



- [\[\] \[\] \[\] \(Changing Upload file size\)](#)

- [PDF \[\] \[\] \[\] \[\] \[\] \[\]](#)

- [\[\] \[\] \[\] \[\] \[\] \[\] \[\] \[\] \[\] \[\] \[\] \[\]](#)

如何更改上传文件大小 (Changing Upload file size)

本文介绍了如何更改 NginX 的上传文件大小。更改上传文件大小对于上传大文件非常有用。本文还介绍了如何更改 NginX 的超时时间。

本文介绍了如何更改 NginX 的上传文件大小。更改上传文件大小对于上传大文件非常有用。本文还介绍了如何更改 NginX 的超时时间。

- NginX 的配置文件
- php.ini 的配置文件
- BookStack 的配置文件

NginX 的配置文件

nginx.conf 是 NginX 的配置文件 (.conf 文件)。它位于 /etc/nginx/nginx.conf。本文介绍了如何更改 NginX 的上传文件大小。

```
client_max_body_size 1024M; # 更改上传文件大小 !
client_body_timeout 2400s; # 更改超时时间 60s 为 2400s .
```

保存更改后，使用以下命令重新加载 NginX 配置。

```
$ sudo nginx -t
nginx : the configuration file /etc/nginx/nginx.conf syntax is ok
nginx : configuration file /etc/nginx/nginx.conf test is successful

$ sudo service nginx reload
```

nginx -t 命令用于测试 NginX 配置。如果配置正确，输出将显示为成功。如果配置不正确，输出将显示错误信息。使用 service nginx reload 命令重新加载 NginX 配置。

php.ini 的配置文件

find php.ini

```
sudo find / -name "php.ini" # (/) php.ini
/etc/php/8.3/cli/php.ini
/etc/php/8.3/apache2/php.ini
/etc/php/8.3/fpm/php.ini
```

FreeBSD fpm/php.ini

```
# nano Ctrl+w
# On
file_uploads=On

# !
upload_max_filesize=1024M

# POST !
post_max_size=1024M

# .60 1 300 5,1800 30
max_execution_time=300

# !
memory_limit=16384M
```

sudo service --status-all php8.3-fpm

```
sudo service --status-all
sudo service php8.3-fpm restart
```

BookStack

.env (#))

PDF

 **HTML** 

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/pdf.js/2.0.466/pdf.min.js"></script>
<style>
canvas[data-pdfurl] {
  background-color: lightgrey;
  width: 100%;
}
.page-content a {
  color: #39f;
  text-decoration: underline;
}
.pdf-wrapper {
  position: relative;
  height: 80vh;
  width: 100%;
}
.pdf-wrapper .download-link {
  position: absolute;
  top: -2em;
  right: 0;
  z-index: 50;
}
.pdf-wrapper .pdf-scroller {
  height: 100%;
  overflow: auto;
}
</style>
<script type="text/javascript">

// ----- THIS SECTION ADDS A PDF BUTTON TO THE EDITOR TOOLBAR THAT ALLOWS YOU TO EMBED
PDFS

// Use BookStack editor event to add custom "Insert PDF" button into main toolbar
window.addEventListener('editor-tinymce::pre-init', event => {
```

```

const mceConfig = event.detail.config;
mceConfig.toolbar = mceConfig.toolbar.replace('link', 'link insertpdf')
});

// Use BookStack editor event to define the custom "Insert PDF" button.
window.addEventListener('editor-tinymce::setup', event => {
  const editor = event.detail.editor;

  // Add PDF insert button
  editor.ui.registry.addButton('insertpdf', {
    tooltip: 'Insert PDF',
    icon: 'document-properties',
    onAction() {
      editor.windowManager.open({
        title: 'Insert PDF',
        body: {
          type: 'panel',
          items: [
            {type: 'textarea', name: 'pdfurl', label: 'PDF URL'}
          ]
        },
        onSubmit: function(e) {
          // Insert content when the window form is submitted
          editor.insertContent('<p id="bkmrk-%C2%A0%C2%A0">&nbsp;<canvas data-pdfurl="' +
e.getData().pdfurl + '"></canvas>&nbsp;</p>');
          e.close();
        },
        buttons: [
          {
            type: 'submit',
            text: 'Insert PDF'
          }
        ]
      });
    }
  });
});

```

//----- THE CODE BELOW SHALL BE ACTIVE IN VIEWING MODE TO EMBED PDFS

```
var renderPdf=function(canvas) {
  var url = canvas.dataset.pdfurl;
  var pdf = null;
  // wrap canvas in div
  var wrapper = document.createElement('div');
  wrapper.className='pdf-wrapper';
  var scroller = document.createElement('div');
  scroller.className='pdf-scroller';
  wrapper.appendChild(scroller);
  canvas.parentNode.insertBefore(wrapper, canvas.nextSibling);
  scroller.insertBefore(canvas, null);

  var downloadLink = document.createElement('a');
  downloadLink.href = url;
  downloadLink.className="download-link";
  downloadLink.innerText = 'Download PDF now ↓';
  wrapper.appendChild(downloadLink);

  var renderPage = function(page) {
    var scale = 1.5;
    var viewport = page.getViewport(scale);
    // Fetch canvas' 2d context
    var context = canvas.getContext('2d');
    // Set dimensions to Canvas
    canvas.height = viewport.height;
    canvas.width = viewport.width;
    canvas.style.maxWidth='100%';
    // Prepare object needed by render method
    var renderContext = {
      canvasContext: context,
      viewport: viewport
    };
    // Render PDF page
    page.render(renderContext);
    if (currentPage < pdf.numPages) {
      currentPage++;
      var newCanvas = document.createElement('canvas');
      scroller.insertBefore(newCanvas, canvas.nextSibling);
      scroller.insertBefore(document.createElement('hr'), canvas.nextSibling);
      canvas=newCanvas;
    }
  }
}
```

```

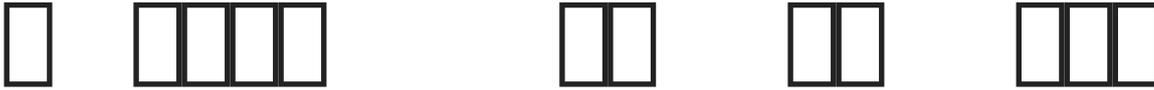
    pdf.getPage(currentPage).then(renderPage);
  }
};
var currentPage = 1;
pdfjsLib.getDocument(url)
.then(function(pdfLocal) {
  pdf = pdfLocal;
  return pdf.getPage(1);
})
.then(renderPage);
};

window.addEventListener('DOMContentLoaded', function() {
  Array.prototype.forEach.call(document.querySelectorAll('canvas[data-pdfurl]'), renderPdf);
});
</script>

```

PDF.js 是一個開源專案，其代碼在 GitHub 上可以公開查看。

1. 在 HTML 文件中，使用 `pdfjsLib` 來初始化 PDF.js。
2. 使用 `pdfjsLib.getDocument(url)` 來獲取 PDF 文檔。
3. 使用 `pdf.getPage(1)` 來獲取 PDF 文檔的第一頁。
4. 在 HTML 文件中，使用 `pdf.getPage(1)` 來獲取 PDF 文檔的第一頁，並將其渲染到 `canvas` 元素中。
5. 使用 `pdf.getPage(1)` 來獲取 PDF 文檔的第一頁，並將其渲染到 `canvas` 元素中。



HTML

```
<!-- Google tag (gtag.js) -->  
<script async src="https://www.googletagmanager.com/gtag/js?id=TAG_ID"></script>  
<script>  
  window.dataLayer = window.dataLayer || [];  
  function gtag(){dataLayer.push(arguments);}  
  gtag('js', new Date());  
  
  gtag('config', 'TAG_ID');  
</script>
```

TAG-ID

24