;" /> | URL: <span class="url"
</div>
</body>
</html>
```

Encryption

When encrypting a PDF document, you can provide a `userPassword` and an `ownerPassword`. The user password is used to restrict certain operations like viewing, printing, copying, and modifying the PDF. The owner password grants full access to the document, including the ability to change permissions and encryption settings.

We support **AES** encryption with **128** bits key to protect your PDF documents.

To trigger encryption process parameter `userPassword` needs to be set. By default, the value of the `permissions` parameter is set to "print", which means that users who enter the password will only be allowed to print the document."

Parameter	Type	Description	Default
<code>userPassword</code>	<code>string</code>	The password that you want users to use for opening the PDF.	<code>null</code>
<code>ownerPassword</code>	<code>string</code>	The owner password grants the ability to perform all actions and also allows for the modification of permissions. Providing non empty <code>ownerPassword</code> determines that this file has such limitations.	<code>null</code>
<code>permissions</code>	<code>array</code>	Field controlling what actions a user entering <code>userPassword</code> can perform.	<code>["print"]</code>

`copy` - allow content copying and copying for accessibility.
`edit` - allow commenting.
`fillform` - allow filling of form fields.
`extract` - allow content copying for accessibility.
`assemble` - allow assembling the document. rotate, insert, delete pages, bookmarks and thumbnails.
`printbest` - allow high resolution printing when 'print' is allowed.

cURL request example

```
curl --output example.pdf --request POST \  
  --url https://api.html2pdf.app/v1/generate \  
  --header 'Content-Type: application/json' \  
  --data '{  
    "apiKey": "f117a90d3a0864645264f700b79161d5da8cb...",  
    "html": "https://example.com",  
    "userPassword": "user",  
    "ownerPassword": "owner",  
    "permissions": ["print", "modify", "copy"]  
  }'
```