

# Package: pdftables (via r-universe)

November 3, 2024

**Type** Package

**Title** Programmatic Conversion of PDF Tables

**Version** 0.1

**Date** 2016-02-15

**Description** Allows the user to convert PDF tables to formats more amenable to analysis ('.csv', '.xml', or '.xlsx') by wrapping the PDFTables API. In order to use the package, the user needs to sign up for an API account on the PDFTables website (<<https://pdftables.com/pdf-to-excel-api>>). The package works by taking a PDF file as input, uploading it to PDFTables, and returning a file with the extracted data.

**License** CC0

**URL** <https://www.github.com/expersso/pdftables> , <https://pdftables.com>

**BugReports** <https://www.github.com/expersso/pdftables/issues>

**Imports** httr, tools

**VignetteBuilder** knitr

**Suggests** knitr, rmarkdown

**RoxygenNote** 5.0.1

**Repository** <https://expersso.r-universe.dev>

**RemoteUrl** <https://github.com/expersso/pdftables>

**RemoteRef** HEAD

**RemoteSha** cddb18a7c991f040b3b9fc0934bc14ec90b2e33d

## Contents

convert_pdf . . . . .	2
get_remaining . . . . .	3
<b>Index</b>	<b>4</b>

---

`convert_pdf`*Convert PDF Tables to format more amenable to analysis*

---

**Description**

Convert PDF Tables to format more amenable to analysis

**Usage**

```
convert_pdf(input_file, output_file = NULL, format = "csv",
            message = TRUE, api_key = Sys.getenv("pdftable_api"))
```

**Arguments**

<code>input_file</code>	The PDF file to be converted. If this is a url to a PDF rather than a file path then it is downloaded to a temporary file via <code>download.file</code> before being sent to pdftables for conversion. If <code>input_file</code> is a url then <code>output_file</code> must not be NULL.
<code>output_file</code>	The desired name for the output file
<code>format</code>	One of 'csv', 'xml', 'xlsx-single', 'xlsx-multiple'
<code>message</code>	If TRUE, outputs a message that conversion was successful
<code>api_key</code>	Your API key (from <a href="https://pdftables.com">https://pdftables.com</a> )

**Value**

Creates an output file with the converted PDF table

**Examples**

```
## Not run:
write.csv(head(iris), file = "test.csv", row.names = FALSE)

# Open test.csv and print as PDF to "test.pdf"
convert_pdf("test.pdf", "test2.csv")

## End(Not run)
```

---

<code>get_remaining</code>	<i>Retrieve the number of pages left on your account</i>
----------------------------	--

---

**Description**

Retrieve the number of pages left on your account

**Usage**

```
get_remaining(api_key = Sys.getenv("pdftable_api"))
```

**Arguments**

`api_key` Your API key (from <https://pdftables.com>)

**Value**

A numeric vector of length 1

**Examples**

```
## Not run: get_remaining()
```

# Index

`convert_pdf`, [2](#)

`get_remaining`, [3](#)