

# BLOCKPDF

## DOCUMENTATION



VERSION  
2024.04.04.1908

BLOCKPDF.DE

COPYRIGHT ©2024 TADELSUCHT UG (LIMITED LIABILITY)

<b>1 Introduction</b>	<b>7</b>
1.1 Functional overview . . . . .	7
1.2 Installation . . . . .	9
1.2.1 Windows . . . . .	9
1.2.1.1 Installer . . . . .	9
1.2.1.2 ZIP file . . . . .	11
1.2.2 MacOS . . . . .	11
1.2.2.1 .app application . . . . .	11
1.2.2.2 ZIP file . . . . .	11
1.2.3 Linux . . . . .	12
1.2.3.1 ZIP file . . . . .	12
1.3 License . . . . .	13
1.3.1 Activation of the license in the software . . . . .	13
1.3.2 Automated license distribution . . . . .	13
<b>2 Technical information</b>	<b>14</b>
2.1 Blocks . . . . .	14
2.2 Data sources . . . . .	15
2.2.1 Constant text . . . . .	15
2.2.2 Date and time . . . . .	15
2.2.3 Excel tables / CSV tables . . . . .	15
2.2.4 JSON . . . . .	16
2.2.5 Microsoft Access . . . . .	17
2.2.6 Microsoft SQL . . . . .	18
2.2.7 MySQL/MariaDB . . . . .	18
2.2.8 Embedded text file . . . . .	19
2.2.9 XML . . . . .	19
2.3 Workflows . . . . .	20
2.3.1 Manipulation nodes . . . . .	21
2.3.1.1 Merge . . . . .	21
2.3.1.2 Replace value . . . . .	21
2.3.2 Output node . . . . .	21
2.4 Data types . . . . .	22
2.4.1 Text . . . . .	22
2.4.2 Integer . . . . .	22
2.4.3 Floating point number . . . . .	22
2.4.4 Boolean value (1 / 0, Enabled / Disabled, ...) . . . . .	22
2.4.4.1 Interpretation as true	22
2.4.4.2 Interpretation as false	23
2.4.5 Color (ARGB) . . . . .	24

2.4.6 File path or binary file (Base64) . . . . .	24
2.4.7 Page Format . . . . .	25
2.4.8 Font weight . . . . .	25
2.4.9 Horizontal alignment . . . . .	25
2.4.10 Vertical alignment . . . . .	25
2.4.11 Image scaling . . . . .	25
2.4.12 Inline Alignment . . . . .	25
2.4.13 Text Alignment . . . . .	25
2.4.14 Direction of the text . . . . .	25
<b>3 Quick Start and Examples</b>	<b>26</b>
3.1 Layout with rows and columns . . . . .	26
3.2 Structuring complex documents . . . . .	26
3.3 Example document: Invoice . . . . .	26
<b>4 User interface in detail (UI)</b>	<b>27</b>
4.1 Home page . . . . .	27
4.1.1 Welcome . . . . .	28
4.1.2 Templates . . . . .	30
4.1.3 License/Update . . . . .	31
4.1.4 Settings . . . . .	34
4.1.4.1 User interface . . . . .	34
4.1.4.2 Document . . . . .	35
4.1.4.3 Data . . . . .	35
4.1.4.4 System . . . . .	35
4.1.4.5 Automatic file backups . . . . .	35
4.1.5 Support window . . . . .	36
4.1.6 Documentation window . . . . .	37
4.2 Document . . . . .	37
4.2.1 Function bar . . . . .	38
4.2.2 Toolbox . . . . .	38
4.2.3 Document blocks (block hierarchy) . . . . .	38
4.2.4 Block properties . . . . .	39
4.2.5 Preview the document . . . . .	41
4.2.6 Block context menu (right click) . . . . .	41
4.3 File names . . . . .	43
4.3.1 Filename toolbox . . . . .	43
4.3.1.1 Data source line number . . . . .	43
4.3.1.2 Constant text . . . . .	43
4.3.1.3 Date/Time . . . . .	43

4.3.1.4 Dynamic data . . . . .	43
4.3.2 Components of the file name . . . . .	44
4.3.3 List of file names . . . . .	44
4.3.3.1 Automatic update . . . . .	44
4.3.3.2 Check for errors . . . . .	44
4.4 Data overview . . . . .	45
4.4.1 Blocks with dynamic data . . . . .	45
4.4.2 Workflows for dynamic data . . . . .	45
4.5 Simple workflow editor . . . . .	46
4.5.1 Data source . . . . .	46
4.5.2 Data source configuration/options . . . . .	47
4.5.3 Data . . . . .	47
4.6 Advanced Workflow Editor . . . . .	47
4.6.1 Toolbox . . . . .	48
4.6.2 Workflow . . . . .	48
4.6.3 Options . . . . .	48
4.6.4 Properties of the element . . . . .	49
4.7 Completion . . . . .	50
4.7.1 Document creation . . . . .	50
4.7.2 Preview . . . . .	51
4.7.3 Completed documents . . . . .	51
<b>5 blocks</b>	<b>52</b>
5.1 Relative blocks . . . . .	52
5.1.1 Visual . . . . .	53
5.1.1.1 Text Fragment Container / Text Fragment . . . . .	53
5.1.1.2 Background . . . . .	54
5.1.1.3 Frame . . . . .	55
5.1.1.4 Image . . . . .	55
5.1.1.5 Line horizontally . . . . .	56
5.1.1.6 Line vertical . . . . .	56
5.1.1.7 Text . . . . .	56
5.1.1.8 Placeholders . . . . .	58
5.1.1.9 Current page number . . . . .	58
5.1.1.10 Total number of pages . . . . .	58
5.1.2 Positioning . . . . .	59
5.1.2.1 Alignment . . . . .	59
5.1.2.2 Aspect ratio . . . . .	59
5.1.2.3 Expand horizontally and vertically . . . . .	59
5.1.2.4 Expand horizontally . . . . .	60

5.1.2.5 Expand vertically . . . . .	60
5.1.2.6 Rotate horizontally . . . . .	60
5.1.2.7 Turn over . . . . .	60
5.1.2.8 Rotate vertically . . . . .	60
5.1.2.9 Height . . . . .	60
5.1.2.10 Min/Max height . . . . .	61
5.1.2.11 Min/Max Width . . . . .	61
5.1.2.12 Inner distance . . . . .	61
5.1.2.13 Turn to the left . . . . .	62
5.1.2.14 Turn to the right . . . . .	62
5.1.2.15 Scaling . . . . .	62
5.1.2.16 Scale as needed . . . . .	62
5.1.2.17 Shrinking . . . . .	62
5.1.2.18 Shrink horizontally . . . . .	63
5.1.2.19 Shrink vertically . . . . .	63
5.1.2.20 Shift . . . . .	63
5.1.2.21 Unrestricted . . . . .	63
5.1.2.22 Width . . . . .	63
5.1.3 Content Flow . . . . .	65
5.1.3.1 Secure your place . . . . .	65
5.1.3.2 Page break . . . . .	65
5.1.3.3 Show completely . . . . .	65
5.1.3.4 Show if . . . . .	65
5.1.3.5 Show once . . . . .	66
5.1.3.6 Skip once . . . . .	66
5.1.3.7 Stop page change . . . . .	66
5.1.4 Layout . . . . .	67
5.1.4.1 Page blocks . . . . .	67
5.1.4.2 Line blocks . . . . .	67
5.1.4.3 Column blocks . . . . .	68
5.1.4.4 Inline . . . . .	69
5.1.4.5 Layers . . . . .	69
5.1.5 Other blocks . . . . .	71
5.1.5.1 Content direction from left to right . . . . .	71
5.1.5.2 Content direction from right to left . . . . .	71
5.1.5.3 Debug area . . . . .	71
5.1.5.4 Default text style . . . . .	71
5.1.5.5 Hyperlink . . . . .	73
5.1.5.6 Section . . . . .	73

5.1.5.7 Section link . . . . .	73
5.2 Absolute blocks . . . . .	75
5.2.1 Image (absolute positioning) . . . . .	75
5.2.2 Form fields . . . . .	76
5.2.2.1 Option buttons . . . . .	76
5.2.2.2 Checkbox . . . . .	77
5.2.2.3 Combo box . . . . .	77
5.2.2.4 Date and time . . . . .	78
5.2.2.5 List field . . . . .	78
5.2.2.6 Signature . . . . .	79
5.2.2.7 Text field . . . . .	80
5.2.3 File attachment . . . . .	80
5.3 Barcode field . . . . .	82
5.3.1 Data Matrix . . . . .	82
5.3.2 Pdf417 . . . . .	82
5.3.3 QR code . . . . .	83
5.3.4 Codabar . . . . .	84
5.3.5 Code11 . . . . .	84
5.3.6 Code128, Code128A, Code128B, Code129C, GS1 Code128 . . . . .	85
5.3.7 Code39, Code93 . . . . .	86
5.3.8 Code UPC . . . . .	86
5.3.9 EAN-13, EAN-8 . . . . .	87
5.4 Other blocks . . . . .	88
5.4.1 Comment . . . . .	88
5.4.2 Iterator . . . . .	88
5.4.3 Repeat . . . . .	89
<b>6 command lines (CLI)</b>	<b>90</b>
6.1 Parameter list . . . . .	90
6.2 Console Application Outputs . . . . .	91
<b>7 documents, templates and block structures</b>	<b>93</b>
7.1 BlockPDF document (.BlockPDF) . . . . .	93
7.1.1 Example .BlockPDF file . . . . .	93
7.2 Template files . . . . .	96
7.2.1 Document template (.BlockPDF.zip) . . . . .	96
7.2.2 Prefabricated block structures (.BlockPart) . . . . .	96
7.2.3 Provision of templates and block structures . . . . .	97
<b>8 Local configuration file</b>	<b>99</b>

8.1 Template area . . . . .	100
8.2 UI area . . . . .	101
8.3 Core area . . . . .	102
8.4 AutoSave area . . . . .	103
<b>9 tips and tricks</b>	<b>104</b>
9.1 Import of normal PDF files as templates . . . . .	104
9.2 Relative paths and system variables in documents and document files templates . . . . .	104
9.3 Relative rows . . . . .	105
9.4 Watermarks . . . . .	105
9.5 Connection string . . . . .	105
9.6 Printing documents at the same time . . . . .	106
9.7 Date formatting . . . . .	106
<b>10 Frequently Asked Questions (FAQ)</b>	<b>109</b>
10.1 License . . . . .	109
10.1.1 How many users can use a license at the same time? be applied? . . . . .	109
10.1.2 On how many computers can a license be used?	109
10.1.3 When is a license valid and for how long? . . . . .	110
<b>11 references</b>	<b>111</b>

# Introduction

Welcome to BlockPDF, your solution for the easy and automatic creation of individual PDF documents, without any programming knowledge.

BlockPDF is particularly suitable for users who need larger volumes of tailored documents for their business processes. With our application you can access data from various sources such as Excel files or SQL databases

integrate into your PDF documents. This allows you, for example, to create invoices that obtain specific information for each customer such as invoice number, item and more directly from your databases.

Block PDF offers flexible design of your documents through modular blocks, including

Texts, images, background colors, form fields, signatures, barcodes and QR codes. Each element can be customized individually, so you can, for example

change the color of texts or additional content under certain conditions can display.

## 1.1 Function overview

BlockPDF offers a variety of features for automated creation

individual, data-dependent PDF documents. Here are the main functions:

1. User-friendly, automated creation of individual, data-based



pending PDF documents without the need for programming knowledge (no-code).

2. Creation of various types of PDF documents including forms, invoices, offers, contracts and certificates, as well as all other documents that are required in larger quantities for business processes become.
3. Data integration from various sources such as Excel, CSV, XML, JSON files and SQL Server, MS Access, MySQL/MariaDB databases  
Filling the PDF documents.
4. Use of modular elements (blocks) to structure the PDF documents, that react to each other and build on each other to create the content and that Define the layout of the documents.
5. Wide range of blocks for various purposes including individual pages, Texts, images, background colors, form fields, signatures, barcodes and QR codes, each with configurable properties such as font size and color and position.
6. Dynamic adjustment and filling of documents through access to data sources, so that content such as invoice numbers, invoice data, customer addresses, ordered items and prices are automatically retrieved from the data sources obtained and used in the PDF documents.
7. Adjust properties of a block based on data sources, what means that not only text, but also every property of a block can be dynamically adjusted, for example by changing the color certain parts of text or the display of additional texts depending on specific conditions.
8. Dynamic naming of PDF files based on the data from the data sources, providing individual and contextual naming for each created PDF document possible.

## 1.2 Installation

The application was designed from the start as a cross-platform solution with the aim of creating a unified basis for all supported platforms.

fen. This approach ensures that the application works on all platforms

In terms of appearance and functionality it is largely identical, apart from some platform-specific adjustments.

The biggest differences between the platforms can be seen in the installation of the application. Below you will find instructions for installation and

Execution of the software on different operating systems and information on which aspects need to be paid particular attention to.

### 1.2.1 Windows

Although the program was fundamentally developed to be cross-platform

the Windows platform (Windows 10 and Windows 11) is the best tested

Version, as this is also where the application is mainly developed.

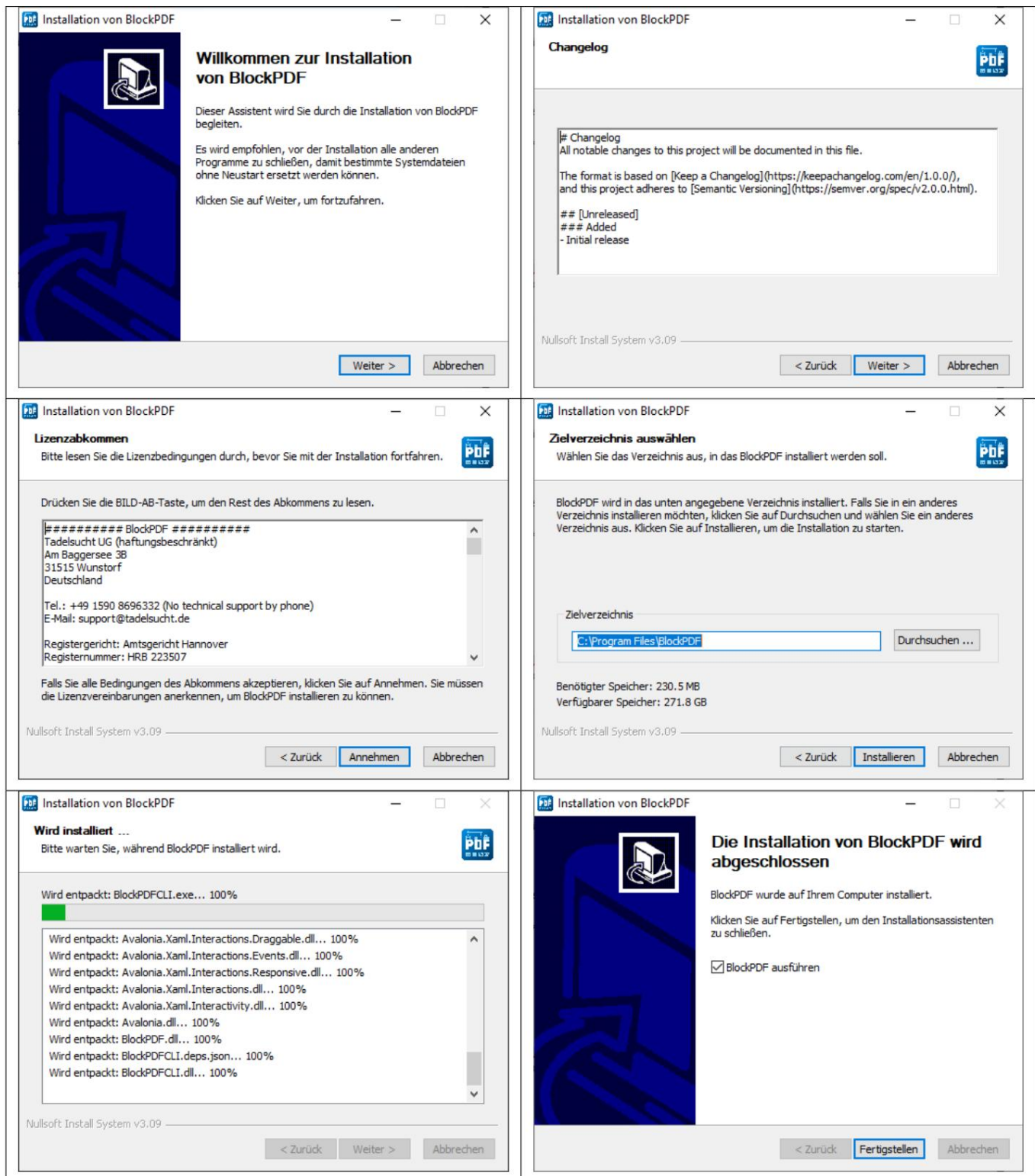
**1.2.1.1 Installation program** The installation program is available in two versions. The

first variant is "BlockPDF-Setup.exe", which tries to gain admin rights

for the installation to get the application for all users of the computer under the path [C:\Program Files\BlockPDF](#) to provide. The second

Variant is "BlockPDF-User-Setup.exe", which does not require admin rights and into the user folder ([C:\Users\%USERNAME%\AppData\Roaming\BlockPDF](#)) in- is installed.

The following necessary steps for installation as screenshots:



The installer can also be used via the console. An installation can be carried out unattended using the "/S" parameter.

If you want to set the installation path, this can be done using the /D=C:\NewInstallDir\ parameter.

**Example 1:** BlockPDF-Setup.exe /S

**Example 2:** BlockPDF-User-Setup.exe /S /D=C:\\NewInstallDir\\

**1.2.1.2 ZIP file** The application in ZIP file format without an installation program can simply be unpacked somewhere. The application is started via the "BlockPDFUI.exe".

### 1.2.2 MacOS

The application is being developed on an older MacBook Pro (Intel) tested with the MacOS version Monterey and a MacBook Air (M2) with the latest available MacOS version.

There are mainly two ways to install BlockPDF on a MacOS system: the .app version and the ZIP file version.

**1.2.2.1 .app application** The .app version of BlockPDF is the easiest method to install and start the application on MacOS. You have to download the .app file from the Internet and move it to your "Applications" folder. Due to MacOS security settings, it is necessary to open the application specifically when starting it for the first time:

1. Navigate to the "Applications" folder in Finder and locate the block PDF.app.
2. Right-click (or Ctrl-click) on the BlockPDF.app and select "Open" from the context menu.
3. A dialog box will appear informing you that the application is off was downloaded from the Internet. Click "Open" to start the application.

This procedure is required to start the application the first time it is started authorize. Once this is done, BlockPDF can be used like any other application be started.

**1.2.2.2 ZIP file** Alternatively, BlockPDF can also be downloaded as a ZIP file and unpacked on the MacOS system. It should be noted that this is the MacOS Sandbox (Gatekeeper) may make the application invisible in one different folder, which is why the update mechanism does not work.

This can be prevented by moving the application once in the "Finder" (see [2]). It may also be the case in the MacOS system settings

BlockPDF must first be activated for execution. The unpacked application can then be started directly via the "BlockPDFUI" file.

### **1.2.3 Linux**

The application is only tested for Ubuntu derivatives during development, which is why an Ubuntu-based Linux system is recommended. For Linux users, BlockPDF is provided as a ZIP file.

#### **1.2.3.1 ZIP file**

1. Download the ZIP file for your processor type from the BlockPDF website down.
2. Unpack the ZIP file into a directory of your choice.
3. Open a terminal window and navigate to the directory where you unpacked the application.
4. Make the BlockPDFUI file executable if it is not already executable with the command: `chmod +x BlockPDFUI`.
5. Start the application by typing `./BlockPDFUI` in the terminal.

It is recommended to unpack the application into a directory to which the user has write access to ensure that all functions of the application work correctly. In particular, the automatic update feature requires write access to the installation directory.

## 1.3 License

Licenses are available as license keys via the website <https://BlockPDF.de/> available. Different license packages are offered there, including single user licenses as well as multi-user or terminal server licenses, each with different durations.

For the following and further questions regarding the license, please take a look at the FAQ in section 10.1:

- How many users can use a license at the same time?  
(See 10.1.1)
- How many computers can a license be used on? (See 10.1.2)
- When is a license valid and for how long? (See 10.1.3)

### 1.3.1 Activation of the license in the software

After you have a license code on <https://BlockPDF.de/> have acquired activate the license in the software. The necessary steps for this are described in section 4.1.3 . During the activation process, a file called LicenseConfirmation is created in the application's working directory.

This file contains the encrypted license code, the expiration date and a Device identification number. For single user licenses, this file is only

ultimately intended for the activated computer and cannot be used on other devices be transmitted. In the case of multi-user or terminal server licenses

The file is not tied to a specific computer and can simply be copied to the working directory of another system for licensing additional computers. The working directory varies depending on the operating system and

Installation and can be viewed in the BlockPDF settings. On Windows it is typically located in the path %AppData%\BlockPDF if the installation was carried out using Setup.exe.

### 1.3.2 Automated license distribution

A multi-user or terminal server license is required for the automated distribution of a license. Single user licenses are not suitable for this.

First, the license key must be entered on a computer in order to Create a LicenseConfirmation file in the working directory.

This file can then be copied and pasted into the working directory of all computers to be licensed.

# Technical information

A detailed overview of the technical aspects of the application is presented below. This introduction aims to provide a deeper understanding of to convey the functionality, configuration and integration of the various components within the application.

## 2.1 Blocks

Blocks are central elements in BlockPDF that enable users to create individual, data-dependent PDF documents in a user-friendly and automated way, without the need for programming knowledge. This

Blocks represent various document elements such as pages, texts, images, Background colors, form fields, signatures, barcodes, and QR codes. They come with a number of customizable properties such as font size, color, and Position that allow users to design their documents as needed and personalize.

The structuring of the document blocks follows a hierarchical structure starts from the document and continues downwards. Within this hierarchy Blocks on the same level are processed from top to bottom. This is especially particularly relevant for the placement of elements such as watermarks, to To be displayed above all other content, be at the end of the block list

should.

Another important aspect is the difference between relative and absolute blocks. Relative blocks dynamically adjust their position and size in relation to other elements in the document. This is helpful for content that needs to flow flexibly within the document layout. Absolute blocks, on the other hand, have one fixed position and size, independent of the other document elements, which is an advantage for static content that needs to be fixed at a specific point in the document.

By dynamically accessing data from sources such as Excel, CSV, XML, JSON files or different database types can be used to fill and customize the blocks with specific information. For example, an invoice document can be created that automatically contains customer-specific data such as invoice number, invoice date, customer address and prices from a data source relates. In addition, the flexibility of the blocks allows each of their properties, from text to coloring, to be adjusted based on the data sources, for example automatically changing the color of a text element certain conditions. These skills close the blocks a powerful tool for creating customized, dynamic PDF documents that effectively meet the requirements of modern business processes satisfy.

## **2.2 Data sources**

To create individual PDF documents, data is required that is dynamic can be loaded for each PDF. BlockPDF supports various options for this Data sources and file formats to be able to obtain data for the PDF files. These data sources are discussed below.

### **2.2.1 Constant text**

Allows you to set a constant text that is repeated n times. Included You get a table with a single column in which the text is simply in each line is repeated.

### **2.2.2 Date and time**

Dates and times in a self-defined format (based on the current date when the program is used).

### **2.2.3 Excel tables / CSV tables**

The following spreadsheet file formats are supported by the program[4]:



File Type	Container Format	File Format	Excel version(s)
.xlsx	ZIP, CFB+ZIP	OpenXml	2007 and newer
.xlsb	ZIP, CFB	OpenXml	2007 and newer
.xls	CFB	BIFF8	97, 2000, XP, 2003 98, 2001, vX, 2004 (Mac)
.xls	CFB	BIFF5	5.0, 95
.xls	-	BIFF4	4.0
.xls	-	BIFF3	3.0
.xls	-	BIFF2	2.0, 2.2
.csv	-	CSV	(All)

For example, a valid table could look like this:

First name	Last name	address	...
Mueller	Bernd	Far-Far-Away	...
Doe	Max	Mustermann Street...	
Model woman	Marie	Doe Street...	
...	...	...	...

There is also the import of spreadsheet files as a separate data source, in which the behavior of columns and rows is swapped, so that, for example.

The following table can also be used:

First name	Bernd	Max	Mary	...
Last name	Mueller	Mustermann	model woman	...
Address	Far-Far-Away	Doe	Doe Street	Doe Doe Street...
...	...	...	...	...

It is worth noting that the use of functions in the Excel files is supported and provides a useful way to process and prepare data before inserting it into the PDF document.

## 2.2.4 JSON

In order to use the data in a JSON file, a certain format must be used of the data must be given. The following JSON structure would be in its complete state: readable in BlockPDF:

```
1  {
2    "root": {
3      "row": [
4        {
5          "Last_name": "Mueller",
6          "First_name": "Bernd",
7          "Address": "Far-Far-Away",
8          "Checkbox_Value": "Yes",
9          "Radio_button_value": "2",
10         "Combobox_Value": "Cake"
11        },
12        {
13          "last_name": "Doe",
14          "First_name": "Max",
15          "Address": "John Doe Street{\ss}e",
16          "Checkbox_Value": "No",
17          "Radio_button_value": "1",
18          "Combobox_Value": "Test"
19        },
20        {
21          "Last_name": "model woman",
22          "First_name": "Marie",
23          "Address": "John Doe Street{\ss}e",
24          "Checkbox_Value": "Yes",
25          "Radio_button_value": "3",
26          "Combobox_Value": "Cookie"
27        },
28        {
29          "Last_name": "Small",
30          "First_name": "Berthold",
31          "Address": "Small Way",
32          "Checkbox_Value": "",
33          "Radio_button_value": "",
34          "Combobox_Value": ""
35        }
36      ]
37    }
38  }
```

### 2.2.5 Microsoft Access

There are several options for using data from a “Microsoft Access” server

Variants for data acquisition are available.

On the one hand, you can choose the option where you simply create a database

File (.accdb or .mdb) and then all available tables for the respective database are automatically displayed. From these tables you can then one can be selected that acts as a data source.

The second option is the ability to enter an SQL query next to the file, which gives you the data you want. This option allows the full functionality of SQL on the Microsoft Access database for the data source to use within BlockPDF.

In some cases it may be necessary to download and install missing dependencies, such as the Microsoft Access Database Engine Redistributable.

### **2.2.6 Microsoft SQL**

There are several variants available for data acquisition when using data from a Microsoft SQL Server.

On the one hand, you can choose the option where you simply enter a “connection string” and then automatically all available ones for the respective database Tables are displayed. One of these tables can then be selected to act as a data source. See for valid “connection string” examples Please see Section 9.5 .

The second option is the option, next to the “connection string”, plain to enter an SQL query that returns the desired data. This This option allows the full functionality of T-SQL on Microsoft SQL Use server for the data source within BlockPDF.

### **2.2.7 MySQL/MariaDB**

The integration of data from a MySQL or MariaDB server into BlockPDF can be implemented in several ways.

One of the options involves entering a “connection string”. which the available tables of the associated database are automatically listed. A table can then be selected from this list as a data source. For examples of valid “connection string” formats, refer to Please refer to Section 9.5.

Another way is to formulate an SQL query directly in addition to the “connection string” that provides the required data. With this method you can use the full range of SQL functions on MySQL or MariaDB Server can be used to configure the data source within BlockPDF eren.

### **2.2.8 Embedded text file**

Embedded text that is interpreted as a data table. For example, this can be XML, JSON, CSV data, as well as data separated by line breaks.

### **2.2.9 XML**

In order to use the data in an XML file, a certain format must be used.

mat of the data must be given. The following XML structure could be read in its entirety into BlockPDF:

```

1 <root>
2   <row>
3     <Last_name>Mueller</Last_name>
4     <First_name>Bernd</First_name>
5     <Address>Far-Far-Away</Address>
6     <Checkbox_Value>Yes</Checkbox_Value>
7     <Radio_button_value>2</Radio_button_value>
8     <Combobox_Value>Cake</Combobox_Value>
9   </row>
10  <row>
11    <Last_name>John Doe</Last_name>
12    <First_name>Max</First_name>
13    <Address>John Smith Straße</Address>
14    <Checkbox_Value>No</Checkbox_Value>
15    <Radio_button_value>1</Radio_button_value>
16    <Combobox_Value>Test</Combobox_Value>
17  </row>
18  <row>
19    <Last_name>Model Woman</Last_name>
20    <First_name>Marie</First_name>
21    <Address>John Smith Straße</Address>
22    <Checkbox_Value>Yes</Checkbox_Value>
23    <Radio_button_value>3</Radio_button_value>
24    <Combobox_Value>Cookie</Combobox_Value>
25  </row>
26  <row>
27    <Last_name>Small</Last_name>
28    <First_name>Berthold</First_name>
29    <Address>Small Way</Address>
30    <Checkbox_Value></Checkbox_Value>
31    <Radio_button_value></Radio_button_value>
32    <Combobox_Value></Combobox_Value>
33  </row>
34 </root>

```

## 2.3 Workflows

In BlockPDF, data sources are used through workflows that enable systematic and flexible processing of data. These workflows

are divided into two main types: simple and advanced workflows.

Simple workflows are characterized by a direct structure consisting of

a single data source and an output node. They are ideal

for tasks where data is obtained directly from a source and without further

Editing should be integrated into the final document.

Advanced workflows, on the other hand, offer the possibility of carrying out more complex data processing.

They can contain multiple data source nodes and manipulation nodes that allow data to be filtered, sorted, or otherwise

edit in a manner before using them in the final document. This

This type of workflow is ideal for more demanding projects that require data from different sources to be merged and extensively manipulated.

For the functionality of every workflow, it is essential that it has both a  
has an input node that defines the data source and an output node that generates the end product. This structure ensures that data flows systematically through the workflow and ultimately in a usable form.

mat will be output.

### **2.3.1 Manipulation nodes**

The manipulation nodes are between the data sources and the output node located and allow various operations to be carried out on the data before they are used elsewhere in the software.

**2.3.1.1 Merge** This function allows more than one connection to receive and merge the data from different sources. At the

When using a merge node, it is important to note that the execution order of the nodes plays a crucial role. The order in which the data sources are processed is based on how they are displayed in the editor: the top node is processed first, followed by the one below it and so on.

To change the order of the data sources, the position of the nodes along the Y axis can be adjusted in the editor. This change is directly reflected in the underlying data model and determines the order in which the  
Data is merged.

**2.3.1.2 Replace value** Allows a fixed text at any point in to replace the data with another one.

### **2.3.2 Output node**

The exit node is always the last node in a workflow. It may from  
give this node only one and the block only takes one connection from another block.

## 2.4 Data Types

In BlockPDF, data for the block properties is represented in different data types. These data types are discussed below.

### 2.4.1 Text

Text data represents sequential character strings that are used to represent information such as names, addresses or any messages. Example: "Hello World", "1234".

### 2.4.2 Integer

Integers are numeric data without decimal places, used to represent integers, including numbers without a fraction. Example: 42, -3.

### 2.4.3 Floating point number

Floating point numbers are numeric data with decimal places (example: 3.14, - 0.001) that can represent a wider range of values, including very small or large numbers. Depending on the system language, decimal places can be represented by a point or a comma. For layout-specific properties, the default unit for floating point numbers is "point" (see reference [3]).

### 2.4.4 Boolean value (1/0, Enabled/Disabled, ...)

BlockPDF automatically converts various text values into a Boolean value around. A variety of input values are supported to make the conversion as flexible as possible and to accept different user input and data source formats.

#### 2.4.4.1 Interpretation as true

— "x"

- "1"

— "true"

— "on"

— "y"

- “yes” (English)
- “ja” (German, Dutch, Danish, Swedish, Romanian, Norwegian)
- “oui” (French)
- “**Да**” (Russian)
- “**是**” (Chinese)
- “sì” (Italian)
- “sim” (Portuguese)
- “yes” (Estonian)
- “kyllä” (Finnish)
- “j” (Latvian)
- “igen” (Hungarian)
- “da” (Slovene, Czech, Slovak)
- “tak” (Polish)
- “**예**” (Korean)
- “**はい**” (Japanese)
- “iya” (Indonesian)
- “evet” (Turkish)
- “taip” (Lithuanian)

#### **2.4.4.2 Interpretation as false**

- “” Space
- “0”
- “false”
- “off”
- “n”
- “no”



- “no” (German)
- “non” (French)
- “**нет**” (Russian)
- “**不**” (Chinese)
- “não” (Portuguese)
- “ei” (Estonian, Finnish)
- “n” (Latvian)
- “no” (Dutch)
- “nej” (Danish, Swedish)
- “nem” (Hungarian)
- “ne” (Slovene, Czech, Slovak, Lithuanian)
- “never” (Polish)
- “nu” (Romanian)
- “**아니**” (Korean)
- “**いいえ**” (Japanese)
- “tidak” (Indonesian)
- “nei” (Norwegian)
- “hayr” (Turkish)

#### **2.4.5 Color (ARGB)**

Color codes in ARGB format (alpha, red, green, blue) enable the precise specification of colors for graphic elements in documents. Example: #FF5733 for a strong orange, #00FF00 for pure green.

#### **2.4.6 File path or binary file (Base64)**

Allow specifying a file path or binary data in Base64 format the integration of external resources such as images or documents into your PDF. Example: “C:/Documents/Image.jpg” or a Base64 encoded file.

#### **2.4.7 Page Format**

The page format defines the size and orientation of the pages in your document, such as A4 or Letter, which is essential for layout design.

#### **2.4.8 Font weight**

The font weight (e.g. bold, normal) determines the visual weight and emphasis of the text to improve readability and text design.

#### **2.4.9 Horizontal alignment**

Horizontal alignment (Left, Center, Right) controls the placement of elements on the horizontal axis of the page to structure the layout.

#### **2.4.10 Vertical alignment**

The vertical alignment (Top, Middle, Bottom) determines the positioning of Elements along the vertical axis of the page, ensuring a balanced page design.

#### **2.4.11 Image scaling**

Image scaling allows you to adjust the image size to the available one Space for optimal display and integration into the document layout to ensure.

#### **2.4.12 Inline Alignment**

Inline alignment controls the alignment of elements within a text flow, which is important for the microstructuring of text content.

#### **2.4.13 Text Alignment**

Text alignment (e.g. bold, normal) affects the appearance and Structure of text blocks to optimize readability and aesthetics.

#### **2.4.14 Direction of the text**

Text alignment (direction), such as LTR (Left-to-Right) or RTL (Right-to-Left), is crucial for the correct display of texts in different languages.



## Quick start and examples

The basic concepts are presented below in the form of a quick introduction as well as exemplary applications.

### **3.1 Layout with rows and columns**

TO DO

### **3.2 Structuring complex documents**

TO DO

### **3.3 Example document: Invoice**

TO DO



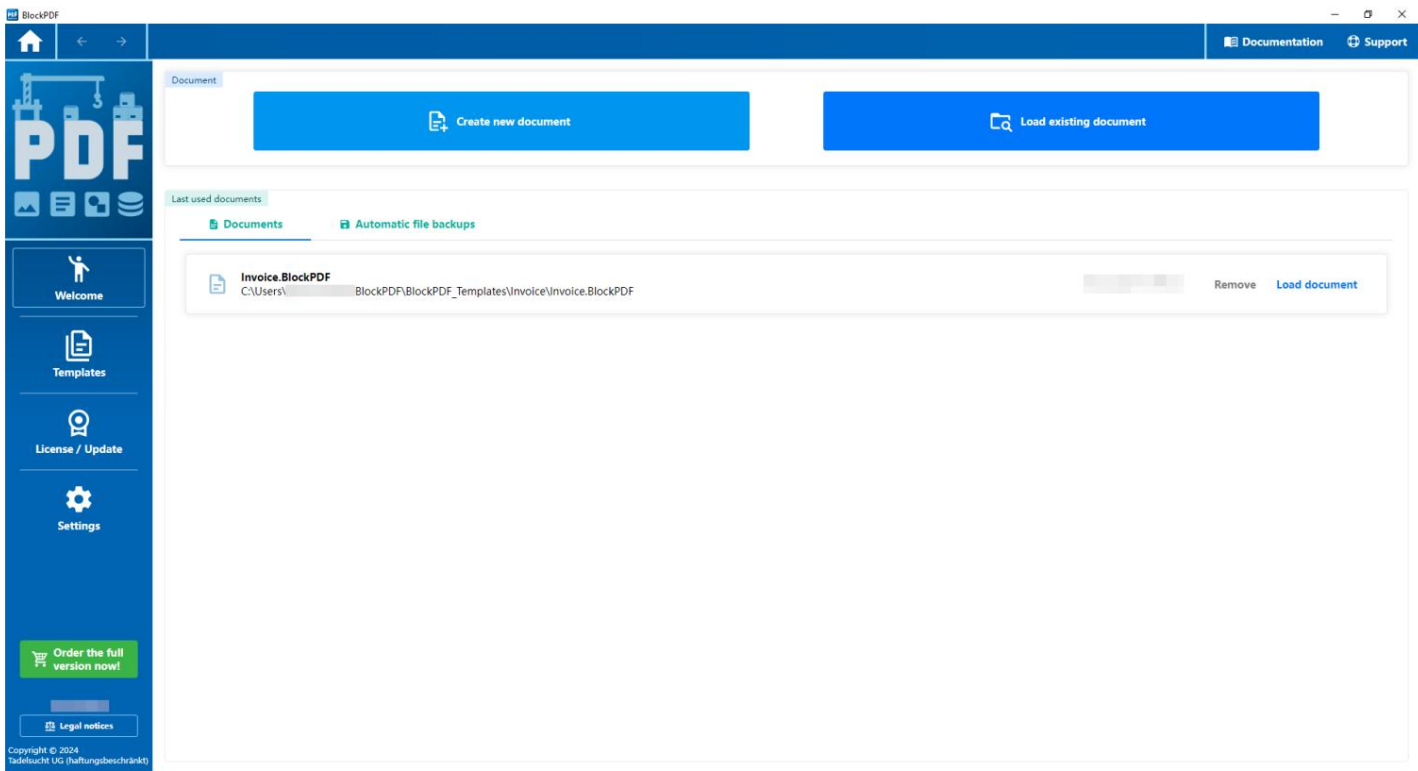
## User interface in Details (UI)

Below we will comprehensively examine all components of the user interface, including additional relevant information about the individual elements.

### **4.1 Home page**

After starting the program, the home page appears, which offers access to a limited selection of subpages, which will be discussed in more detail below. By default, the welcome page is the first page displayed upon startup. There is a house symbol in the top bar that you can use to return to the home page at any time. The meaning of the arrow symbols in the function bar is described in section 4.2.1. In addition, there are buttons to open the documentation window (see 4.1.6) and the support window (see 4.1.5).

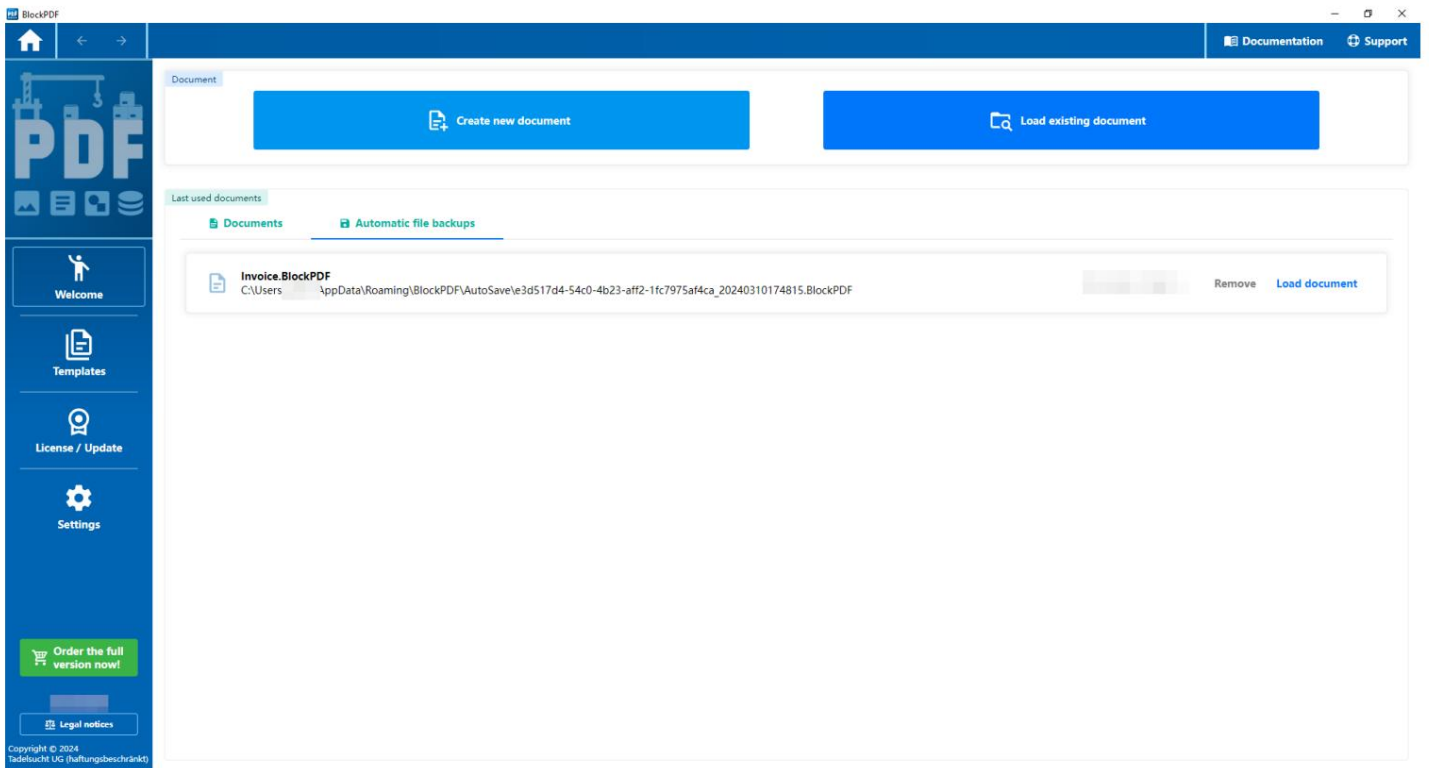
## 4.1.1 Welcome



Home page

The welcome page provides access to the documents area, where users have the option to create a new document or load an existing one.

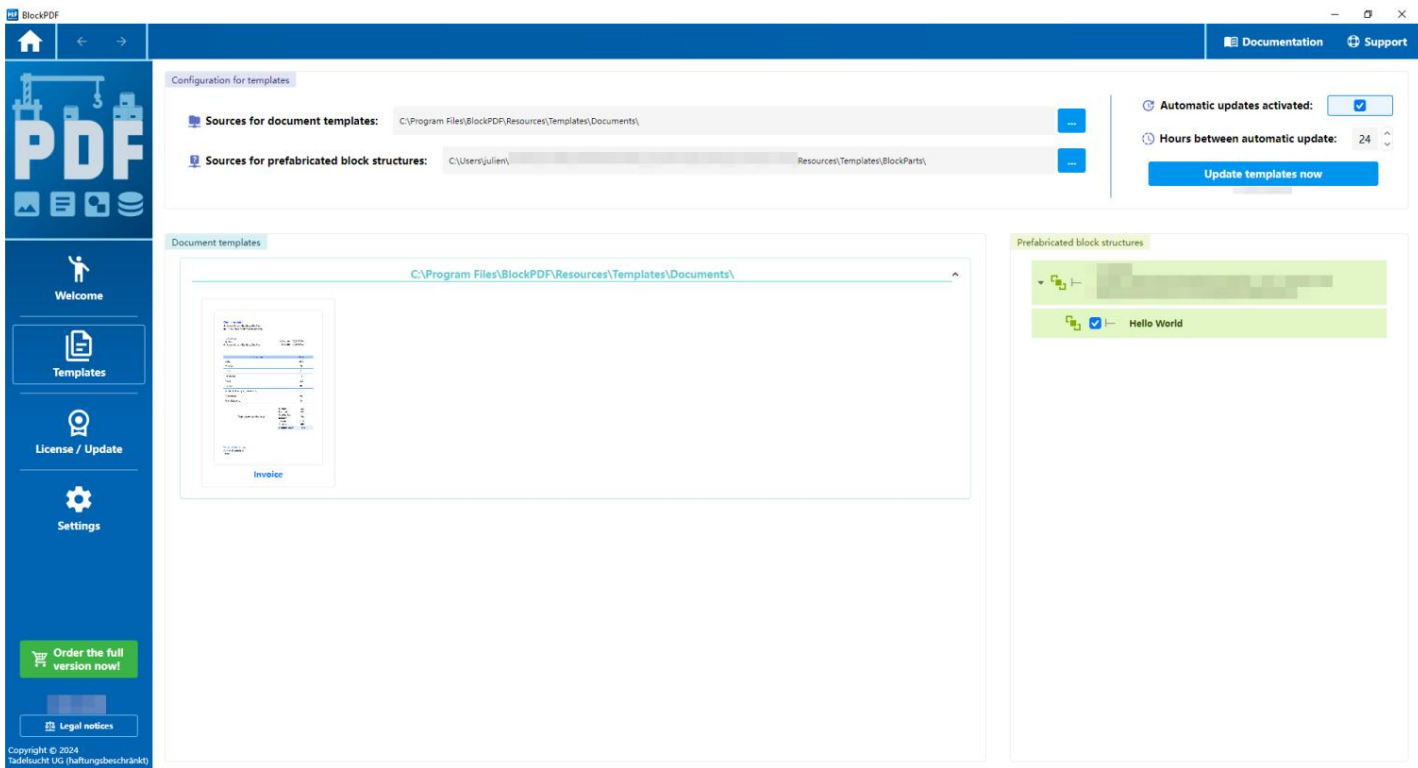
Supported file formats for loading include .BlockPDF, .BlockPDF.zip templates, and PDF files, with PDFs converted accordingly for editing (see [Section 9.1 for more information](#)). Below this area, a list of recently used files will appear, which can either be reloaded or removed from the list.



Home - Automatic file backup selected

The automatic file backup section lists the files that are automatically backed up according to the time intervals defined in the settings became.

## 4.1.2 Templates



### Home - Templates

The template page provides access to templates and block structures from various sources, which can be specified using the text fields above. Further details about the sources and the specific templates are discussed in Section 7.2 . Updating sources can be initiated manually using the Update Templates Now button to ensure the latest content is retrieved. Alternatively, an automatic update takes place at the interval specified in the settings if this option is activated.

Templates and block structures are clearly organized according to their sources and can be collapsed for better clarity. The individual block structures can be activated or deactivated individually for use in the block toolbox when creating documents.

### 4.1.3 License/Update

The screenshot displays the 'License / Update' section of the software interface. On the left, a navigation sidebar includes 'Welcome', 'Templates', 'License / Update' (selected), and 'Settings'. The main content area is divided into two sections:

- Software license packages:** Three options are shown:
  - 1 Year Single user
  - 3 Years Single user
  - Multi-user / Terminal Server Licences
 Each option has a 'Learn more' button. A note below states: '\* Only the current prices shown on the website are binding.'
- Activate system with a license:** This section contains input fields for:
  - Software license key: (placeholder: XXX-XX-XX-XX-N)
  - Remaining activations / already used activations:
  - Expiration date of the software license:
  - Status of the software license:
 An 'Activate license' button is located at the bottom of this section.

Below these sections is the 'Software update' area, which includes:

- Installed program version:** (input field)
- Program version channel:** (dropdown menu, currently set to 'Release')
- Available program version:** (input field)
- Notify when a new version is available:** (checkbox, currently checked)
- View local version changelog** (button)
- Newest view online changelog** (button)
- Update to new version** (button)

At the bottom left, there is a green button 'Order the full version now!' and a 'Legal notices' link. The footer contains copyright information: 'Copyright © 2024 Tadelsucht UG (haftungsbeschränkt)'.

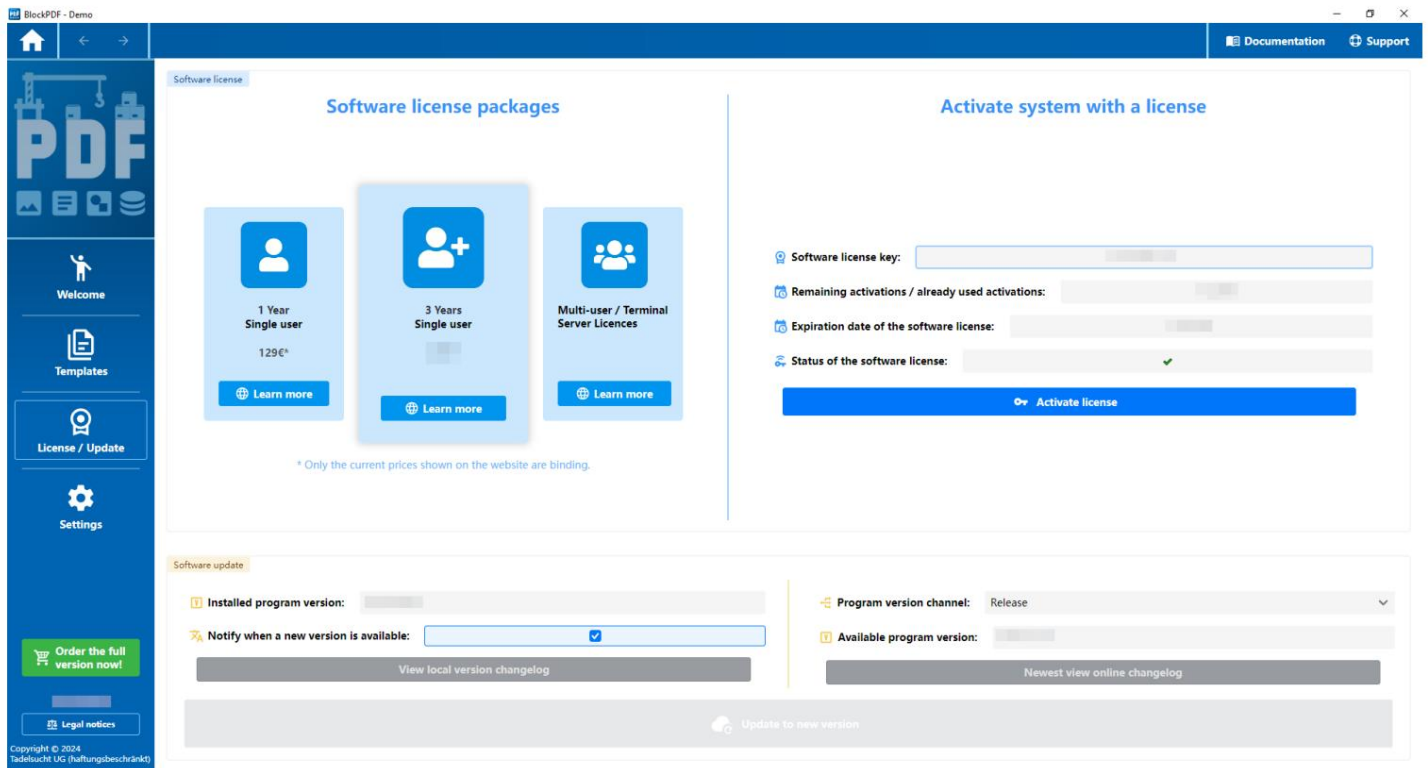
Home - License / Update - No license

This page covers the license for watermark-free use of the software and the update mechanism of the application. In the software update area you can see the currently installed version as well as the version that is currently available on the updated update channel. If there is a new version, a button will appear on the welcome page to update the program. If you

If you don't want this, you can deactivate it in this area. Lastly, that remains Button to update the application now if a new version is available.

For software licensing, there is a small overview of the most common license packages at the top of the page and then the actual elements for administering the software license. If no license has been entered or activated, only the text for entering the license is active.





Home page - License / Update - License code entered

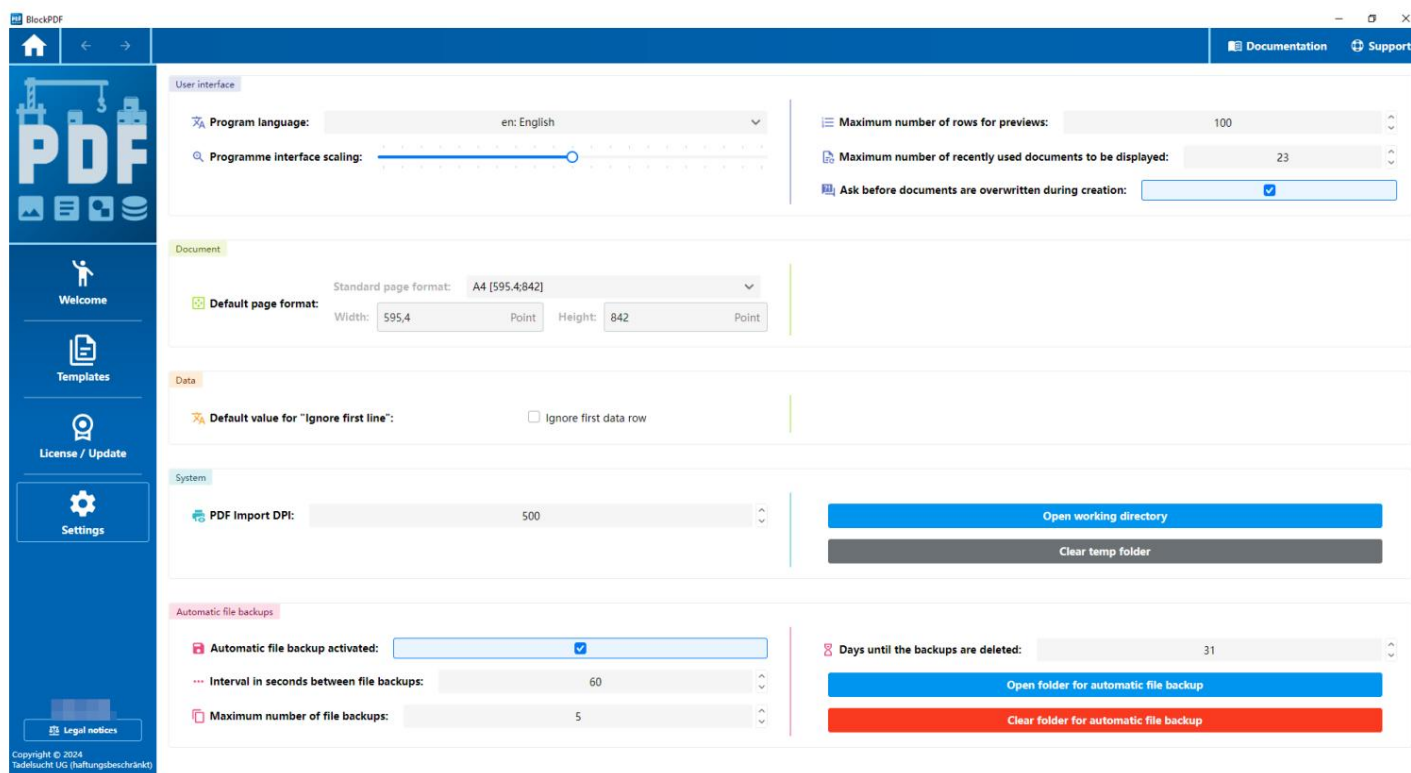
After a license code has been entered, the server will provide information regarding the “remaining activations”, “activations already used”, “expiry date of the software license” and the “status of the software license”. If the license code is valid and there are still enough activations left, the system can be activated using the “Activate license” button.

The screenshot shows the BeckPDF software interface. On the left is a blue sidebar with navigation options: Home, Welcome, Templates, License / Update (highlighted), and Settings. At the bottom of the sidebar is a green button 'Order the full version now!' and a 'Legal notices' link. The main content area is divided into two sections. The top section, 'Software license packages', displays three license options: '1 Year Single user', '3 Years Single user', and 'Multi-user / Terminal Server Licences', each with a 'Learn more' button. A note below states '\* Only the current prices shown on the website are binding.' The right section, 'Activate system with a license', contains fields for 'Software license key', 'Remaining activations / already used activations', 'Expiration date of the software license', and 'Status of the software license' (which shows a green checkmark). A prominent red button labeled 'Remove license' is located below these fields. The bottom section, 'Software update', shows 'Installed program version' and 'Available program version' fields, a 'Program version channel' dropdown set to 'Release', and a checked checkbox for 'Notify when a new version is available'. It also includes buttons for 'View local version changelog' and 'Newest view online changelog', and an 'Update to new version' button.

Home page - License / Update - License activated

After activation, the current information from the server continues to be displayed and there is the option to remove the license from the system. However, if this has not expired, it is not advisable to do so, as no activations will be reactivated. The function is for switching from an old license that has already expired to a new one.

## 4.1.4 Settings



### Home - Settings

The settings are divided into categories, which are discussed below

becomes:

#### 4.1.4.1 User Interface

- **Program language:** The language for the user interface can be set here. be placed.
- **Scaling of the program interface:** Here the scaling of the user interface can be adjusted using a slider.
- **Maximum number of lines for previews:** This determines how many Maximum rows can be displayed in the preview. The default value is 100.
- **Maximum number of recently used documents:** This is where you can determine the maximum number of recently used documents to be displayed should. The default value is 23.
- **Ask before documents are overwritten during creation:** This option is enabled by default and ensures that confirmation is obtained before documents are overwritten.

#### 4.1.4.2 Document

- **Standard page format:** The format for the pages can be selected here. The default format is A4.

#### 4.1.4.3 Data

- **Default value for "Ignore first line":** Here you can set whether the first line in a data file is treated as a header and not used for creating PDF documents.

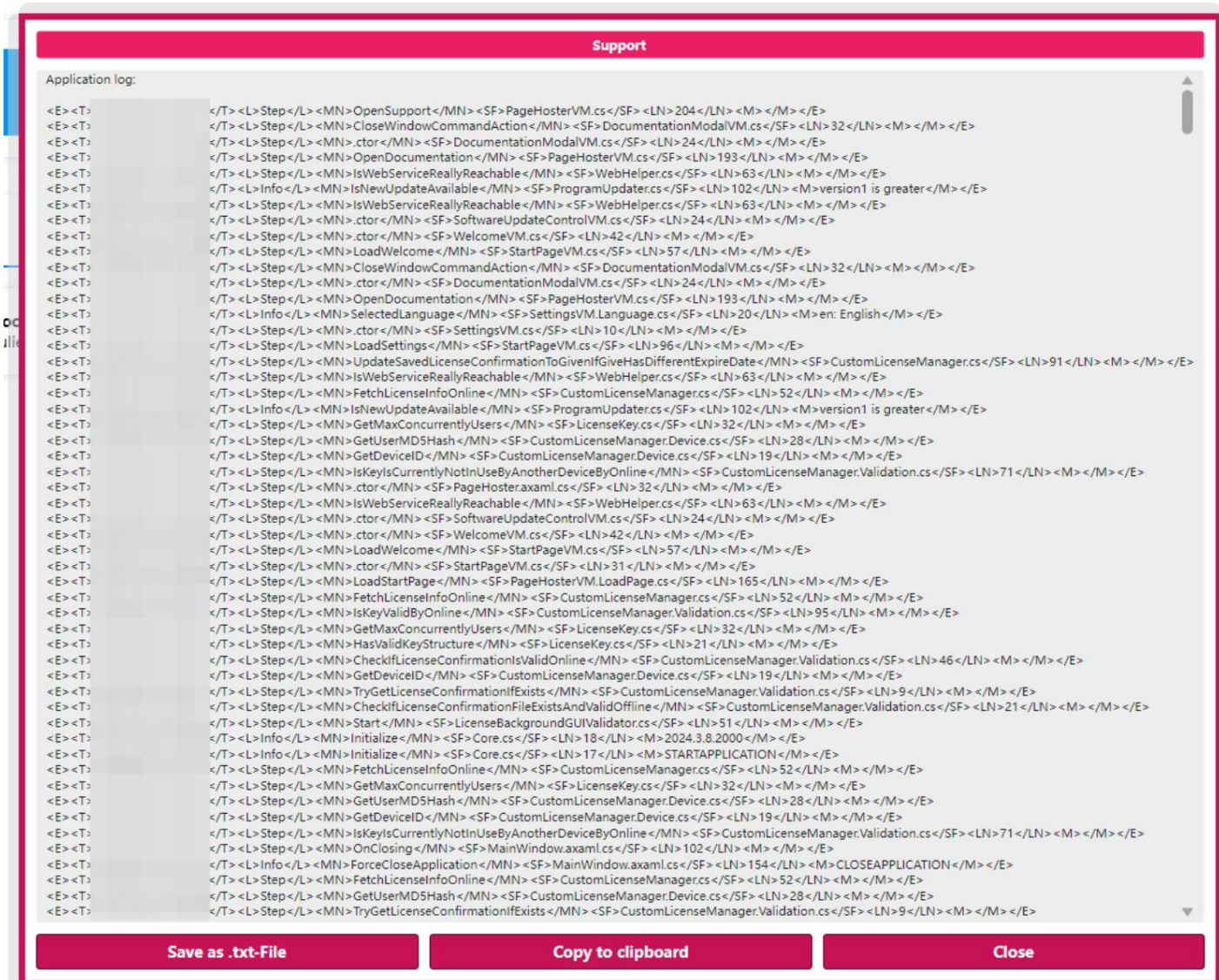
#### 4.1.4.4 System

- **PDF import DPI:** This sets the point density (DPI) at which PDFs are imported. The default value is 500 DPI.
- **Open working directory:** This opens the current working directory open where the project data and settings are saved.
- **Delete Temp folder:** This option can be used to empty the temporary folder in which temporary data is stored.

#### 4.1.4.5 Automatic file backups

- **Automatic file backup enabled:** This option is enabled by default. It activates and ensures the automatic backup of files.
- **Interval in seconds between file backups:** The interval for automatic file backups can be set here. The default value is 60 seconds.
- **Maximum number of file backups:** The maximum number of file backups to be retained can be set here. The default value is 5.
- **Days until backups are deleted:** Shows how many days remain until backups are deleted. In the example it is 31 days.
- **Open automatic file backup folder:** One button to do this. Open the directory where the automatic backups are stored become.
- **Empty automatic file backup folder:** A button to empty the directory containing the automatic backups.

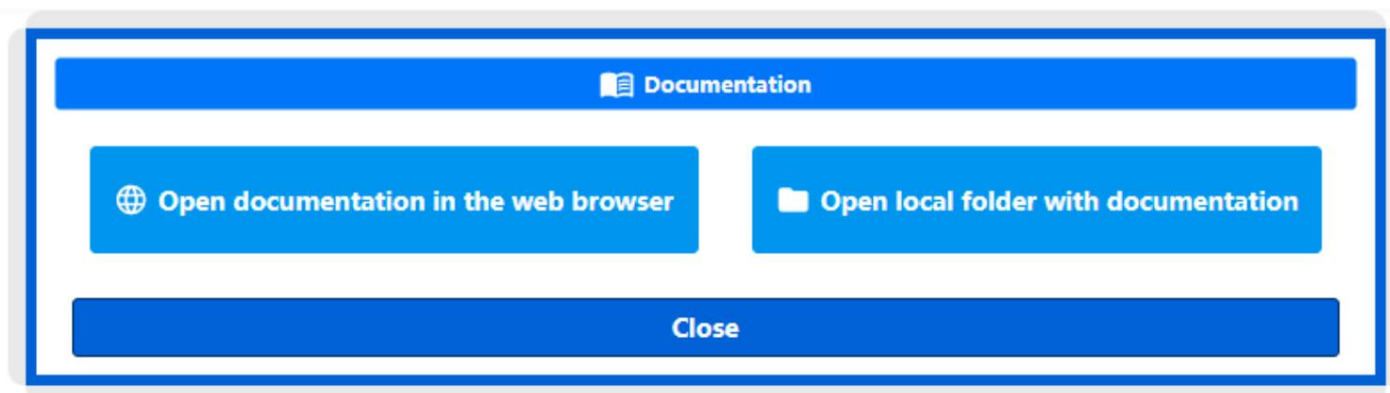
## 4.1.5 Support window



### Support window

The support window shows information that is relevant for an email to the support port in order to speed up the problem resolution or even make it possible in the first place. Included are used program paths and the last n-thousand lines of the program log.

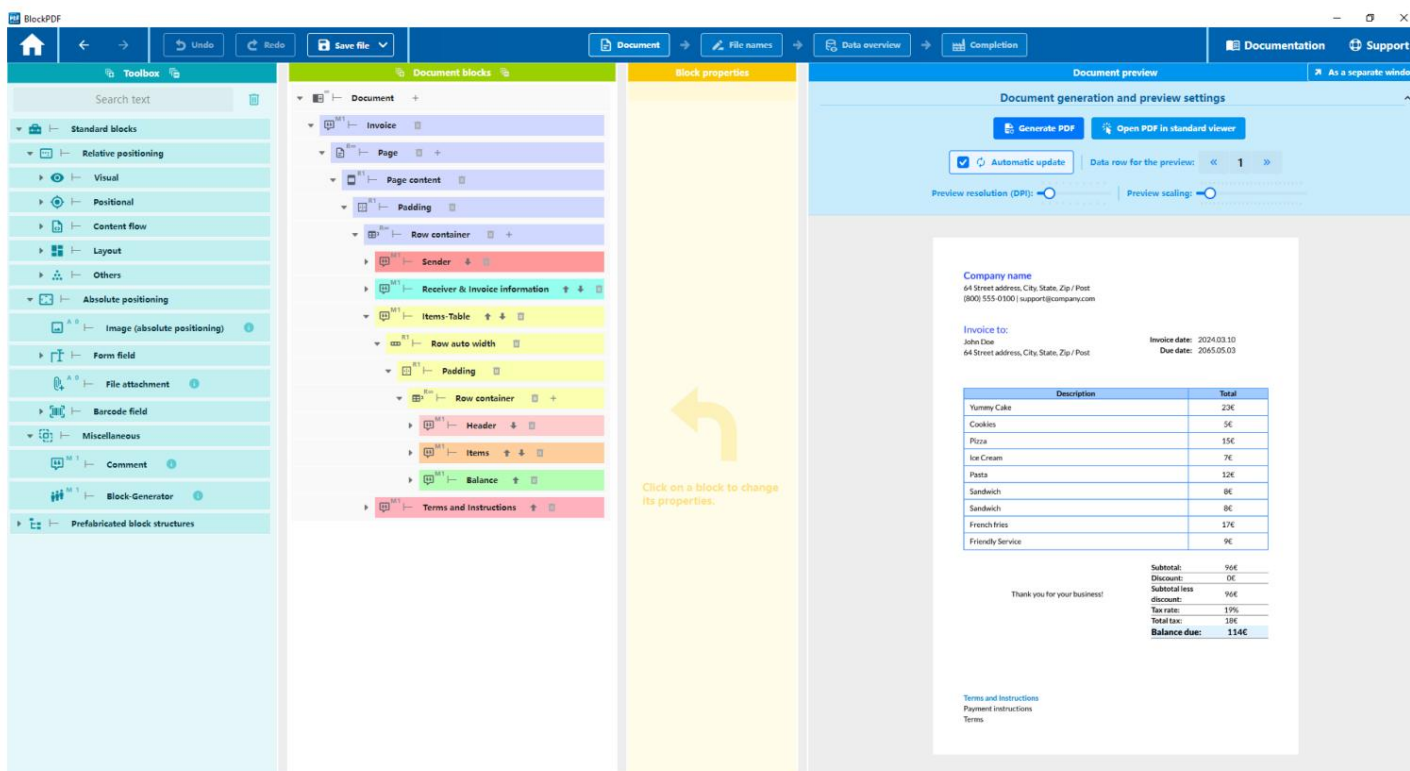
### 4.1.6 Documentation window



Documentation window

The documentation window allows you to view the latest version of the documentation directly. You can choose to view the documentation in the web browser or the local folder with the offline copy of the Documentation.

## 4.2 Document



Document page

The figure presents the application's user interface where the document can be edited. An overview of the function is then given.

sbar at the top of the page and across the different areas of the page given yourself.

#### 4.2.1 Function bar

The function bar initially offers arrow symbols that allow you to quickly go to previous ones Pages can be navigated. There are also rounded arrows that make it possible to undo changes made to the document or to carry out again. The following storage options are available via the following button:

- **Save:** Saves the document to the location where it was last saved. If there is no final storage location, you will be asked where you would like to save the file.
  
- **Save as...:** Opens a dialog for selecting where the document should be saved. should be secured.
  
- **Save as template...:** Opens a dialog to select where the document should go should be saved as a template. The template has a .BlockPDF.zip file extension and contains the document and all files referenced in the document. For example, an Excel file that is used as a data source would be included in the template file.

Finally, the view provides access to additional pages of the application that can be edited step by step to create the documents.

#### 4.2.2 Toolbox

The Toolbox section provides a categorical listing of all available ones Blocks for document design. These versatile blocks can be made using Integrate drag and drop into the block hierarchy of the Document Blocks section. A block is only successfully inserted into the hierarchy if it is compatible with the adjacent blocks.

#### 4.2.3 Document blocks (block hierarchy)

This section visualizes the block hierarchy, which has its starting point in the document block and contains all blocks relevant to the document description. blocks

can be repositioned within this hierarchy using drag-and-drop or via the context menu (right-click). Each entry in the hierarchy has specific controls: If there are multiple blocks on the same level, sorting arrows make it easier to arrange the blocks. A delete function, symbolized by a trash can symbol, allows you to remove one

Blocks individually or including all subordinate blocks. A plus button is used to add new, compatible blocks below the current block.

Status indicators provide an overview of the states of individual blocks: Red

Arrows indicate the path to blocks with errors, indicated by a red flash

Icon highlighted are usually caused by problems with the data

a block property. A yellow warning symbol indicates missing required ones

Properties, while a blue database symbol signals that the

Block dynamic data from a workflow used.

#### 4.2.4 Block properties

The screenshot displays the BlockPDF software interface. On the left is a 'Toolbox' with various block categories like 'Standard blocks', 'Relative positioning', 'Visual', 'Positional', 'Content flow', 'Layout', 'Others', 'Absolute positioning', 'Image (absolute positioning)', 'Form field', 'File attachment', 'Barcode field', 'Miscellaneous', 'Comment', 'Block-Generator', and 'Prefabricated block structures'. The central 'Document blocks' pane shows a hierarchical tree of an invoice document, with the 'Items-Table' block selected. The 'Block properties' pane on the right shows settings for the selected 'Items-Table' block, divided into 'Required' and 'Optional' sections. The 'Required' section includes 'Page format' (A4 [595,4;842]), 'Width' (595,4), and 'Height' (842). The 'Optional' section includes 'Margin left', 'Margin top', 'Margin right', and 'Margin bottom', each with a 'Floating point number with unit' input field. The rightmost pane shows a 'Document preview' of the invoice, including a table of items and a summary table.

Description	Total
Yummy Cake	23€
Cookies	5€
Pizza	15€
Ice Cream	7€
Pasta	12€
Sandwich	8€
All Meat Hamburger (Extra Bacon)	17€
French fries	9€
Friendly Service	0€
<b>Subtotal:</b>	<b>96€</b>
<b>Discount:</b>	<b>0€</b>
<b>Subtotal less discount:</b>	<b>96€</b>
<b>Tax rate:</b>	<b>19%</b>
<b>Total tax:</b>	<b>18€</b>
<b>Balance due:</b>	<b>114€</b>

Document page - block selected

After selecting a block within the document blocks (block hierarchy)

All associated properties of the selected block are displayed in the Block Properties section.

These properties are divided into two main categories: Required and Optional. Required properties must be mandatory



must be defined in order to be able to display the block correctly. Optional properties, on the other hand, do not necessarily require an explicit value assignment, because they either use a default value or none with no value set have an impact. For each property the name, the data type and the currently set value is displayed. If no data is specified, then in the selection bar, the strikethrough page icon is selected. If constant data are selected, "Constant" is selected in the selection and an interface for setting the value is displayed for the data type.

The screenshot displays the BlockPDF software interface. On the left is a 'Toolbox' with various block categories like 'Standard blocks', 'Relative positioning', 'Visual', 'Positional', 'Content flow', 'Layout', 'Others', 'Absolute positioning', 'Image (absolute positioning)', 'Form field', 'File attachment', 'Barcode field', 'Miscellaneous', 'Comment', and 'Block-Generator'. The main workspace shows a document structure with blocks like 'Invoice', 'Page', 'Page content', 'Padding', 'Row container', 'Sender', 'Row auto width', 'Receiver & Invoice information', 'Items-Table', 'Header', 'Items', 'Balance', and 'Terms and Instructions'. A 'Block properties' panel is open for a 'Text' block, showing 'Required' and 'Optional' sections. The 'Required' section has 'Text' selected, with 'Dynamic' chosen over 'Constant'. It includes options for 'Edit data source' (Simple/Advanced), 'Selected workflow' (Customer data), 'Selected column' (H), and a 'Data preview' showing 'John Doe' and 'Hans Haus'. The 'Optional' section includes 'Font Bold' and 'Font Extra Black' with 'Dynamic' selected. On the right, a 'Document preview' window shows an invoice for 'John Doe' with a table of items and a total of 114€.

Description	Total
Yummy Cake	20€
Cookies	5€
Pizza	15€
Ice Cream	7€
Pasta	12€
Sandwich	8€
All-Meat Hamburger (Extra Bacon)	17€
French fries	9€
Friendly Service	0€
<b>Subtotal:</b>	<b>96€</b>
<b>Discount:</b>	<b>0€</b>
<b>Subtotal less discount:</b>	<b>96€</b>
<b>Tax rate:</b>	<b>19%</b>
<b>Total tax:</b>	<b>18€</b>
<b>Balance due:</b>	<b>114€</b>

Document - block with dynamic data selected

When selecting dynamic data from a workflow, the Dynamic option is selected and activated in the selection bar. Buttons are available within this selection

Available to create a new workflow or edit the current workflow. Editing can be done using either the Simple button (see section 4.5 for more information) or Advanced (see

Section 4.6). Below these options is a menu where the workflow to be used can be selected. A button next to it allows changing the workflow name. Below is the selection of data source for the specific property. If the data contains date information, it is possible to convert it into a specific date format using a conversion option. Finally enabled

a fold-out button previews the ones selected from the workflow

Data.

#### 4.2.5 Preview the document

This section allows you to preview the PDF document to be created. Users have the option to generate this preview manually or open directly in the standard PDF viewer. By activating the function Auto Refresh automatically refreshes the preview after any modification to the block hierarchy or block properties. Additionally can the specific data set can be selected for the preview. It also exists the option to adjust the preview resolution (DPI) and its scaling, which can speed up preview loading, especially on slower computers. For an optimized working environment, especially when using If you have multiple screens, the entire preview area can be moved to an independent window using the As a separate window button, which can speed up the document creation process.

#### 4.2.6 Block context menu (right click)

The screenshot displays the BlockPDF - Demo application interface. It is divided into several main sections:

- Toolbox (Left):** A vertical sidebar containing various block types categorized into Standard blocks, Relative positioning, Visual, Positional, Content flow, Layout, Others, Absolute positioning, Image (absolute positioning), Form field, File attachment, Barcode field, and Miscellaneous. Each category lists specific block types like Comment, Iterator, Repeat, and Prefabricated block structures.
- Document blocks (Middle-Left):** A hierarchical tree view of the document structure. The root is 'Document', which contains 'Invoice'. 'Invoice' includes 'Page', 'Page content', 'Padding', 'Row container', 'Sender', 'Row auto width', 'Padding', 'Row container', 'Row auto width', 'Text', 'Row', 'Receiver & Im', 'Items-Table', 'Row auto w', 'Padding', 'Row c', 'Header', 'Header', 'Header', 'Terms and ins'.
- Block properties (Middle-Right):** A panel for editing the selected block's properties. It shows 'Text' properties such as 'Required', 'Text', 'Constant/Dynamic', 'Fixed constant value' (with a text input field containing 'Company name'), 'Font: Bold', 'Font: Extra Black', 'Font: Italic', and 'Font: Strikethrough'. Each font property includes a 'Boolean value (Yes/No, On/Off, 1/0, ...)' and 'Constant/Dynamic' options.
- Document preview (Right):** A window showing the generated PDF document. It includes 'Document generation and preview settings' with buttons for 'Generate PDF' and 'Open PDF in standard viewer'. Below are controls for 'Automatic update', 'Data row for the preview', 'Preview resolution (DPI)', and 'Preview scaling'. The preview itself shows an invoice for 'John Doe' with a table of items and a total of 114€.

Document - context menu of a block (right click)

The figure shows the context menu that can be opened by right-clicking on one Block opens. The following describes the various operations,

which can be carried out via this menu. It is important to note that these operations are only performed if the block hierarchy is still in a valid state afterwards. This is ensured by checking that the blocks in the new structure are compatible with each other everywhere before performing an operation.

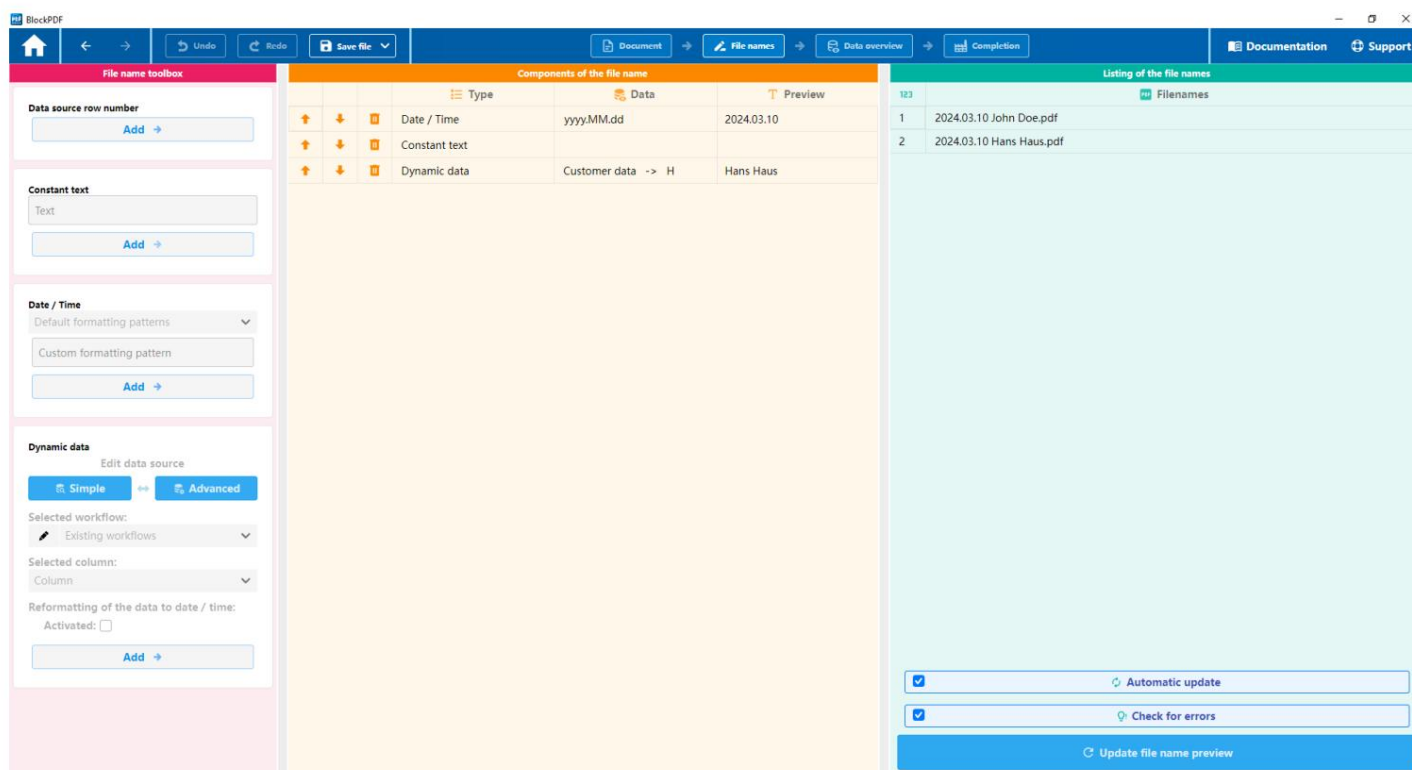
**Single element:**

- **Paste:** Inserts a single block regardless of whether a block with or without sub-elements is in the clipboard.
- **Copy:** Copies only the block on which the right click was executed.
- **Delete:** Deletes only the individual block.
- **Import:** Allows importing a block from a .BlockPDFPart file into the document.
- **Export:** Allows you to export the selected block for use in another document into a .BlockPDFPart file.

**Element with sub-elements:**

- **Insert:** Allows you to insert a block including all of its sub-elements.
- **Copy:** Copies the block including all its sub-elements.
- **Delete:** Removes the block and all its sub-elements from the Document.
- **Import:** Loads a block with its sub-elements from a .BlockPDFPart File in the current document.
- **Export:** Exports the block with all its sub-elements to a .BlockPDFPart File that can then be used in other documents.

## 4.3 File names



Filename pages

The generation of file names for the PDF documents can be configured on this page.

### 4.3.1 Filename toolbox

In the toolbox group there are some elements that can be added as components to the file name.

**4.3.1.1 Data Source Line Number** Adds the current line number of the data as a filename component.

**4.3.1.2 Constant Text** Adds any constant text as a file name component. For example, a space bar (" ") can be added to the file name as a separator.

**4.3.1.3 Date/Time** Inserts the current date in any format added to the file name.

**4.3.1.4 Dynamic Data** Adds dynamic data to the file name.

### 4.3.2 Components of the file name

All components of the file name are displayed here. The type, a representation of the data and a preview of the component are displayed for the individual components. The order is relevant for generating the file name. This can be changed using the arrows on the left. The part of the file name can also be removed using the trash can symbol.

### 4.3.3 List of file names

The preview of all file names is shown here.

**4.3.3.1 Automatic update** The function causes the application to update the preview immediately after a change in the components of the file name.  
tualize.

**4.3.3.2 Check for errors** Our software automatically checks the validity of the file name you enter based on the operating system you are running on.

Here are some guidelines to help you choose a valid filename:

#### **General rules for all operating systems:**

- The file name cannot be empty.
- The maximum length of a file name is 255 characters.

#### **Specific rules for Windows:**

- Avoid using the following characters in the file name: <, >, ;, ", /, \, |, ?, Ÿ.
- The following reserved names may not be used as a complete file name (not even with different upper and lower case letters).  
exercises): CON, PRN, AUX, NUL, COM1 to COM9, LPT1 to LPT9.
- The file name cannot end with a space or a period.

#### **Specific rules for Linux and macOS:**

- The file name cannot contain the characters / or the null character \0.

#### **Additional rules for macOS:**

- The file name should correspond to the normative Unicode representation. This means that special characters or symbols entered in a non-standard form may cause problems. It is recommended that you avoid such characters or check that they are displayed correctly.

## 4.4 Data overview

The screenshot displays the 'Data overview' page in the BlockPDF application. The interface is organized into three main panels:

- Left Panel (Navigation):** A tree view showing the document structure, including 'Document', 'invoice', 'Page', 'Page content', 'Padding', 'Row container', 'Receiver & Invoice information', 'Row auto width', 'Column container', 'Columns (Relative)', 'invoice information', 'Items-Table', 'Items', and 'Iterator'.
- Middle Panel (Blocks with dynamic data):** This panel shows details for three selected blocks:
  - Block 1:** Block property name: Text. Selected workflow: Current date. Selected column: A. Number of rows found: 100.
  - Block 2:** Block property name: Generator data. Selected workflow: Customer data. Selected column: A. Number of rows found: 3.
  - Block 3:** Block property name: Text. Selected workflow: Items. Selected column: A.
- Right Panel (Workflows for dynamic data):** This panel lists three workflows:
  - Workflow: Current date:** Number of uses: 1. Data source: Date and time (yyyy.MM.dd | 100).
  - Workflow: Customer data:** Number of uses: 2. Data source: Excel spreadsheet (C:\Users\... \Invoice\InvoiceData.xlsx).
  - Workflow: Items:** Number of uses: 2. Data source: Data table (table with columns A and B).

Data overview page

This page provides an overall representation of the dynamic data used in the document, associated workflows and the data sources used.

### 4.4.1 Blocks with dynamic data

This section lists the applications of dynamic data in individual blocks on. The hierarchy levels up to the block that integrates the dynamic data are shown on the left. Furthermore, details such as the specific block property, the assigned workflow, the selected data column as well as the data record size shown in the form of the number of lines. Adjustments to the Workflows can be activated using the buttons on the right. Advanced or Advanced.

### 4.4.2 Workflows for dynamic data

In this part all workflows are listed with their names and number their uses and an overview of their data sources (input nodes). In addition, a visualization of the respective settings or the

Data provided by yourself. The buttons on the right allow you to customize the workflows in simple or advanced mode. Should  
 If a workflow does not apply to the document, an option is added to it  
 Deletion is displayed. Above the list there are functions to define  
 new workflows. There is also an option to discard all current workflow data and  
 reload it. This feature comes in handy when  
 For example, changes were made to an Excel file that was being edited in parallel and the updated data  
 should be transferred to BlockPDF.

## 4.5 Simple workflow editor

Simple workflow editor

The simplified workflow editor makes it possible to create one with minimal configuration  
 Create workflow by selecting a single data source. In contrast to more complex  
 workflows, the workflow configured in this way is simply based  
 on two components: an input node, in this case an Excel file,  
 and an output node.

### 4.5.1 Data source

The type of data source for the simple workflow can be selected here.

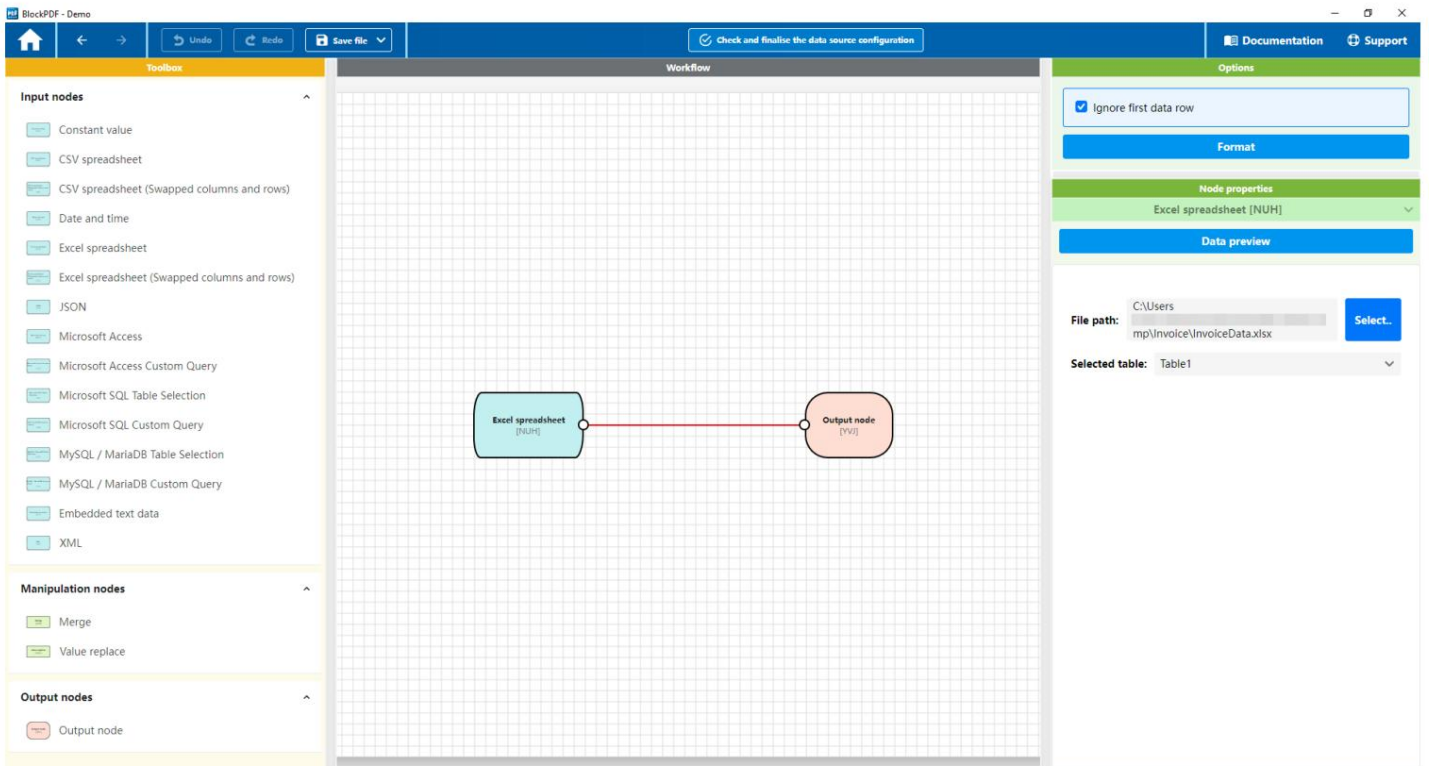
### 4.5.2 Data source configuration / options

This section contains the specific configuration of the selected data source possible. In addition, the options area offers the possibility, regardless of The type of data source determines whether the first line should be ignored as the header. There is also the function to update the data preview manually. alize.

### 4.5.3 Data

This area presents a preview of the data taken from the data source. Not only the data itself, but also details are included the number of usable columns and rows.

## 4.6 Advanced Workflow Editor



Advanced workflow editor

The advanced workflow editor provides a visualization of the workflow that is a represents a closer approximation to the actual programming and provides extensive configuration options. A workflow within

This application can consist of three different types of nodes-set:



- **Input nodes:** These nodes serve as a source of data and can take different forms, for example Excel files, SQL servers etc.
- **Manipulation nodes:** These nodes are responsible for processing the data. They enable actions such as merging data streams or replacing values.
- **Output node:** Only from this node can be used within a workflow one exist. It marks the end of the data flow and the handover the processed data for further use.

The nodes can be connected to each other to determine the data processing path. Data always flows from a starting point of a node recognizable by the anchor point on the right to the entry point of another node, which can be found on the left side.

**Note:** If you cannot select a connection directly, move it. Simply select one of the associated nodes easily. The resulting diagonally running stroke is easier to select.

#### 4.6.1 Toolbox

The toolbox contains all node types that are available for creating a workflow. The nodes can be dragged and dropped into the workflow area to become part of the process.

#### 4.6.2 Workflow

In this central area the current workflow with all its nodes and the connections between them are visualized. The workflow can be edited interactively: nodes and connections can be added, relinked or removed via drag and drop (context menu / right click).

#### 4.6.3 Options

Global settings for the workflow can be made under the options, such as deciding whether to ignore the first record by default. There is also the option to rearrange the workflow to increase clarity.

#### 4.6.4 Properties of the element

The specific properties of each node can be viewed and configured here. The properties of a node become visible when it is selected in the workflow overview or when the mouse is hovered over it for a second. At this point it is also possible to preview the data as it appears up to this point in the workflow.

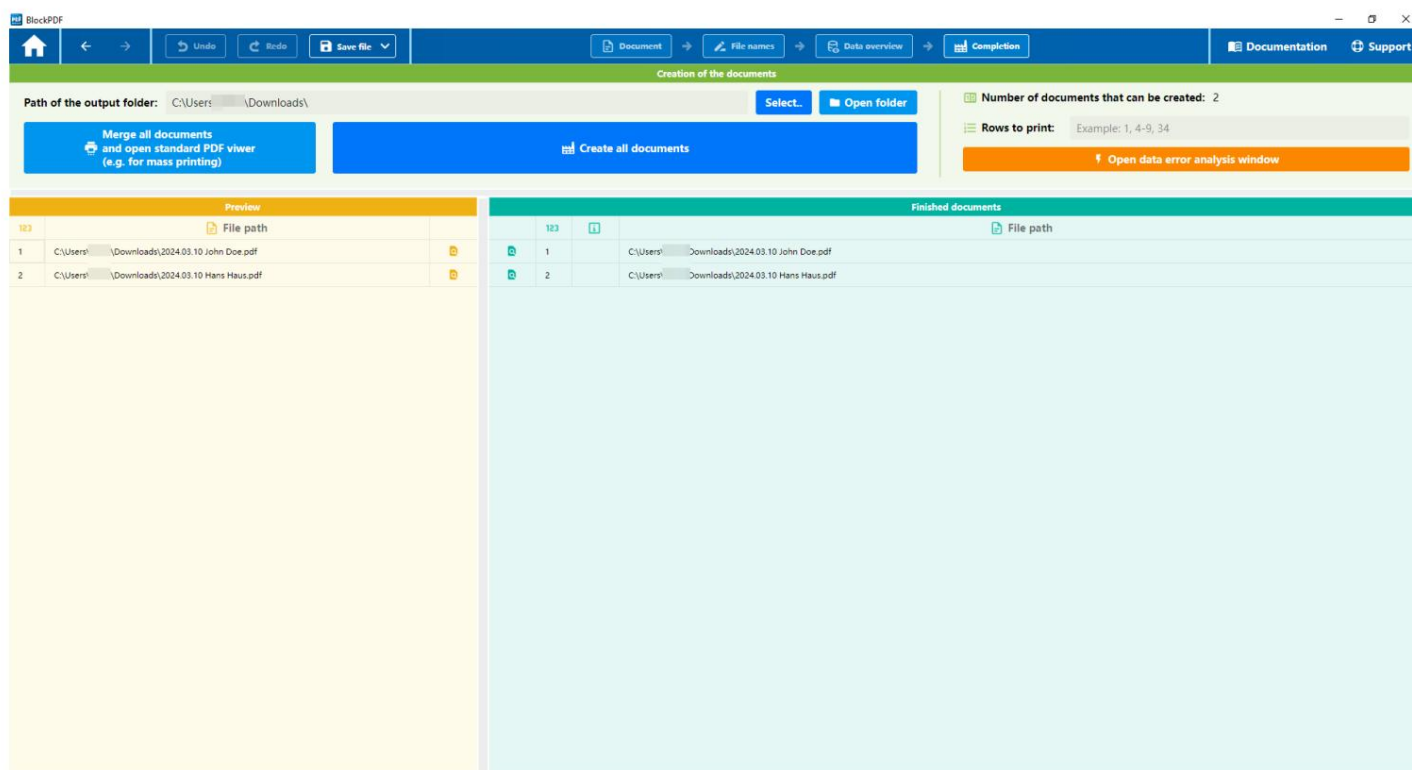
The screenshot shows the 'Data preview' dialog window in the Advanced Workflow Editor. The dialog displays a table with two rows of data. The first row is for 'John Doe' and the second for 'Hans Hase'. The table columns are labeled A through H. The background shows the workflow editor's toolbar, toolbox, and options panel.

	A	B	C	D	E	F	G	H
1	Description;Total Cake;23€ Cookies;5€ Pizza;15€ Ice Cream;7€ Pasta;12€ Sandwich;9€ All-Meat Hamburger [Extra Bacon];17€ French fries;9€ Friendly Service;0€ Description;Total Chocolate cake;4€ Vanilla ice cream with whipped cream and chocolate sauce;5€ Fruit salad with yogurt;3€ Cheesecake with raspberry sorbet;4€ Latte macchiato;3€ Cappuccino;2€ Espresso;2€ Apple spritzer;2€ Mineral water;2€ Bread basket with various types of bread and butter;3€ Olives;2€	96€	0€	96€	19%	18€	114€	John Doe
2	Mixed salad;4€ Soup of the day;5€ Vanilla ice cream with whipped cream and chocolate sauce;5€ Fruit salad with yogurt;3€ Cheesecake with raspberry sorbet;4€ Latte macchiato;3€ Cappuccino;2€ Espresso;2€ Apple spritzer;2€ Mineral water;2€ Bread basket with various types of bread and butter;3€ Olives;2€ Mixed salad;4€ Soup of the day;5€	41€	0€	41€	19%	8€	49€	Hans Hase

Advanced Workflow Editor - Data

The figure shows the dialog window that opens to save the data to look at a specific point in the workflow.

## 4.7 Completion



Completion page

This page is used to create the PDF documents when the rest of the configuration is completed. The following describes the individual parts of the surface received.

### 4.7.1 Document creation

In order to create PDF documents, a target folder must first be defined.

This can be done either by selecting using the "Select..." button

or by entering the path directly in the file path field. Afterwards there are two options: The first, "Merge all documents and display them in the standard PDF viewer (e.g. for mass printing)", created

a collected document, while the second option, "Create all documents", individual PDF documents generated.

Additional information is provided in a separate area. A

Counter, "Number of documents that can be created", informs about the maximum number of documents that can be created based on the existing workflows and data sources can be generated. It should be noted that the workflow with the lowest number of data records sets the upper limit for the total number of PDFs that can be produced. Furthermore, the option "Lines to be printed" allows

a specific selection of which of the potentially created documents should actually be produced.

Lastly, there is a button that opens the "Data Error Analysis Window".

This window lists potential errors due to incorrect or inappropriate data could occur during document creation. A typical example

An error like this would be trying to put a text like "cake" into a field

which expects a numeric value, such as text size. Since the

Text cannot be converted to a number, this would normally be the case

lead to an error. However, with this analysis tool, such problems can be identified and avoided at an early stage.

#### 4.7.2 Preview

The preview function is organized in a table with three columns. The first

Column shows the line number, which reflects the order of the documents. The second column lists the file path where the PDF file should be saved. The third column contains

one for each document

specific button. By pressing this button you can preview the

Create a corresponding PDF document and open it in the standard PDF viewer.

#### 4.7.3 Completed documents

The overview of the completed documents is displayed in a table with four columns shown:

1. **Open PDF:** In the first column you will find an action or a link to to open the corresponding PDF document directly in the standard PDF viewer.  
nen.
2. **Line number:** The second column displays the line number, which allows for unique identification of each document within the list.
3. **Error information:** If the creation of a PDF document is not possible, be successful, detailed error information is provided in the third column provided.
4. **File Path:** The fourth and last column gives the location of the PDF document at.



## blocks

This section introduces the different types of blocks, which can be used to design PDF documents. blocks are in this application the basic building blocks of a PDF document and enable a versatile and precise arrangement of content. They can be divided into two main categories: relative and absolute blocks.

### **5.1 Relative blocks**

Relative blocks are flexible and are based on the structure and dimensions of the document. They dynamically adapt to the surrounding content and are ideal for creating responsive layouts that automatically adapt to different page lengths and widths. Among the relative blocks there are subcategories such as visual elements, positioning, Content flow, layout and other specialized blocks.

### 5.1.1 Visual

In the Visual category you will find blocks that focus on visual design and Focus on displaying content in your PDF document. This includes Blocks for flexible text design, such as 'Text Fragment Container' and 'Text Fragment', which enable individual adaptation of text segments with different styles. In addition, this category includes elements for visual enhancement such as background colors, frames, images and lines that help Make your document attractive and readable. It also offers functions for integrating images with variable scaling for highlighting through horizontal and vertical lines as well as for embedding text with extensive style options. Placeholder blocks support the planning of the layout, by reserving the space for future content. Enabling together These visual blocks provide precise and creative control over the appearance of your document, from the detailed design of individual text fragments to the Overall aesthetics.

**5.1.1.1 Text Fragment Container / Text Fragment** These two blocks, 'Text Fragment Container' and 'Text Fragment', are designed to contain text within your PDF document to be flexible. The 'Text Fragment Container' serves as a container for various 'Text Fragment' elements that allow you to create text segments with individual style settings. This allows you to display complex structured text with different fonts, sizes and colors in a coherent section.

#### **Text Fragment Container Properties (Optional):**

- **Bold:** If selected, the text will appear in bold.
- **Extra Black:** If selected, the text will appear in extra black font strength shown.
- **Italic:** If selected, the text will be italicized.
- **Strikethrough:** If selected, the text will appear strikethrough.
- **Underlined:** If selected, the text will be underlined.
- **Font size:** Defines the size of the font.
- **Font color:** Defines the color of the text.
- **Background Color:** Defines the background color of the text.
- **Font Family:** Defines the font family for the text.
- **Line height:** Defines the line spacing.

- **Letter spacing:** Defines the space between letters.
- **Line Break Everywhere:** When selected, allows line breaks at everyone place in the text.
- **Text alignment:** Defines the alignment of the text.
- **Thin:** If selected, the text will be displayed in thin font weight.
- **Extra Light:** If selected, the text will be in extra light font weight shown.
- **Light:** If selected, the text will be displayed in a light font weight.
- **Normal:** If selected, the text will be displayed at normal font weight.
- **Medium:** If selected, the text will be displayed in medium font weight.
- **Semi-bold:** When selected, the text is displayed in semi-bold.
- **Extra Bold:** If selected, the text will be displayed in extra bold.
- **Black:** When selected, the text is displayed in black font weight.
- **Normal Position:** When selected, the text will be in normal vertical position Position shown.
- **Subscript:** If selected, the text will be displayed in subscript.
- **Superscript:** If selected, the text will be displayed in superscript.

#### **Text fragment properties:**

##### **Necessary:**

- **Text:** The text to display.

##### **Optional:**

The optional properties of the 'Text Fragment' correspond to those of the 'Text Fragment' Fragment containers' and enable individual adjustments for each text segment. ment within the container.

**5.1.1.2 Background** The block is used to add a background color to an area of your PDF document. This can be used for visual design, such as demarcating or highlighting certain areas. the.

#### **Characteristics:**

##### **Necessary**

- **Color:** Sets the background color of the block. The specification is in hexadecimal ARGB format, including transparency (alpha) and the color values for Red, green and blue.

**5.1.1.3 Border** This block is used to create one or more borders to draw an area of your PDF document. Margins can be customized for each side (left, top, right, bottom) can be set, both in width as well as in their color. This is useful for visually separating or highlighting elements.

**Characteristics:**

**Optional**

- **Left Width:** Defines the width of the left margin.
- **Top Width:** Defines the width of the top margin.
- **Right Width:** Defines the width of the right margin.
- **Bottom Width:** Defines the width of the bottom margin.
- **Color:** Sets the color of the border. The information is given in hexadecimal ARGB format, including transparency (alpha) and the color values for Red, green and blue.

**5.1.1.4 Image** This block allows you to insert an image into your PDF document. You can insert the image from a file or data source (Base64 or file path) and adjust its scaling to optimally fit it into your document. The scaling options allow you to scale the image either in width, height or the available area, making it versatile in different layout contexts.

**Characteristics:**

**Necessary**

- **Image file:** The path to the image file or the image data encoded in Base64 that is stored in the PDF should be displayed.

**Optional**

- **Scaling:** Defines how the image in the block should be scaled. Options are 'Adjust Width' (adjusts the width to the block), 'Adjust Height' (adjusts the height to the block) and 'Fit Area' (fits the image proportionally into the available area). If no scaling is specified, Fit Range will be used by default.



**5.1.1.5 Line horizontal** This block allows you to draw a horizontal line in your PDF document. It can be used to create visual separations between different sections or to highlight certain content. The customization options include the thickness of the line as well as its color, which allows for flexible design according to your design ideas.

**Properties:**

**Required**

- **Size:** Defines the thickness of the line.

**Optional**

- **Color:** Determines the color of the line, specified in hexadecimal ARGB format, including transparency (alpha) and the color values for red, green, and Blue.

**5.1.1.6 Line Vertical** This block allows you to draw a vertical line in your PDF document. This feature can be used to differentiate content or to add design elements that draw attention

direct to specific areas. The thickness and color of the line can be customized well, giving you the ability to customize the block to fit the overall design of your document.

**Properties:**

**Required**

- **Size:** Determines the thickness of the line.

**Optional**

- **Color:** Sets the color of the line. The value is specified in hexadecimal ARGB format and includes transparency (alpha) and the color values for red, green, and blue.

**5.1.1.7 Text** This block allows you to insert text into your PDF document and offers a variety of customization options to tailor the text style to your exact needs. You can individually adjust the font size, color, style and many other aspects of the text.

**Characteristics:**

**Necessary**

- **Text:** The text to be displayed in the block. Also supports formatting strings for date and time.

**Optional**

- **Bold:** If selected, the text will appear in bold.
- **Extra Black:** If selected, the text will appear in extra black font strength shown.
- **Italic:** If selected, the text will be italicized.
- **Strikethrough:** If selected, the text will appear strikethrough.
- **Underlined:** If selected, the text will be underlined.
- **Font size:** Defines the size of the font.
- **Font color:** Defines the color of the text.
- **Background Color:** Defines the background color of the text.
- **Font Family:** Defines the font family for the text.
- **Line height:** Defines the line spacing.
- **Letter spacing:** Defines the space between letters.
- **Line break everywhere:** Allows a line break at any point in the Text.
- **Text alignment:** Defines the alignment of the text.
- **Thin:** If selected, the text will be displayed in thin font weight.
- **Extra Light:** If selected, the text will be in extra light font weight shown.
- **Light:** If selected, the text will be displayed in a light font weight.
- **Normal:** If selected, the text will be displayed at normal font weight.
- **Medium:** If selected, the text will be displayed in medium font weight.
- **Semi-bold:** When selected, the text is displayed in semi-bold.
- **Extra Bold:** If selected, the text will be displayed in extra bold.
- **Black:** When selected, the text is displayed in black font weight.
- **Normal Position:** When selected, the text will be in normal vertical position Position shown.
- **Subscript:** If selected, the text will be displayed in subscript.
- **Superscript:** If selected, the text will be displayed in superscript.

**5.1.1.8 Placeholder** This block is used to create a placeholder in your PDF to create document. Wildcards can be used for various purposes for example to reserve space for future content or to help design the layout before the final content is available are. The flexibility to add optional text allows the purpose or to mark the planned use of the placeholder, which is particularly important in the can be helpful in the early stages of document creation.

**Characteristics:**

**Optional**

- **Text:** Allows you to specify a text that will be displayed in the placeholder. This can be used to provide information about the intended in- to convey the meaning or purpose of the placeholder.

**5.1.1.9 Current Page Number** The Current Page Number block is used to display the Automatically insert the number of the current page on which the block is located. This allows readers to easily identify which page of the document they are currently on. You can position this block flexibly in your document to use it in footers, headers, or other areas where the page number might help the reader orient themselves. The available formatting options correspond to those of the text block, as in section

5.1.1.7 described.

**5.1.1.10 Total Page Number** The Total Page Number block indicates the total number of pages in your document. This is particularly helpful to readers to give an overview of how extensive the document is. Similar As with the current page number, this block can be placed anywhere in the document, but is often used in combination with the block for the current page number used to create formats like page X of Y. The available formatting options correspond to those of the text block, as shown below. Section 5.1.1.7 described.

### 5.1.2 Positioning

The Positioning category groups together blocks that focus on the precise placement and alignment of content within your PDF document. They enable fine control over horizontal and vertical expansion, scaling, rotation, shrinkage, translation as well as the determination of the width and height of specific elements. These tools are essential to achieve a precise and aesthetically pleasing layout by ensuring that all elements from text to images to graphics are optimally positioned and displayed.

**5.1.2.1 Alignment** This block is used to align content within an area in your PDF document. You can use both horizontal and

Also configure vertical alignment to place your content exactly where you need it. This offers a flexible design option to present your content in an appealing and clear manner.

#### **Features: Optional**

— **Horizontal alignment:** Defines the horizontal positioning of the content. Possible values are 'Left', 'Middle' and 'Right'. This setting determines where the content will be horizontally aligned within the block.

— **Vertical Alignment:** Defines the vertical positioning of the content. Possible Values are 'Top', 'Middle' and 'Bottom'. This setting determines where the content will be vertically aligned within the block.

**5.1.2.2 Aspect Ratio** This block is used to set the aspect ratio an area in your PDF document. The aspect ratio determines how the width and height of the content relate to each other, which is particularly useful for correctly sizing and displaying visual elements such as images or diagrams.

#### **Characteristics:**

##### **Necessary**

— **Aspect Ratio:** Sets the ratio between the width and height of the block. The value is a floating point number that indicates the ratio (e.g. 1.6 for an aspect ratio of 16:10).

**5.1.2.3 Expand Horizontally and Vertically** This block is used to expand the content within an area of your PDF document to make maximum use of the available space. It is particularly useful for ensuring

that the content, such as images or blocks of text, completely fills the assigned area without having to specify specific dimensions.

**5.1.2.4 Expand Horizontally** This block is used to expand the content within to expand horizontally within an area of your PDF document so that it uses the entire available width. It is ideal for ensuring that horizontal elements, such as lines or blocks of text, have the full width of the assigned of the area without having to provide specific width information.

**5.1.2.5 Expand Vertically** This block is used to expand the content within half of an area of your PDF document to expand vertically so that it covers the entire uses available height. This is particularly good for ensuring that vertical elements, such as columns or images, the full height of the assigned Occupy the area without the need for specific height information.

**5.1.2.6 Rotate Horizontally** This block is used to rotate the content within to flip half of an area of your PDF document horizontally. This function is useful to achieve effects or special layout requirements by reversing content such as text, images or graphics on the horizontal axis being represented.

**5.1.2.7 Flip** This block is used to flip the content within a Rotate the area of your PDF document 180 degrees as if it were upside down would be provided. This function can be used for creative layout designs or special ones Display effects can be helpful by allowing content such as text, images or Presenting graphics in a way that makes them appear to be upside down.

**5.1.2.8 Rotate Vertically** This block is used to rotate the content inside Flip an area of your PDF document vertically. This feature is particularly useful for creating effects or special layouts by reversing content such as text, images or graphics along the vertical axis, resulting in a mirror-image representation.

**5.1.2.9 Height** This block is used to set the height of an area in your Set PDF document. This is useful to ensure that content like Text, images or graphics take up exactly the desired height, which is what the precise layout design and compliance with design specifications.

**Characteristics:**

**Necessary**

— **Height:** Sets the height of the block. The information is given as a floating point number.

**5.1.2.10 Min/Max Height** This block is used to set the minimum and

Set the maximum height of an area in your PDF document. This flexibility allows you to design content so that it can vary within a defined height range, which is particularly useful for responding to different amounts of content or ensuring that the layout remains consistent under different conditions.

**Features: Optional**

- **Minimum Height:** Sets the minimum height of the block. The information is given as a floating point number. This ensures that the block does not shrink below a certain height.
- **Maximum Height:** Sets the maximum height of the block. The information is given as a floating point number. This prevents the block from having a specific height grows.

**5.1.2.11 Min/Max Width** This block is used to set the minimum and maximum width of an area in your PDF document. By specifying these values, you can ensure that your content is displayed flexibly within a defined width range. This is particularly advantageous in order to be able to react to varying amounts of content or to ensure a consistent layout across different content sizes.

**Features: Optional**

- **Minimum Width:** Sets the minimum width of the block. The information is given as a floating point number. This ensures that the block does not become narrower than a certain value.
- **Maximum Width:** Sets the maximum width of the block. The information is given as a floating point number. This prevents the block from becoming wider than a certain value.

**5.1.2.12 Padding** This block allows you to apply padding around the content within an area of your PDF document. Adding padding allows you to adjust the distance between the edge of the block and its contents, which helps provide visual separation and improve readability. This property is particularly useful for presenting content in an aesthetically pleasing manner.

**Features: Optional**

- **Left padding:** Sets the padding on the left side of the content. The Specification is given as a floating point number.
- **Top Spacing:** Sets the spacing at the top of the content. The Specification is given as a floating point number.
- **Right Spacing:** Sets the spacing on the right side of the content. The Specification is given as a floating point number.
- **Bottom padding:** Sets the padding at the bottom of the content. The Specification is given as a floating point number.

**5.1.2.13 Rotate Left** This block is used to rotate the content within an area of your PDF document to the left.

**5.1.2.14 Rotate Right** This block is used to rotate the content within an area of your PDF document to the right.

**5.1.2.15 Scaling** This block is used to scale the content within a area of your PDF document. Scaling can help Content such as text, images or graphics according to your design requirements be it to accommodate more content on a page or to highlight certain elements.

**Characteristics:**

**Necessary**

- **Scaling:** Defines the scaling factor as an integer. A value of 100 corresponds to the original size, values greater than 100 increase and Values less than 100 reduce the size of the content.

**5.1.2.16 Scale on demand** This block is used to scale the content within within an area of your PDF document so that it fits completely in fits the assigned area without changing the aspect ratio of the content. This is particularly useful for ensuring that images, graphics or other visual elements are displayed in full without being cut off or going beyond the intended area.

**5.1.2.17 Shrink** This block is used to shrink the content inside Resize an area of your PDF document to fit the available space without resizing the area itself. This feature is particularly helpful to prevent overflow and ensure that all content remains visible, even if space is limited.

**5.1.2.18 Shrink Horizontally** This block is used to display the content to shrink horizontally within an area of your PDF document so that it fits into the available width. This functionality is special useful for ensuring that content such as text, images or diagrams remain visible within the intended horizontal space without causing overload. run or unwanted upheavals.

**5.1.2.19 Shrink Vertically** This block is used to vertically shrink the content within an area of your PDF document so that it fits in the available height is suitable. This functionality allows content such as texts, images or tables so that they remain within the specified vertical space without overflow. This is particularly useful for improving the readability and layout of the document to ensure limited space.

**5.1.2.20 Shift** This block allows you to move the content within an area of your PDF document along the X and Y axes. This Function can be used to precisely position the content or to Achieve special layout effects by adding content such as text, images or graphics be moved relative to their original position.

### **Characteristics:**

#### **Optional**

- **X Displacement:** Defines the displacement of the content along the X axis (horizontal). The information is given as a floating point number.
- **Y Shift:** Defines the displacement of the content along the Y axis (vertical). The information is given as a floating point number.

**5.1.2.21 Unrestricted** This block is used to display the content within a region of your PDF document without any restrictions the size to render. This means that the content remains its natural size, regardless of the size limitations of the surrounding container. This can be useful if you want to ensure that certain content, like Images or diagrams can be displayed in their full size without being automatically scaled.

**5.1.2.22 Width** This block is used to set the width of an area in your PDF document. This is particularly helpful to ensure



that content such as text blocks, images or graphics take up exactly the desired width, which supports the precise design of layouts and compliance with design specifications.

**Characteristics:**

**Necessary**

- **Width:** Sets the width of the block. The information is given as a floating point number.

### 5.1.3 Content Flow

The Content Flow category groups together blocks that determine the layout of your PDF document in a targeted manner. You control the spacing between content, force page breaks or prevent them for optimal readability and to ensure attractive design. These include functions for conditional display, one-time display or the targeted skipping of content. Such control elements are essential for precise document structuring, which improves both the information presentation and the aesthetic document design.

**5.1.3.1 Secure Space** This block is used to ensure that in a certain minimum vertical distance within an area of your PDF document status is present. This can be useful to ensure that enough space is available for subsequent content or to break a page, before a new section begins if there is not enough space available.

#### **Characteristics:**

##### **Necessary**

- **Height:** Defines the minimum height of the room to be ensured. The information is given as an integer.

**5.1.3.2 Page Break** This block is used to create a page break in your PDF document. This means that the content that follows following this block starts on a new page. This can be helpful to clearly separate document sections, start chapters or simply ensure that certain content always appears at the top of a page.

**5.1.3.3 Show Full** This block is used to ensure that the content within an area of your PDF document is completely visible without parts of it being cut off. This is particularly useful to display content such as images, tables or diagrams in their entirety if this means the surrounding container needs to be adjusted to record the entire content.

**5.1.3.4 Show if** This block is used to display content within your PDF document based on a specific condition. If the condition is met, the content is displayed; otherwise the content will be skipped and not displayed in the document. This enables a dynamic

dynamic content representation based on predefined criteria or user input can be made dependent.

**Characteristics:**

**Necessary**

- **To be displayed:** Defines whether the content is based on a Boolean condition should be displayed. The information is given as a Boolean value.

**5.1.3.5 Show Once** This block is used to ensure that the content within an area of your PDF document is displayed exactly once regardless of how often the block appears in the document.

**5.1.3.6 Skip Once** This block is used to skip the content within a region of your PDF document on its first occurrence and only render the content on subsequent instances of the same block.

This is useful for situations where specific information or sections only appear after the first occurrence of a block in the document should.

**5.1.3.7 Stop page breaks** This block is used to prevent the automatic insertion of page breaks within a specific content area of your PDF document. This means that all content remains

that comes after this block, on the same side, assuming there is enough space. This can be useful to ensure certain sections or elements are not separated by a page break, such as tables, graphics or blocks of text that are presented together. that should.

### 5.1.4 Layout

This category contains blocks that define the structure and arrangement of content within the PDF document. From the basic division in pages and rows to detailed placement in columns, inline formatting and layers on top of each other, these blocks enable the document to be designed precisely.

**5.1.4.1 Page blocks** Page blocks form the basic structure of every PDF document. You define the individual pages of the document and keep

There are specific blocks underneath that contain the content, the header and the footer structure a page. The flexible design makes it possible to customize each page and tailor it to the needs of the document. It is important to note that the block for the content of the page is always under the page must exist in order to create a document.

#### Specific sub-blocks:

- **Content of the page:** This sub-block is essential and must be on every page to be available. It houses the main content of the page, such as text and images or diagrams. The content of the page block is the central element that makes up the primary information of the document.
- **Page footer:** The footer is an optional sub-block that can display additional information at the bottom of each page. Typically it includes dates, page numbers or legal information. The footer is used for orientation within the document and can be used to reinforce brand identity.
- **Page header:** Similar to the footer, the header offers space for recurring information at the top of the page. She can go to For example, contain the company logo, the document title or the chapter heading. The header increases readability and professionalism Appearance of the document.

**5.1.4.2 Line Blocks** Line blocks are essential for the vertical organization of content in your PDF document. Lines can have an automatic or have a fixed height, adapting to different design and layout requirements.

**Automatic page breaks:** An important aspect of lines with automatic Height is the handling of page breaks. If the content of a line is If the available space on a page exceeds, the system automatically creates one

Page break through. This ensures that all content is complete and displayed correctly without manual intervention. The automatic page break function is particularly useful for documents with long text passages or variable content lengths, as it ensures consistent readability across the entire document.

#### Line types:

- **Line (Automatic):** Dynamically adjusts its height to the contained content at. This adaptability is ideal for content of different sizes and automatically performs page breaks if necessary to ensure optimal results to ensure representation.
- **Line (Constant):** Has a fixed height that is the same regardless of the content remains. This enables a uniform appearance and is advantageous when a consistent line height across different parts of a document is needed.

**Row container:** The parent container for row blocks coordinates them vertical arrangement and enables flexible positioning of the content at automatic or constant heights. This structured organization supports the creation of complex layouts.

**5.1.4.3 Column Blocks** Column blocks offer a flexible method to display content to arrange next to each other in column form in your PDF document. They are particularly useful for organizing text, images and other elements in a structured and to present an aesthetically pleasing layout. The columns can be adjusted automatically, constant or relative to the available width to meet a variety of design requirements.

#### Column types:

- **Columns (Automatic Width):** These columns automatically adjust their width to the content. They are ideal for content whose scope is not known in advance or can vary. The automatic width adjustment ensures that the content is displayed optimally without manual adjustments are required.
- **Columns (Constant):** Constant columns have a fixed width, which depends on the user is defined. This option is suitable for layouts that require a strict require visual consistency between pages or elements. Through By setting a constant width, the appearance of the document can be precisely controlled.

- **Columns (Relative):** Relative columns use a percentage of the available width. This flexibility makes it possible to dynamically adapt content to the overall width of the container, which is particularly advantageous for responsive designs. Relative width adjustment adjusts the column width in relation to the other elements and the available space.

**Column container:** The column container serves as a parent container that holds and coordinates the different column types. It ensures that columns are arranged and rendered correctly according to their specific properties. Within this container, content can be flexibly organized in automatic, constant or relative columns to create complex layouts and structures.

**5.1.4.4 Inline** The Inline block is a versatile component that allows Arrange content horizontally within your PDF document. this function is particularly useful for laying out text, images and other elements in a flowing line, making it easier to create complex and visually appealing documents. The inline array supports various alignment and spacing parameters to ensure precise placement of elements.

ten.

#### **Characteristics:**

##### **Optional**

- **Vertical Spacing:** Allows you to set the spacing between the Inline elements in vertical direction.
- **Horizontal Spacing:** Allows you to set the spacing between the Inline elements in horizontal direction.
- **Baseline Alignment:** Determines the vertical alignment of elements relative to the baseline of the surrounding text line. Options include top, middle and bottom.
- **Inline Alignment:** Sets the horizontal alignment of the inline elements within their container. Possible values are left, center, right, justified and around space.

**5.1.4.5 Layers** The Layers block enables the overlapping arrangement of content in your PDF document by providing different layers on which elements can be placed. This is particularly useful for visual effects such as shading, overlays or for creating depth

to achieve. The special feature of this block is that the elements are processed hierarchically from top to bottom, with the last one added Element (the lowest block in the hierarchy) is visually located above the others.

### 5.1.5 Other blocks

In addition to the specific blocks for designing PDF documents, such as barcodes and form fields, there are a number of other blocks that offer a wide range of functionality to improve the structure and appearance of your document. These include blocks for content direction, debugging, text styles, hyperlinks, and sections.

**5.1.5.1 Left to Right Content Direction** This block is used to set the left to right alignment of the content within an area of your PDF document. This is particularly useful for supporting reading order in languages that are written from left to right and can help improve the clarity and readability of the document.

**5.1.5.2 Right to Left Content Direction** This block is used to set the right to left alignment of the content within an area of your PDF document. This is particularly useful for supporting languages that are traditionally written from right to left, such as Arabic or Hebrew, and helps improve the readability and comprehension of the document.

**5.1.5.3 Debug Area** This block is used to mark a debug area in your PDF document. This can be particularly true during development

During the development phase, it can be helpful to place visual clues within the document to help with troubleshooting or checking the document structure.

The debug area can optionally contain text that provides additional information or comments.

#### **Features: Optional**

- **OptionalText:** Allows you to specify a text that will be displayed in the debug area. This text can be used to identify or provide additional information in debug mode.

**5.1.5.4 Default Text Style** This block is used to set the default text style for content within an area of your PDF document. This includes a wide range of style properties such as font size, font family, bold, italic, underline, strikethrough and many others that style the text the way you want. The flexibility of this block allows for consistent text representation, which is particularly useful when creating large documents with consistent style guidelines.



## Characteristics:

### Optional

- **Bold:** If selected, the text will appear in bold.
- **Extra Black:** If selected, the text will appear in extra black font strength shown.
- **Italic:** If selected, the text will be italicized.
- **Strikethrough:** If selected, the text will appear strikethrough.
- **Underlined:** If selected, the text will be underlined.
- **Font size:** Defines the size of the font.
- **Font color:** Defines the color of the text.
- **Background Color:** Defines the background color of the text.
- **Font Family:** Defines the font family for the text.
- **Line height:** Defines the line spacing.
- **Letter spacing:** Defines the space between letters.
- **Line break everywhere:** Allows a line break at any point in the Text.
- **Font Weight:** Defines the font weight.
- **Text alignment:** Defines the alignment of the text.
- **Thin:** If selected, the text will be displayed in thin font weight.
- **Extra Light:** If selected, the text will be in extra light font weight shown.
- **Light:** If selected, the text will be displayed in a light font weight.
- **Normal:** If selected, the text will be displayed at normal font weight.
- **Medium:** If selected, the text will be displayed in medium font weight.
- **Semi-bold:** When selected, the text is displayed in semi-bold.
- **Extra Bold:** If selected, the text will be displayed in extra bold.
- **Black:** When selected, the text is displayed in black font weight.

— **Normal Position:** When selected, the text will be in normal vertical position  
Position shown.

— **Subscript:** If selected, the text will be displayed in subscript.

— **Superscript:** If selected, the text will be displayed in superscript.

**5.1.5.5 Hyperlink** This block is used to create a hyperlink in your  
Create PDF document. With it you can use text or other elements

Link to a URL so that clicking the link takes users to a web page or other online resource. This is  
particularly useful for accessing further information, external documents or other

point out relevant online content.

**Characteristics:**

**Necessary**

— **URL:** The URL to which the link should point. This enables the operator  
Users can go directly to the specified website or online resource by clicking on the linked content.

**5.1.5.6 Section** This block is used to create a section in your PDF document. A section can be used to  
group content thematically or to support a structured structure of the document.

Naming a section not only makes the document easier to navigate, but can also be helpful in creating a  
table of contents or assigning content to specific parts of the document

be.

**Characteristics:**

**Necessary**

— **Section Name:** The name of the section. This name serves as an identifier  
tion of the content area and can be used for referencing or navigation within the document.

**5.1.5.7 Section Link** This block is used to create a reference to a section in your PDF document. It allows  
you to set up a clickable link that takes the user to a predefined section within

half of the document. This is particularly useful for creating tables of contents or navigating large  
documents, improving document usability and accessibility.

**Properties:**

**Required**

- **Section name:** The name of the section to which the link should point. This name must match the name of a previously defined section in Document match to ensure successful linking ten.

## 5.2 Absolute blocks

Absolute blocks enable precise placement of elements at a fixed position within the document. This category is particularly useful for including images with precise positioning, form fields that need to be inserted at specific locations in the document, and barcode fields that need to be precisely aligned. In addition to these specific features, absolute blocks also support the inclusion of file attachments and other supporting materials that can be added to the document.

nen.

### 5.2.1 Image (absolute positioning)

This block allows you to insert an image into a specific position in your PDF document. The image can be loaded from a file or encoded as Base64. Additionally, you can set the position (X, Y) of the image on the page and optionally its size (width and height) to adjust the image according to your layout requirements.

#### Characteristics:

##### **Necessary**

- **Image File:** The path to the image file or image data encoded as Base64 to be displayed in the PDF.
- **Position X-Axis:** The X position of the image on the page, given as an integer number.
- **Position Y-Axis:** The Y position of the image on the page, specified as an integer number.

##### **Optional**

- **Size-X-Axis:** The width of the image, specified as an integer. If this property is set, SizeY must also be specified.
- **Size-Y-Axis:** The height of the image, specified as an integer. If this property is set, SizeX must also be specified.

## 5.2.2 Form fields

In the Form Fields category you will find a variety of blocks designed to enrich your PDF document with interactive elements. These elements range from radio buttons to check boxes to combination fields, which enables flexible design of surveys or forms. The radio buttons, in individual and group formats, provide the opportunity for individual selections and can be customized to create a coherent user experience. Checkboxes are ideal for consent or multiple selections, while combo boxes allow selection from a drop-down list, expanding input options.

Date and time fields provide a standardized way to capture time-related data, and list fields allow users to select multiple options from a list. Signature fields add a layer of authentication and verification through the ability to incorporate digital signatures,

which can be essential for official documents. After all, these are text fields

Basic framework for any interactive form in which users can enter their own text.

Each of these blocks comes with a set of customizable properties that allow the appearance and behavior of the form elements to be tailored to the exact needs of your document.

**5.2.2.1 Radio Buttons** These two blocks, Radio Button (Single) and Radio Button (Group), work together to create a group of radio buttons in your PDF document. While Radio Button (Individual) represents individual radio buttons, Radio Button (Group) is used to organize these buttons as a cohesive group.

The radio button (group) sets the general properties of the group, including the group name and the index of the default selected button. Multiple radio button (individual) instances can be placed within this block to represent each choice.

Each radio button can be individually customized, for example in terms of color, frame thickness, font and size.

Designing as a group allows radio buttons to be distributed throughout the document while remaining logically connected to each other. Only one button in the group can be selected, which is ensured by the group membership.

### **Radio Button (Single) Properties: Optional**

---

— **Color:** Defines the color of the text or symbol in the radio button.

- **Flatten:** Determines whether the radio button is displayed as non-interactive in the final PDF  
Element should be flattened.
- **Border width:** Specifies the width of the border around the radio button.
- **Text alignment:** Sets the alignment of the text in the radio button.
- **Font:** Determines the font of the text in the radio button.
- **Font size:** Defines the size of the text in the radio button.

#### **Radio button (group) properties:** \_\_\_\_\_

##### **Necessary**

- **Group name:** The unique name of the group to which the radio buttons belong  
belong.

##### **Optional**

- **Index of selected item:** Specifies the index of the radio button that should be  
selected by default.

**5.2.2.2 Checkbox** This block adds a checkbox to your PDF document. You can choose whether to have the box checked by default and whether to render it as a non-interactive element when exporting the PDF to preserve its current selection. In addition, the width of the frame around the box can be adjusted.

#### **Features:** \_\_\_\_\_

##### **Optional**

- **Checked:** Specifies whether the check box should be checked by default.
- **Flatten:** Specifies whether the field should be flattened in the final PDF, meaning  
it will not be interactive but the selected state will be visible.  
remains cash.
- **Border Width:** The width of the border around the check box.

**5.2.2.3 Combobox** This block adds a ComboBox to your PDF document that allows users to select an option from a drop-down list. You can define the available options, set a default element, and optionally flatten the field when exporting the PDF.

Additionally, you can customize the font and size for displaying the elements in the field.

#### **Features:** \_\_\_\_\_

##### **Optional**

- **Items:** A newline-separated list of options that appear in the combo box should be available.
- **Selected Index:** The index of the element that should be selected by default. The counting starts at 1.
- **Flatten:** Specifies whether the field should be flattened in the final PDF, meaning it will not be interactive but the selected value will be visible remains.
- **Font and font size (FontSize):** The font and size to use to display the options in the field. The font can be an embedded font.

#### **5.2.2.4 Date and Time** This block adds a date to your PDF document.

and time field. You can change the format of the date and time

to meet the specific needs of your document.

Optionally, you can also set the font and size for displaying the date and time.

This enables flexible design of forms or documents that require date and/or time entry.

#### **Features:**

##### **Optional**

- **Date and time format:** The format in which the date and time are displayed should be.
- **Font:** The font to use for the date and time. The font can be an embedded font.
- **Font Size:** The size of the font used for the date and time. should be applied.

**5.2.2.5 List box** This block adds a list box to your PDF document. List boxes allow users to select one or more options from a given list. You can define the elements of the list, set a default element as selected, and optionally flatten the field when exporting the PDF so that it is no longer interactive but the selected value remains visible.

#### **Features:**

##### **Optional**

- **Elements:** A newline-separated list of values expressed as Options should appear in the list box.

- **Selected Index:** The index of the element that should be selected by default. The counting starts at 1.
- **Flatten:** Specifies whether the field should be flattened in the final PDF, meaning it will be non-interactive but the selected value will remain visible.
- **Font and Font Size:** The font and size to use for the text in the field. The font can be an embedded font.

**5.2.2.6 Signature** This block allows you to add a signature field to your PDF document. You can optionally add a digital signature with a certificate, including contact information, location data, and the reason for the signature. You can also include an image to visually represent the signature. This feature is useful for legally signing documents and ensuring the authenticity and integrity of the document.

ten.

Please note that the password for the certificate is stored in plain text in the .BlockPDF file. This could pose a security risk and it is important to take appropriate precautions to ensure the protection of sensitive information.

**Properties:**

**Required**

- **Field name:** The name of the signature field, unique within the document.

**Optional**

- **Signature file:** The path to the certificate file or the binary data (Base64 encoded) of the certificate to be used for the digital signature.
- **Signature password:** The password for the certificate.
- **Signature Contact Info:** Contact information associated with the signature should be.
- **Signature Location Info:** Location information associated with the signature should be.
- **Signature reason:** The reason for the signature.
- **Image file:** An image to display in the signature field, e.g. B. a legal cant signature.



**5.2.2.7 Text field** This block allows adding a text field to a PDF document. A text field can be used to create interactive forms that allow the user to enter text. You can customize various properties of the text field, such as the text content, text color, font and size, as well as whether the field should be multi-line or not. You can also specify whether the field should be retained or flattened when exporting the PDF, which means that the text is permanently embedded in the PDF

and is no longer interactive.

**Characteristics:**

**Necessary**

- **Field name:** The name of the text field, unique within the document.
- **Position X-Axis, Position Y-Axis:** The position of the field on the page.
- **Size-X-Axis, Size-Y-Axis:** The size of the field.

**Optional**

- **Text:** The default text displayed in the field.
- **Color:** The text color.
- **Flatten:** Specifies whether the field should be flattened in the final PDF.
- **Border Width:** The width of the border around the text field.
- **Text Alignment:** The alignment of the text in the field.
- **Multi-line:** Specifies whether multi-line entries are allowed.
- **Font, Font Size:** The font and size of the text.

**5.2.3 File attachment**

This block allows you to add a file as an attachment to your PDF document.

You can specify the file name, the attachment path, or the attachment encoded as Base64, and optionally a description. This feature can be useful to provide additional information to be included along with the PDF

document should be transmitted, such as: B. Source documents, additional data or supporting materials.

**Characteristics:**

**Necessary**

- **Filename:** The name of the file as attached to the PDF document should appear.

- **Attachment file:** The path to the file or the binary data is encoded as Base64 File to add as an attachment.

### **Optional**

- **Description:** An optional description of the attachment, which can provide additional information about the attachment.

## 5.3 Barcode field

The Barcode Fields section covers various types of barcodes that you can add to your PDF document to encode data efficiently and effectively. These include DataMatrix, PDF417, QR code, Codabar, Code11, various Code128 variants, Code39, Code93, UPC, as well as EAN-13 and EAN-8.

### 5.3.1 Data Matrix

This block adds a DataMatrix barcode to your PDF document. DataMatrix barcodes are suitable for storing large amounts of data in a small space and are often used for marking products, documents and packages.

#### Characteristics:

##### **Necessary**

- **Position X Axis:** Sets the X position of the barcode on the page. The value is given as an integer.
- **Position Y-Axis:** Sets the Y position of the barcode on the page. The value is given as an integer.
- **Content:** Determines the content of the barcode. This is specified as a character string.

##### **Optional**

- **XDimension:** Defines the width of a single module in the barcode. The Value is given as a floating point number.
- **Size:** Sets the size of the DataMatrix barcode based on predefined ones sizes. This is determined by the 'PdfDataMatrixSize' enumeration.

### 5.3.2 Pdf417

This block adds a PDF417 barcode to your PDF document. PDF417 is a 2D barcode designed to store large amounts of data and is commonly used in logistics, identification and compliance applications.

#### Properties:

##### **Required**

- **Position X Axis:** Sets the X position of the barcode on the page. The value is given as an integer.
- **Position Y-Axis:** Sets the Y position of the barcode on the page. The value is given as an integer.

- **Content:** Determines the content of the barcode. This is specified as a character string.
- **Size X axis:** Defines the width of the barcode on the page. The value will be given as an integer.
- **Size Y axis:** Defines the height of the barcode on the page. The value will be given as an integer.

#### Optional

- **Error Correction Level:** Sets the level of error correction the barcode should use. This allows the barcode to still be read to a certain extent even if it is damaged.

### 5.3.3 QR code

This block adds a QR code to your PDF document. QR codes are widely used for quick access to websites, information and to connect devices. They offer high storage capacity and quick readability.

#### Properties:

##### Required

- **Position X axis:** Sets the X position of the QR code on the page. The value is given as an integer.
- **Position Y-Axis:** Sets the Y position of the QR code on the page. The value is given as an integer.
- **Content:** Determines the content of the QR code. This will be used as a string specified.
- **Size X axis:** Defines the width of the QR code on the page. The value will be given as an integer.
- **Size Y axis:** Defines the height of the QR code on the page. The value will be given as an integer.

#### Optional

- **Error correction level:** Sets the error correction level that the QR code should use to ensure readability even in the event of damage.
- **Input mode:** Determines the mode of data processing for the QR code, e.g. whether data is processed as text or in a binary mode.

### 5.3.4 Codabar

This block allows you to insert a Codabar barcode into your PDF document.

Codabar barcodes are widely used in libraries, blood banks and various courier services. The specific settings allow you to precisely adapt the barcode to your requirements, such as positioning, size and specific barcode properties.

#### Properties:

##### Required

- **Position X axis:** Determines the X (horizontal) position of the barcode on the Page.
- **Position Y-Axis:** Determines the Y (vertical) position of the barcode on the Page.
- **Content:** Defines the text content of the barcode.
- **Bar height:** Sets the height of the bars in the barcode.
- **Narrow Bar Width:** Determines the width of the narrow bars in the barcode.
- **Size-X-Axis:** Defines the width of the barcode.
- **Size-Y-Axis:** Defines the height of the barcode.

### 5.3.5 Code11

This block allows you to insert a Code 11 barcode into your PDF document.

The Code 11 barcode is commonly used to identify telecommunications devices and is known for its ability to efficiently encode numerical information. Due to the configurability of the position, the content,

the bar height as well as the widths and sizes of the narrow bars, the barcode can be designed exactly according to your needs.

#### Characteristics:

##### Necessary

- **Position X axis:** Determines the X (horizontal) position of the barcode on the Page.
- **Position Y-Axis:** Determines the Y (vertical) position of the barcode on the Page.
- **Content:** Defines the text content of the barcode.

- **Bar height:** Sets the height of the bars in the barcode.
- **Narrow Bar Width:** Determines the width of the narrow bars in the barcode.
- **Size-X-Axis:** Defines the width of the barcode.
- **Size-Y-Axis:** Defines the height of the barcode.

### 5.3.6 Code128, Code128A, Code128B, Code129C, GS1 Code128

These blocks allow you to insert different types of Code 128 barcodes into your PDF document. Code 128 barcodes are highly efficient for encoding alphanumeric or numeric data and are used in various industries for logistics, transportation, and merchandise management. Each variant has its special

facts:

**Code 128A** supports ASCII characters from 00 to 95 (09, AZ and control characters), as well as special characters.

**Code 128B** includes ASCII characters from 32 to 127 (09, AZ, az, and some special characters).

**Code 128C** is optimal for double encoding of numeric values and supports digit pairs 00 to 99, allowing for a more compact representation.

**GS1-128 (formerly UCC/EAN-128)** is used to globally uniquely identify elements within the supply chain and includes special application identifiers for additional data integrity.

#### **Properties:**

##### **Required**

- **Position X axis:** Determines the X (horizontal) position of the barcode on the Page.
- **Position Y-Axis:** Determines the Y (vertical) position of the barcode on the Page.
- **Content:** Defines the text content of the barcode.
- **Bar height:** Sets the height of the bars in the barcode.
- **Narrow Bar Width:** Determines the width of the narrow bars in the barcode.
- **Size-X-Axis:** Defines the width of the barcode.
- **Size-Y-Axis:** Defines the height of the barcode.

### 5.3.7 Code39, Code93

These blocks allow you to insert Code 39 and Code 93 barcodes into your PDF document. Both Code 39 and Code 93 are versatile barcode standards used in a variety of industries for inventory, identification and tracking management.

**Code 39** is one of the oldest and most widely used barcodes, known for its simplicity and reliability. It can encode alphanumeric characters (AZ, 0-9) as well as some special characters. **Code 93** extends the capabilities of Code 39 by providing higher data density and the ability to encode the entire ASCII character set. Code 93 is often used for inventory management and logistical tracking because it can store more information in a smaller space.

#### **Properties:**

##### **Required**

- **Position X-Axis:** Determines the horizontal position of the barcode on the Page.
- **Position Y-Axis:** Determines the vertical position of the barcode on the page.
- **Content:** Defines the text content of the barcode to be encoded.
- **Bar Height:** Sets the height of the bars within the barcode.
- **Narrow Bar Width:** Determines the width of the narrowest bars in the barcode and thus influences the overall width of the barcode.
- **Size-X-Axis:** Specifies the overall width of the barcode.
- **Size-Y-Axis:** Specifies the overall height of the barcode.

### 5.3.8 Code UPC

This block allows you to insert a UPC (Universal Product Code) barcode into your PDF document. UPC barcodes are widely used in retail, especially especially in the USA and Canada, for the clear identification of products at checkouts and in inventory systems. The UPC barcode consists of 12 digits that contain specific information about the manufacturer and the product.

This type of barcode is essential for retail and provides a quick and efficient way to scan and identify products.

#### **Characteristics:**

##### **Necessary**

- **Position X-Axis:** Determines the horizontal position of the barcode on the Page.
- **Position Y-Axis:** Determines the vertical position of the barcode on the page.
- **Content:** Defines the numeric content of the barcode to be encoded.
- **Bar Height:** Sets the height of the bars within the barcode.
- **Narrow Bar Width:** Determines the width of the narrowest bars in the barcode, which affects the overall width of the barcode.
- **Size-X-Axis:** Specifies the overall width of the barcode.
- **Size-Y-Axis:** Specifies the overall height of the barcode.

### 5.3.9 EAN-13, EAN-8

These blocks allow the creation of EAN-13 and EAN-8 barcodes in your PDF document. EAN barcodes are globally used standards for marking products in retail and offer a universal method for product identification.

EAN-13 is the more commonly used standard, consisting of 13 digits that represents a global item identification number. It is used primarily for international trade and includes a country identifier, a manufacturer identifier, an item number and a check digit. EAN-8 is a shortened version of the EAN-13 barcode, consisting of 8 digits, and is used for products where space for a barcode is limited. It contains a country code, a manufacturer code and a check digit.

#### Characteristics:

##### **Necessary**

- **Position X-Axis:** Determines the horizontal position of the barcode on the Page.
- **Position Y-Axis:** Determines the vertical position of the barcode on the page.
- **Content:** Defines the numeric content of the barcode to be encoded.
- **Bar Height:** Sets the height of the bars within the barcode.
- **Narrow Bar Width:** Determines the width of the narrowest bars in the barcode and thus influences the overall width of the barcode.
- **Size-X-Axis:** Specifies the overall width of the barcode.
- **Size-Y-Axis:** Specifies the overall height of the barcode.



## 5.4 Other blocks

This block category includes blocks that cannot be fully assigned to either relative blocks or absolute blocks.

### 5.4.1 Comment

This special block type is designed for structuring and commenting within the document to make editing and overview easier. The comment block displays the name specified in the properties and a selected background color in the block hierarchy. This visual marking helps quickly identify and organize sections or important notes in the document drafting process. The background color of a comment block is also applied to all subsequent blocks until another comment block updates this setting with a new color.

This function enables a clear visual separation of different documents  
ment sections or notes.

#### Characteristics:

##### Optional

- **Text:** The text of the comment serves as a note or description for the section or the specific function of the subsequent blocks.
- **Background color:** The background color provides a visual demarcation of the annotated area. It supports the visual structuring of the document and helps to distinguish different areas at a glance.

### 5.4.2 Iterator

**The Iterator block** in BlockPDF is used to iteratively display content based on the data of a data set. This allows dynamic creation of PDF content, such as iterating through a list of data and outputting each element as a separate section or line in the document. The main function of the Iterator block is to iterate through a set of data and draw specific sub-blocks for each data element, which facilitates the automated creation of complex documents with repeating structures or patterns.

#### Characteristics:

##### Necessary

- **Iterator workflow name:** The name of the workflow that determines how the iteration through the data takes place. This name is used to describe the specific

Identify the specific workflow responsible for processing the data is.

- **Iterator data:** The actual data being iterated over. These dates can come from different sources and be in formats such as XML, JSON, CSV or separated by line breaks.
- **Iterator data type:** The type of data that defines the structure of the iterator data. For example, this could be a specific schema or format that specifies how the data should be interpreted and presented.

### 5.4.3 Repeat

This block makes it possible to replicate certain content or block structures based on a defined number. This mechanism is particularly useful for creating recurring structures such as lists, table rows, or any type of repetitive patterns within a PDF document.

The core function of this block is to draw child blocks multiple times according to the value of the property. This allows users to efficiently design complex documents with dynamic content without manual repetition.

#### Features: Optional

- **Show n times:** Determines the number of repetitions for the child blocks. This property allows the amount of replications to be dynamically adjusted and is crucial to the flexibility of the **repeat block** in document creation.



## Command lines (CLI)

The BlockPDF software's command line interface (CLI) allows users to generate PDF documents by executing specific commands.

This interface supports various options to customize the document creation process.

### 6.1 Parameter list

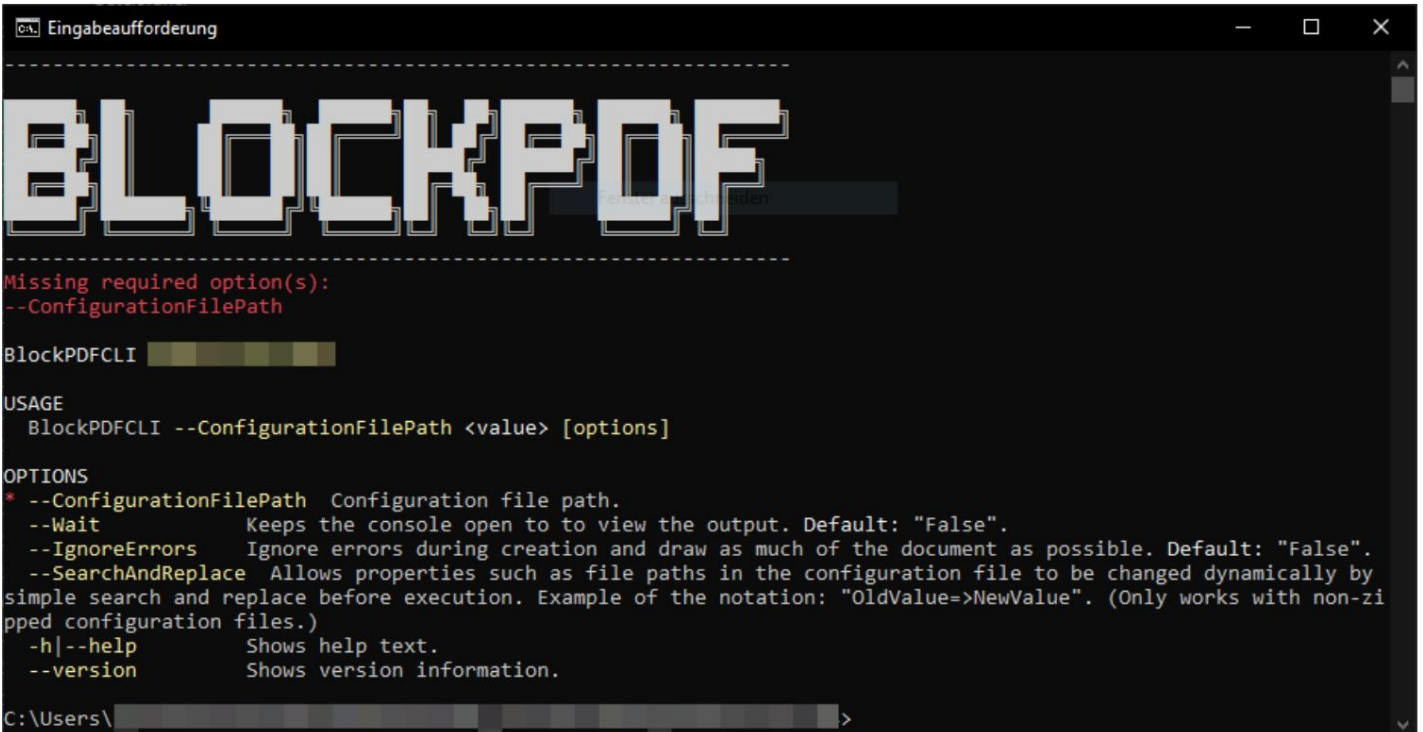
The CLI accepts several parameters that control the generation of documents:

- **ConfigurationFilePath** (Required): The path to the configuration file contains the specifications for the document to be generated.
- **Wait** (Optional): Holds the console open so the user can see the output. Useful for debugging purposes or to confirm successful execution.
- **IgnoreErrors** (Optional): Allow document generation even if errors occur and tries to draw as much of the document as possible.  
nen.
- **SearchAndReplace** (Optional): Allows you to dynamically change properties in the configuration file using simple search-and-replace operations.

Operations before execution.

## 6.2 Console Application Outputs

The CLI outputs information about the status of document generation, including success messages and errors. Success messages contain details about the documents created, while error messages indicate problems in the generation process.



```
-----  
BLOCKPDF  
-----  
Missing required option(s):  
--ConfigurationFilePath  
  
BlockPDFCLI ██████████  
  
USAGE  
BlockPDFCLI --ConfigurationFilePath <value> [options]  
  
OPTIONS  
* --ConfigurationFilePath Configuration file path.  
--Wait Keeps the console open to to view the output. Default: "False".  
--IgnoreErrors Ignore errors during creation and draw as much of the document as possible. Default: "False".  
--SearchAndReplace Allows properties such as file paths in the configuration file to be changed dynamically by  
simple search and replace before execution. Example of the notation: "OldValue=>NewValue". (Only works with non-zi  
pped configuration files.)  
-h|--help Shows help text.  
--version Shows version information.  
  
C:\Users\██████████ >
```

CLI - Without arguments

This figure shows the CLI output when the command is run without required arguments. In this case, users are typically asked to specify the necessary parameters.

```
Eingabeaufforderung
C:\Users\>BlockPDFCLI.exe --ConfigurationFilePath C:\U
sers\test.BlockPDF
-----
BLOCKPDF
-----
Version:
License: DEMO
Loading configuration file...
Generating documents...
{"Type":"Success","Index":1,"FilePath":"C:\\Users\\\\File name.pdf"}
All documents generated successfully.
C:\Users\>
```

CLI - Successful document creation

The successful creation of documents is shown here, including the paths to the generated PDF files.

```
Eingabeaufforderung
C:\Users\>BlockPDFCLI.exe --ConfigurationFilePath C:\U
sers\test.BlockPDF
-----
BLOCKPDF
-----
Version:
License: DEMO
Loading configuration file...
Generating documents...
{"Type":"Failure","Index":1,"Information":"Exception of type 'BlockPDF_Core.RequiredPropertyIsNotSetException' was
thrown."}
Some documents could not be generated.
C:\Users\>
```

CLI - error output

This figure shows the CLI output when an error occurs during document creation.

# Documents, templates and Block structures

BlockPDF uses multiple file types for creating and managing Sub-elements of PDF documents can be used. this includes

.BlockPDF documents, template files and ready-made block structures that make the creation of documents easier and faster.

## **7.1 BlockPDF document (.BlockPDF)**

The .BlockPDF document file is a JSON formatted file used to define document layouts in BlockPDF. It allows the elements and structures of a document to be described. It contains all blocks and set properties in a more or less human-readable form.

In certain cases it may be advisable to adapt individual aspects of a saved document outside of the BlockPDF application, for example if you want to specify a relative path for the output folder or use system environment variables such as %USERPROFILE% in the path under Windows.

### **7.1.1 Example .BlockPDF file**

The main elements of the file structure include, in this example, the following elements:  
ment:

- **\$id and \$type:** Identify instances and types of the objects. \$type returns the name of the type, which is important for editing within the BlockPDF software.
- **Version:** Specifies the version of the document file.
- **Document:** The root element containing details such as *Identifier*, *BlocksShallThrowExceptions*, and *Children* contains:
  - *Identifier:* A unique ID of the document.
  - *BlocksShallThrowExceptions:* Whether blocks throw exceptions in the event of errors should fen.
  - *Children:* List of child elements that make up the document.
- **PageBlock:** Represents a PDF page, with *PageFormatSize* and children's elements for content.
- **TextBlock:** Example of a content block, with properties such as *text* and *bold* for text display.
- **FilenameGenerator:** Defines the generation of the file name for the output PDF.
- **Paths:** *OutputDirectoryPath*, *LastSaveDirectoryPath*, *LastSaveFileName* save locations and names of the last saved document.

---

```

1 {
2   "$id": "1",
3   "$type": "BlockPDF_Core.PersistenceObject, BlockPDF_Core",
4   "Version": "1.0.0.0",
5   "Document": {
6     "$id": "2",
7     "$type": "BlockPDF_Core.BlockDocument, BlockPDF_Core",
8     "Identifier": "a94781dd-9af5-4d02-9b07-dccb3f99b53d",
9     "BlocksShallThrowExceptions": true,
10    "Children": [
11      {
12        "$id": "4",
13        "$type": "BlockPDF_Core.PageBlock, BlockPDF_Core",
14        "PageFormatSize": {
15          "$id": "5",
16          "$type":
17            "BlockPDF_Core.ConstantData`1[[BlockPDF_Core.PageFormat,
            BlockPDF_Core]], BlockPDF_Core",
18        "Data": {

```

```
18         "$id": "6",
19         "$type": "BlockPDF_Core.PageFormat, BlockPDF_Core",
20         "Width": 595.4,
21         "Height": 842.0,
22         "Unit": "Point"
23     }
24 },
25     "Children": [
26     {
27         "$id": "7",
28         "$type": "BlockPDF_Core.PageContentBlock, BlockPDF_Core",
29         "Child": {
30             "$id": "8",
31             "$type": "BlockPDF_Core.TextBlock, BlockPDF_Core",
32             "Text": {
33                 "$id": "9",
34                 "$type":
35                 ·   "BlockPDF_Core.ConstantData`1[[BlockPDF_Core.GenericClassValue`1[[System.String,
36                 ·   System.Private.CoreLib]], BlockPDF_Core]],
37                 ·   BlockPDF_Core",
38             "Data": {
39                 "$id": "10",
40                 "$type":
41                 ·   "BlockPDF_Core.GenericClassValue`1[[System.String,
42                 ·   System.Private.CoreLib]], BlockPDF_Core",
43                 "Value": "Hello, World!"
44             }
45         },
46         "Bold": {
47             "$id": "11",
48             "$type":
49             ·   "BlockPDF_Core.ConstantData`1[[BlockPDF_Core.GenericStructValue`1[[System.Boolean,
50             ·   System.Private.CoreLib]], BlockPDF_Core]],
51             ·   BlockPDF_Core",
52             "Data": {
53                 "$id": "12",
54                 "$type":
55                 ·   "BlockPDF_Core.GenericStructValue`1[[System.Boolean,
56                 ·   System.Private.CoreLib]], BlockPDF_Core",
57                 "Value": true
58             }
59         },
60         "TreeViewsExpanded": true
61     },
62     "Children": [
63     {
64         "$ref": "8"
```



```

55         }
56     ],
57     "TreeViewsExpanded": true
58 }
59 ],
60     "TreeViewsExpanded": true
61 }
62 ],
63     "TreeViewsExpanded": true
64 },
65     "FilenameGenerator": {
66         "$id": "13",
67         "$type": "BlockPDF_Core.FilenameGenerator, BlockPDF_Core",
68         "Elements": [
69             {
70                 "$id": "14",
71                 "$type": "BlockPDF_Core.FilenamePartConstant, BlockPDF_Core",
72                 "Constant": "Example file name"
73             }
74         ]
75     },
76     "OutputDirectoryPath": "C:\\Users\\USER\\Downloads\\",
77     "LastSaveDirectoryPath": "C:\\Users\\USER\\Downloads",
78     "LastSaveFileName": "example.BlockPDF"
79 }

```

---

## 7.2 Template files

### 7.2.1 Document template (.BlockPDF.zip)

The document template (.BlockPDF.zip) is a compressed file that contains a .BlockPDF file and additional resources such as images, Excel files, fonts, etc.

holds. This facilitates the exchange and distribution of document templates, as all the required elements are bundled in a single file. Document templates can be created in the application via the save dialog.

All file paths are automatically adapted for exchange. Included

Please note that only file paths are adjusted. If an SQL connection with a password is used, this is still in the .BulkPDF file

in the template and must be removed by manual post-processing.

### 7.2.2 Prefabricated block structures (.BlockPart)

Prefabricated block structures (.BlockPart) are building blocks for .BlockPDF documents, which encapsulate commonly used block structures. Users can use this structure

Insert structures into your documents to quickly create complex layouts without having to start from scratch every time.

### 7.2.3 Provision of templates and block structures

BlockPDF uses a flexible system for finding and integrating templates and block structures. The software searches for specified directories Files with relevant file extensions, for example .BlockPDF for document templates and .BlockPart for block structures, and identifies them as usable Resources. This process allows templates directly from local file systems or network folders without requiring additional indexing. is required.

In addition to direct search in folders, BlockPDF offers the option to create templates and Define structures using JSON files containing detailed information for the individual resources. These JSON files consist of a Array of objects, where each object represents a nameable, writable entry with a source. The 'Source' can point to a local file, point to a network path or URL.

An example of the contents of such a JSON file could look like this:

---

```
1  [
2    {
3      "name": "Template1",
4      "Type": "BlockPart",
5      "Source": "https://example.com/vorlage1.BlockPart",
6      "Description": "A block structure for frequently used
7      ÿ Forms."
8    },
9    {
10     "name": "Template2",
11     "Type": "DocumentTemplate",
12     "Source": "https://example.com/vorlage2.BlockPDF.zip",
13     "Description": "Standard document template."
14   },
15   {
16     "name": "Template3",
17     "Type": "DocumentTemplate",
18     "Source": "C:\\Documents\\Templates\\template3.BlockPDF.zip",
19     "Description": "Complete template package including graphics and
20     ÿ Fonts."
21   }
22 ]
```

---

The two methods automatically recognize originals by scanning of folders by file types and optionally defining resources in JSON files allow customizable integration into different work contexts. Users can easily create templates locally or via the Obtain network and use as needed. This supports the shared use of document templates and structures.



## Local configuration file

This chapter describes the configuration of the BlockPDF software through the Configuration.ini file, which is located in the working directory of the program (under Windows probably: %appdata%\BlockPDF). The Configuration.ini file allows flexible customization of various aspects of the software to suit specific needs.

The contents of an example Configuration.ini file can be seen below and in the following sections discuss the categories and their individual options:

```

1  [UI]
2  IntroductionHasBeenShown = True
3  LastOpenFileDialogPath = C:\Users\USER\Downloads
4  LastOpenFolderDialogPath = C:\Users\USER\Downloads
5  LastSaveFileDialogPath = C:\...\Templates\Documents
6  AutomaticPreviewUpdate = True
7  GlobalUIScale = 1
8  =
9  Language = en
10 CheckForFileNameErrors = True
11 ShowWelcomeText = False
12
13 [template]
14 LastTemplateUpdate = 01/01/2024 00:00:00
15 DocumentTemplateSources = C:\...\Templates\Documents\
16
17 [AutoSave]
18
19 [Core]
20 SelectedUpdateChannel = Release

```

## 8.1 Template area

- **DocumentTemplateSources:** Directories in which document templates are searched become. Multiple directories can be specified separated by semicolons become.

*Example:* C:\Users\Users\Documents\Templates;C:\Templates\Shared

- **BlockPartSources:** Directories in which block part templates are searched. Multiple directories can be specified separated by semicolons become.

*Example:* C:\Users\Users\Documents\BlockParts;C:\BlockParts\Shared

- **AutomaticTemplatesUpdateEnabled:** Enables or disables automatic template updates. If selected, templates will be automatically table updated.

*Example:* true

- **HoursBetweenAutomaticUpdate:** The number of hours between automatic chen updates.

*Example:* 24

— **LastTemplateUpdate:** Time of the last template update.

*Example:* 2023-01-01T12:00:00

## 8.2 UI area

— **Language:** The language of the user interface.

*Example:* de

— **LastOpenFileDialogPath:** Last used path in the file open dialog.

*Example:* C:\Users\Users\Documents

— **LastSaveFileDialogPath:** Last used path in the file save dialog.

*Example:* C:\Users\Users\Documents

— **LastOpenFolderDialogPath:** Last used path in the folder selection dialog log.

*Example:* C:\Users\Users\Documents

— **AutomaticPreviewUpdate:** If selected, the preview will be updated automatically table updated.

*Example:* true

— **PreviewDPI:** DPI value for the preview.

*Example:* 100

— **PreviewScale:** Scaling value in percent for the preview.

*Example:* 50

— **AutomaticUpdateFilenamePreview:** If selected, the filename preview is preview updated automatically.

*Example:* true

— **MaximumNumberOfRowsForAPreview:** Maximum number of rows for a review Preview.

*Example:* 50

— **GlobalUIScale:** UI scaling factor.

*Example:* 1.0

— **DefaultValueIgnoreFirstRowOfData:** If selected, the first row the data is ignored.

*Example:* false

- **MaxNumberOfLastUsedDocuments:** Maximum number of last used documents mente.

*Example:* 5

- **CheckForFileNameErrors:** Checks for errors in file names.

*Example:* true

- **AskIfDocumentsShouldBeOverwritten:** Asks if documents should be overwritten should be.

*Example:* true

- **IntroductionHasBeenShown:** Indicates whether the introduction has already been shown became.

*Example:* false

- **BlockDocumentGifIntroductionHasBeenShown:** Indicates whether the introduction has already been displayed.

*Example:* false

- **NotifyWhenANewVersionIsAvailable:** Notify when a new version is available version is available.

*Example:* true

- **ShowWelcomeText:** Shows the welcome text.

*Example:* true

- **SkippedVersion:** Specifies a skipped version for notifications ments.

*Example:* 2.1.0

- **MaximumNumberOfSoftwarePageChangesSaved:** Maximum number saved secured page changes.

*Example:* 10

### 8.3 Core area

- **MaximumNumberOfPersistenceObjectStates:** Maximum number of states the one for persistence objects.

*Example:* 10

— **DefaultPageFormat:** Default format for pages.

*Example:* 595.4;842;Point

— **PDFImportDPI:** DPI for PDF imports.

*Example:* 300

— **SelectedUpdateChannel:** Selected update channel.

*Example:* Release

— **NodesCacheMaximum:** Maximum number of items in the nodes cache.

*Example:* 500

## 8.4 AutoSave area

— **AutoSaveEnabled:** When selected, auto save is enabled.  
tivated.

*Example:* true

— **AutoSaveInterval:** Auto save interval in minutes.

*Example:* 30

— **MaxAutoSaveAgeDays:** Maximum age of automatic saves  
in days.

*Example:* 7

— **MaxAutoSaveCopies:** Maximum number of automatic save copies.

*Example:* 3

TO DO





## Tips and Tricks

A selection of tips and tricks to help you create PDF documents to be able to help.

### **9.1 Import of normal PDF files as templates**

Using the Load Existing Document option on the Home page, you can select a PDF document, which is then converted into a BlockPDF document. This conversion takes place in a simple process in which the individual pages of the PDF can be embedded as images in the new document. For each page, a page block is created, to which a layer block is subordinated, into which the page image is inserted. Adjustments to the document can now be made by layering below the image additional block structures can be inserted.

### **9.2 Relative paths and system variables in documents and documentation templates**

Relative paths and system variables can be used in the document files become. The files (see section 7.1) must be linked to an external text editor (e.g. Visual Studio Code). The relative paths

and system variables are then resolved by BlockPDF when loading the file.

### 9.3 Relative rows

In contrast to column containers, which offer the possibility of relative columns  
To define, there is no direct equivalent for relative rows in row containers. However, similar  
behavior can be replicated through a creative approach. One method to simulate relative  
rows exists

is to first rotate a column container to the left and then the

Rotate the contents of each column to the right. This approach makes it possible to adjust  
the layout to resemble that of relative rows. It

However, it is important to note that this technique has limitations

This involves page breaks and therefore does not represent a complete solution in all  
situations.

### 9.4 Watermarks

Watermarks can be effectively designed and included using the Layers block  
PDF document can be integrated. For example, a layer block can be used directly  
after the "Contents of the pages" block, with the watermark then  
is the first block structure and the actual page content is the subsequent block structure.  
Depending on the sorting, the watermark will then be below  
Content or above this. This technique makes it possible to create semi-transparent images  
or to position texts above or below the main content without disturbing it.

### 9.5 Connection string

Examples of valid connection strings would be these for a connection with user name and password

```
1 Server=myServer;Database=myDataBase;User Id=myUser;Password=myPassword;
```

and for a "Trusted Connection" using Active Directory login, the following:

```
1 Server=myServer;Database=myDataBase;Trusted_Connection=True;
```

For further examples please take a look at the following website [https://  
www.connectionstrings.com/sql-server/](https://www.connectionstrings.com/sql-server/).

## 9.6 Print documents simultaneously

If all created documents are to be physically printed at the same time, the button “Merge all documents and open standard PDF viewer (e.g. for mass printing)” can be used on the last program page. After further confirmation, this recreates all documents and combines them into a complete PDF document. The created document will then be opened in the standard PDF viewer and can be printed immediately from there.

There you can also use the print options to set the pages that should not be printed.

## 9.7 Date formatting

A formatting string is used to format the date. This string consists of a few characters that define the format. The format **dd.mm.yyyy** is a valid date format that, for example, outputs the day, month and year (example: November 25th, 2021).

See the following table from the Microsoft .NET documentation[5]:

Format specifier	Description	Examples
"d"	The day of the month, from 1 to 31. More information: The "d" Custom Format Specifier.	2009-06-01T13:45:30 ->1 2009-06-15T13:45:30 ->15
"dd"	The day of the month, from 01 to 31. More information: The "dd" Custom Format Specifier.	2009-06-01T13:45:30 ->01 2009-06-15T13:45:30 ->15
"ddd"	The abbreviated name of the day of the week. More information: The "ddd" Custom Format Specifier.	2009-06-15T13:45:30 ->Mon (en-US) 2009-06-15T13:45:30 ->(ru-RU)
"dddd"	The full name of the day of the week. More information: The "dddd" Custom Format Specifier.	2009-06-15T13:45:30 ->Monday (en-US) 2009-06-15T13:45:30 ->(ru-RU) 2009-06-15T13:45:30 ->lundi (fr-FR)
"f"	The tenths of a second in a date and time value. More information: The "f" Custom Format Specifier.	2009-06-15T13:45:30.6170000 ->6 2009-06-15T13:45:30.05 ->0
"ff"	The hundredths of a second in a date and time value. More information: The "ff" Custom Format Specifier.	2009-06-15T13:45:30.6170000 ->61 2009-06-15T13:45:30.0050000 ->00
"fff"	The milliseconds in a date and time value. More information: The "fff" Custom Format Specifier.	6/15/2009 1:45:30.617 ->617 6/15/2009 13:45:30.0005 ->000
"ffff"	The ten thousandths of a second in a date and time value. More information: The "ffff" Custom Format Specifier.	2009-06-15T13:45:30.6175000 ->6175 2009-06-15T13:45:30.0000500 ->0000
"fffff"	The hundred thousandths of a second in a date and time value. More information: The "fffff" Custom Format Specifier.	2009-06-15T13:45:30.6175400 ->61754 6/15/2009 13:45:30.000005 ->00000
"ffffff"	The millionths of a second in a date and time value. More information: The "ffffff" Custom Format Specifier.	2009-06-15T13:45:30.6175420 ->617542 2009-06-15T13:45:30.0000005 ->000000
"fffffff"	The ten millionths of a second in a date and time value. More information: The "fffffff" Custom Format Specifier.	2009-06-15T13:45:30.6175425 ->6175425 2009-06-15T13:45:30.0001150 ->0001150
"F"	If non-zero, the tenths of a second in a date and time value. More information: The "F" Custom Format Specifier.	2009-06-15T13:45:30.6170000 ->6 2009-06-15T13:45:30.0500000 ->(no output)
"FF"	If non-zero, the hundredths of a second in a date and time value. More information: The "FF" Custom Format Specifier.	2009-06-15T13:45:30.6170000 ->61 2009-06-15T13:45:30.0050000 ->(no output)

"FFF"	If non-zero, the milliseconds in a date and time value.  More information: The "FFF" Custom Format Specifier.	2009-06-15T13:45:30.6170000 ->617  2009-06-15T13:45:30.0005000 ->(no output)
"FFFF"	If non-zero, the ten thousandths of a second in a date and time value.  More information: The "FFFF" Custom Format Specifier.	2009-06-15T13:45:30.5275000 ->5275  2009-06-15T13:45:30.0000500 ->(no output)
"FFFFF"	If non-zero, the hundred thousandths of a second in a date and time value.  More information: The "FFFFF" Custom Format Specifier.	2009-06-15T13:45:30.6175400 ->61754  2009-06-15T13:45:30.0000050 ->(no output)
"FFFFFF"	If non-zero, the millionths of a second in a date and time value.  More information: The "FFFFFF" Custom Format Specifier.	2009-06-15T13:45:30.6175420 ->617542  2009-06-15T13:45:30.0000005 ->(no output)
"FFFFFFF"	If non-zero, the ten millionths of a second in a date and time value.  More information: The "FFFFFFF" Custom Format Specifier.	2009-06-15T13:45:30.6175425 ->6175425  2009-06-15T13:45:30.0001150 ->000115
"g", "gg"	The period or era.  More information: The "g" or "gg" Custom Format Specifier.	2009-06-15T13:45:30.6170000 ->AD
"h"	The hour, using a 12-hour clock from 1 to 12.  More information: The "h" Custom Format Specifier.	2009-06-15T01:45:30 ->1  2009-06-15T13:45:30 ->1
"hh"	The hour, using a 12-hour clock from 01 to 12.  More information: The "hh" Custom Format Specifier.	2009-06-15T01:45:30 ->01  2009-06-15T13:45:30 ->01
"H"	The hour, using a 24-hour clock from 0 to 23.  More information: The "H" Custom Format Specifier.	2009-06-15T01:45:30 ->1  2009-06-15T13:45:30 ->13
"HH"	The hour, using a 24-hour clock from 00 to 23.  More information: The "HH" Custom Format Specifier.	2009-06-15T01:45:30 ->01  2009-06-15T13:45:30 ->13
"K"	Time zone information.  More information: The "K" Custom Format Specifier.	With DateTime values:  2009-06-15T13:45:30, Child Unspecified ->  2009-06-15T13:45:30, Child Utc ->Z  2009-06-15T13:45:30, Kind Local ->-07:00 (depends on local computer settings)  With DateTimeOffset values:  2009-06-15T01:45:30-07:00 ->-07:00  2009-06-15T08:45:30+00:00 ->+00:00
"m"	The minute, from 0 to 59.  More information: The "m" Custom Format Specifier.	2009-06-15T01:09:30 ->9  2009-06-15T13:29:30 ->29
"mm"	The minute, from 00 to 59.  More information: The "mm" Custom Format Specifier.	2009-06-15T01:09:30 ->09  2009-06-15T01:45:30 ->45
"M"	The month, from 1 to 12.  More information: The "M" Custom Format Specifier.	2009-06-15T13:45:30 ->6
"MM"	The month, from 01 through 12.  More information: The "MM" Custom Format Specifier.	2009-06-15T13:45:30 ->06
"MMM"	The abbreviated name of the month.  More information: The "MMM" Custom Format Specifier.	2009-06-15T13:45:30 ->Jun (en-US)  2009-06-15T13:45:30 ->juin (fr-FR)  2009-06-15T13:45:30 ->Jun (to-ZA)
"MMMM"	The full name of the month.  More information: The "MMMM" Custom Format Specifier.	2009-06-15T13:45:30 ->June (en-US)  2009-06-15T13:45:30 ->June (da-DK)  2009-06-15T13:45:30 ->uJune (to-ZA)
"s"	The second, from 0 through 59.  More information: The "s" Custom Format Specifier.	2009-06-15T13:45:09 ->9
"ss"	The second, from 00 through 59.  More information: The "ss" Custom Format Specifier.	2009-06-15T13:45:09 ->09
"t"	The first character of the AM/PM designator.  More information: The "t" Custom Format Specifier.	2009-06-15T13:45:30 ->P (en-US)  2009-06-15T13:45:30 ->(yes-JP)  2009-06-15T13:45:30 ->(fr-FR)

'dd'	The AM/PM designator. More information: The "tt" Custom Format Specifier.	2009-06-15T13:45:30 ->PM (en-US) 2009-06-15T13:45:30 ->(yes-JP) 2009-06-15T13:45:30 ->(fr-FR)
'y'	The year, from 0 to 99. More information: The "y" Custom Format Specifier.	0001-01-01T00:00:00 ->1 0900-01-01T00:00:00 ->0 1900-01-01T00:00:00 ->0 2009-06-15T13:45:30 ->9 2019-06-15T13:45:30 ->19
'yy'	The year, from 00 to 99. More information: The "yy" Custom Format Specifier.	0001-01-01T00:00:00 ->01 0900-01-01T00:00:00 ->00 1900-01-01T00:00:00 ->00 2009-06-15T13:45:30 ->19
'yyy'	The year, with a minimum of three digits. More information: The "yyy" Custom Format Specifier.	0001-01-01T00:00:00 ->001 0900-01-01T00:00:00 ->900 1900-01-01T00:00:00 ->1900 2009-06-15T13:45:30 ->2009
'yyyy'	The year as a four-digit number. More information: The "yyyy" Custom Format Specifier.	0001-01-01T00:00:00 ->0001 0900-01-01T00:00:00 ->0900 1900-01-01T00:00:00 ->1900 2009-06-15T13:45:30 ->2009
'yyyyy'	The year as a five-digit number. More information: The "yyyyy" Custom Format Specifier.	0001-01-01T00:00:00 ->00001 2009-06-15T13:45:30 ->02009
'z'	Hours offset from UTC, with no leading zeros. More information: The "z" Custom Format Specifier.	2009-06-15T13:45:30-07:00 ->-7
'zz'	Hours offset from UTC, with a leading zero for a single-digit value. More information: The "zz" Custom Format Specifier.	2009-06-15T13:45:30-07:00 ->-07
'zzz'	Hours and minutes offset from UTC. More information: The "zzz" Custom Format Specifier.	2009-06-15T13:45:30-07:00 ->-07:00
':'	The time separator. More information: The ":" Custom Format Specifier.	2009-06-15T13:45:30 ->: (en-US) 2009-06-15T13:45:30 ->.(it-IT) 2009-06-15T13:45:30 ->.(yes-JP)
'/'	The date separator. More information: The "/" Custom Format Specifier.	2009-06-15T13:45:30 ->/ (en-US) 2009-06-15T13:45:30 ->.(ar-DZ) 2009-06-15T13:45:30 ->.(tr-TR)
'strings'	Literal string delimiter.	2009-06-15T13:45:30 ("arr:" h:mt) ->arr: 1:45 P
'string'	More information: Character literals.	2009-06-15T13:45:30 ('arr:' h:mt) ->arr: 1:45 P
'%'	Defines the following character as a custom format specifier. More information: Using Single Custom Format Specifiers.	2009-06-15T13:45:30 (%h) ->1
'\'	The escape character. More information: Character literals and Using the Escape Character.	2009-06-15T13:45:30 (h \h) ->1 h
Any other characters	The character is copied to the result string unchanged. More information: Character literals.	2009-06-15T01:45:30 (arr hh:mm t) ->arr 01:45 A

# 10

## frequently asked Questions (FAQ)

### **10.1 License**

#### **10.1.1 How many users can use a license at the same time? the?**

This depends on the offer you choose. Single licenses may only be used by one user at the same time, while a multi-user/terminal server license, for example, may be used by several users at the same time, depending on the offer.

#### **10.1.2 On how many computers can a license be used?**

A license can be installed on multiple computers, but can only be used by one of those computers at the same time. When you start the application and during use, a check is made to see whether the license is already being used on more than one computer. If this is the case, the program will be thrown back into demo mode until the computer is again the only one using the software license. Until then, no progress is lost and the program does not close automatically or anything like that. The only difference is that a watermark appears on the created PDF documents again.

### **10.1.3 When is a license valid and for how long?**

The license usage period begins immediately after purchase! Since the number of days in the year varies (leap year) and is therefore on average 365.24 days, the usage period of days for the 1-year license was set to 367 days and 1101 days for the 3-year license.

1. 3 month license = 94 days
2. 1 year license = 367 days
3. 3 year license = 1101 days

# 11

## References References

- <https://BlockPDF.de/>  
[1] BlockPDF
- [2] MacOS - App Translocation Warning  
<https://web.archive.org/web/20230930025305/https://help.obdev.at/de/littlesnitch4/alert-warnings>
- [3] Point (typography)  
[https://en.wikipedia.org/wiki/Point\\_\(typography\)](https://en.wikipedia.org/wiki/Point_(typography))
- [4] ExcelDataReader  
<https://github.com/ExcelDataReader/ExcelDataReader>
- [5] .NET date formatting <https://docs.microsoft.com/en-us/dotnet/standard/base-types/custom-date-and-time-format-strings>