Barcode Studio

Barcode Designer

Version 17.0

User Manual

24 February 2025

TEC-IT Datenverarbeitung GmbH
Hans-Wagner-Strasse 6
A-4400 Steyr, Austria

t ++43 (0)7252 72720
f ++43 (0)7252 72720 77
office@tec-it.com
www.tec-it.com

1 Content

1.1 Table of Figures

1.2 List of Tables

2 Disclaimer

The actual version of this product (document) is available as is. TEC-IT declines all warranties which go beyond applicable rights. The licensee (or reader) bears all risks that might take place during the use of the system (the documentation). TEC-IT and its contractual partners cannot be penalized for direct and indirect damages or losses (this includes non-restrictive, damages through loss of revenues, constriction in the exercise of business, loss of business information or any kind of commercial loss), which is caused by use or inability to use the product (documentation), even if TEC-IT has been advised of or TEC-IT has pointed out the possibility of such damages.

TEC-IT Datenverarbeitung GmbH
Hans-Wagner-Str. 6

A-4400 Austria
t.: +43 (0)7252 72720
f.: +43 (0)7252 72720 77
www.tec-it.com

3 Introduction

3.1 About

TEC-IT **Barcode Studio** is a tool for creating and printing barcodes and related codes. With **Barcode Studio** you can

- create more than 100 different code symbologies: linear, 2D, postal and GS1 composite codes are supported.
- adjust all code parameters.
- export your codes as raster image files (e.g. for use in your artwork) and as high quality vector images (for graphic design or pre-press applications),
- copy them to the clipboard or print them directly to any printer.
- generate them in batch mode, applying automatically generated, imported, or manually entered data.
- print them as simple labels.

If you have any questions, please contact us.

Address:	Hans-Wagner-Str. 6 AT-4400 Steyr Austria/Europe
Phone:	+43 / (0)7252 / 72 72 0
Fax:	+43 / (0)7252 / 72 72 0 – 77
Email:	office@tec-it.com
Web:	www.tec-it.com

3.2 Supported Operating Systems

- macOS® (11, 12, 13, 14, 15) for Intel and ARM
- Windows 11
- Windows 10 (1809 or later)
- Windows Server 2019
- Windows Server 2022
- Linux/UNIX (please refer to www.tec-it.com or request a build)

► On ARM Macs, you must install *Rosetta 2* to be able to start **Barcode Studio**. The first time you start **Barcode Studio**, macOS® will ask if you want to install it.

► See also https://support.apple.com/en-us/102527.

3.3 Restrictions of the Demo Version

► A demo marker across the code indicates that the demo version is active. The correctness of the codes is not affected.

► To obtain a license key (without the demo marker), please order **Barcode Studio** online at www.tec-it.com/order/default.aspx.

4 Installation

4.1 Install Barcode Studio on Microsoft® Windows

Please follow the steps below.

1. Open the installer by double-clicking it.
2. Decide if you want to install for "*your user"* or for "*all users*".
3. Follow the installation wizard instructions.

- ► An "all users" installation requires administrator privileges.
- ► A "for your user" installation installs **Barcode Studio** to your home directory.

By default, **Barcode Studio** is installed to the following locations:

```
-- installation type: for your user
<user_dir>\AppData\Local\Programs\TEC-IT\BCStudio17 (binary files + documentation)
<user_dir>\AppData\Local\TEC-IT\BCStudio\17.0       (templates, option file, etc.)

-- installation type: for all users (requires administrator privileges)
C:\Program Files\TEC-IT\BCStudio17                  (binary files + documentation)
C:\ProgramData\TEC-IT\BCStudio\17.0                 (templates, option file, etc.)
```

4.2 Install Barcode Studio on macOS® (11 or higher)

For macOS® (version 11 and higher), Barcode Studio is provided as a zipped .pkg file.

Please follow the steps below:

1. Open the ZIP file by double-clicking on it.
2. Also open the contained .pkg file by double-clicking on it.
3. Follow the Setup Wizard instructions.

5 Quick-Start

5.1 Introduction

This chapter will guide you through the most important features of **Barcode Studio** by showing you how to create an EAN13 barcode. For more detailed information about the user interface and the available functions and settings, see the chapters *6* through *10*.

5.2 How to Create an EAN13 Barcode

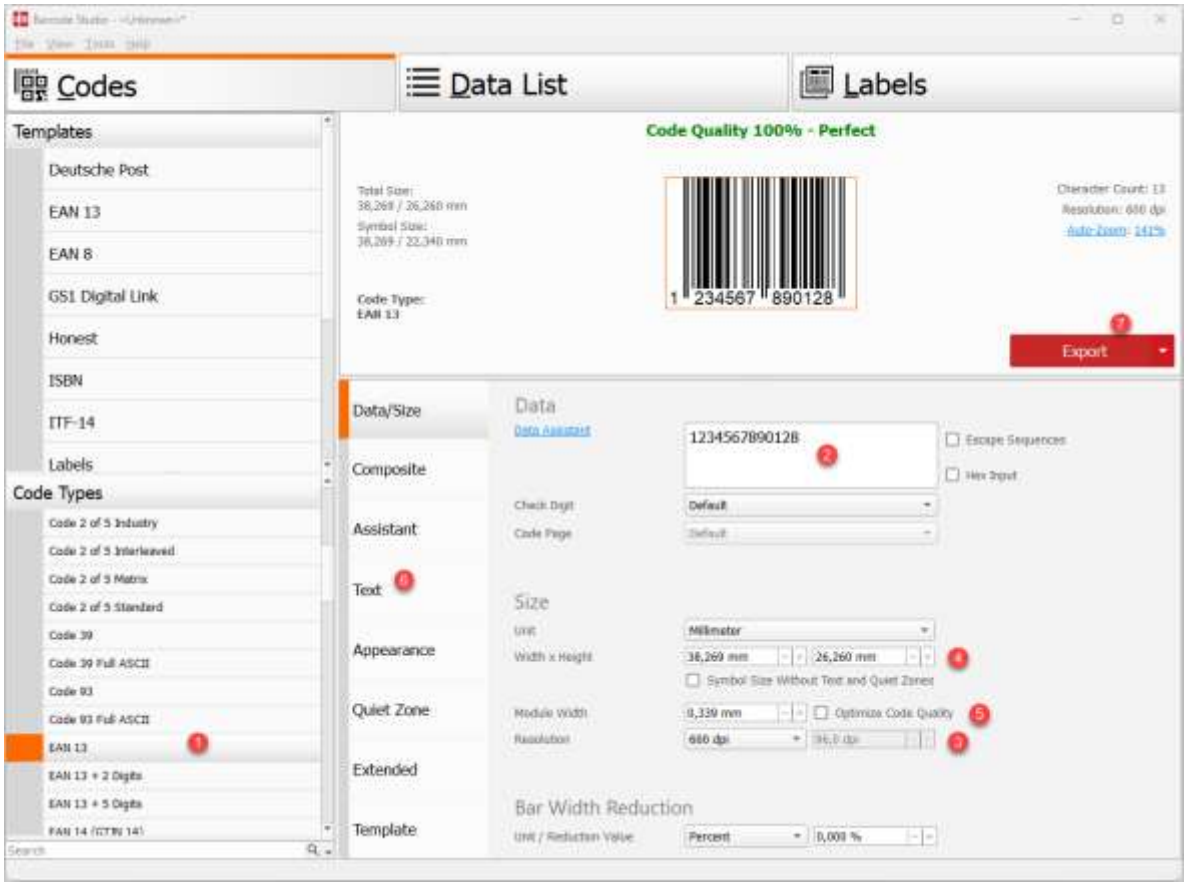
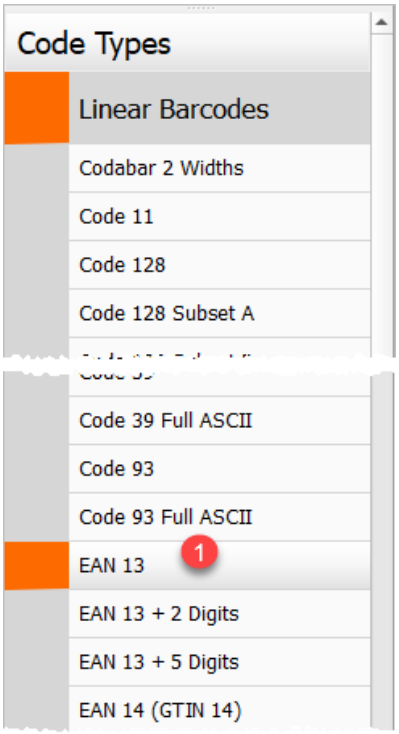


Figure 1: Quick-Start – Barcode Studio Main Window

To create an EAN13 barcode, please follow these steps:

❶ *Select the Code Type (Symbology)* – see section *5.2.1*.
❷ *Enter the Code Data* – see section *5.2.2*.
❸ *Select the Output Resolution* – see section *5.2.3*.
❹ *Specify the Code Dimensions* – see section *5.2.4*.
❺ *Fine-tune the Module Width* – see section *5.2.5*.
❻ *Set Font Style and Size* – see section *5.2.6*.
❼ *Export/Print Codes* – see section *5.4*.

5.2.1 Select the Code Type (Symbology)



Select the code type (= symbology) “EAN 13” from ❶.
Tip: Press *E* to jump to the first type beginning with "E."

5.2.2 Enter the Code Data

The EAN13 symbology requires exactly 12 digits, or 13 digits including the check digit. The default data is "123456789012".

Modify the *Data* as depicted in ❷. Alternatively, use the *Data Assistant* (chapter *7.15*).

You will see the barcode as soon as the data is valid, meaning it contains 12 or 13 digits.

If you enter less than 12 or more than 13 digits, the data will be considered invalid, and you will receive an error message similar to the following:

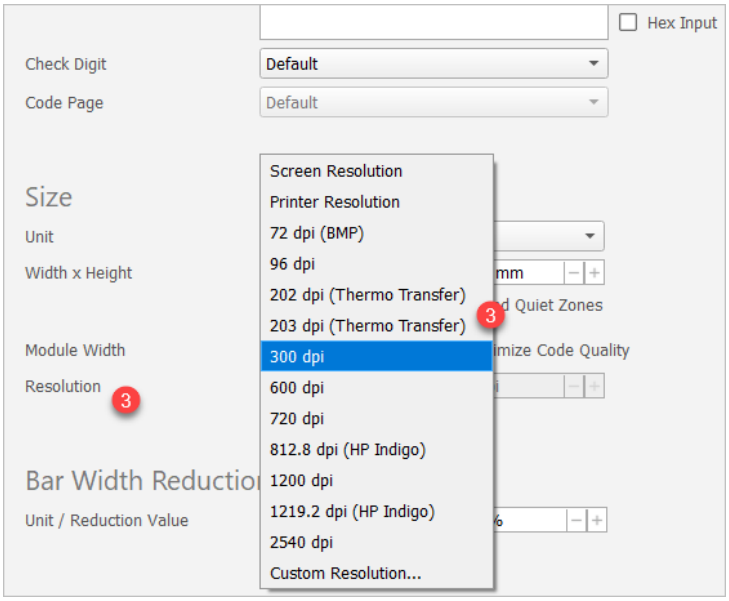
For a list of the most common error messages, see *Appendix B: Error Messages.*

► The 13th digit in the EAN13 code (in this case, "3") is the check digit. It is calculated automatically.

► You can manually enter the correct check digit. Adding an incorrect check digit will prompt an error message.

5.2.3 Select the Output Resolution

- **Barcode Studio** uses the selected resolution to generate code images. To ensure readability, it is essential to choose an appropriate resolution. (For more details, see *5.2.5 Fine-tune the Module Width*.)
- Higher resolutions generally improve code quality. However, it's crucial to select a resolution that matches the output device or image processing software.



Set the output resolution to 300 dpi (❸).

If you plan to print the code on a laser printer, it is recommended that you select a higher resolution (600 or 1200 dpi).

To ***export*** the code ***as an image*** for your website, choose "*Screen Resolution*" or set the DPI to 72 or 96.

- When using code images in prepress applications, avoid scaling or resizing them, as this may cause inaccuracies and distortions, that render the codes unusable. Instead, generate them at the exact resolution and size required.

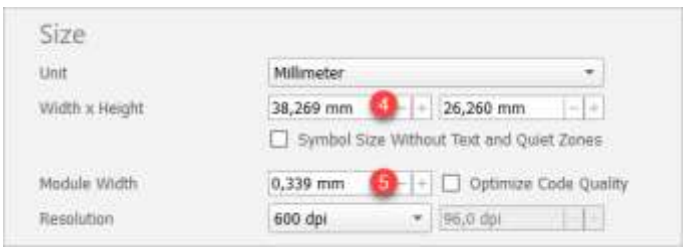
5.2.4 Specify the Code Dimensions

You can modify the code dimensions by changing the **width**, **height**, and **module width**.

Most label and code specifications require specific code dimensions. The following values are common for EAN13:

Magnification factor	Module width [mm]	Width [mm]	Height [mm]
0.80	0.264 (SC0)	29.83	20.73
1.00	**0.330 (SC2)**	**37.29**	**25.91**
1.50	0.495 (SC6)	55.94	38.87
2.00	0.660 (SC9)	74.58	51.82

Table 1: EAN13 Dimensions (not complete)



In this example, we use a zoom factor of 1.00.

Change the width and the height (❹). The appropriate module width (❺) will be calculated automatically.

5.2.5 Fine-tune the Module Width

On the top of the view, you can check the code quality. Ideally, the quality should be **Perfect (100%)**.

We recommend fine-tuning the module width (❺). You can do this in the following ways:

- Check the option *Optimize Code Quality*.
- Select the unit *Pixel* and enter a whole-number value in the *Module Width* field.
- Increase the output resolution.

The first two options both result in perfect code quality.

5.2.6 Set Font Style and Size

Navigate to the *Text* section (see Figure 1 / ❻) and adjust the font properties in the *Select Font* dialog. You can open this dialog by clicking on the *Font* button (❻).

5.3 Code Templates

> ► **Barcode Studio**'s predefined templates already comply with your coding standards. They help save time during optional barcode certification.

To create an EAN 13 barcode using the default settings, follow these steps:

1. Select "EAN 13 / SC2 (100%)..." from the template list.
2. *Enter the Code Data* – see section *5.2.2*.
3. Adjust any other settings as needed.
4. *Export/Print Codes* – see section *5.4*.

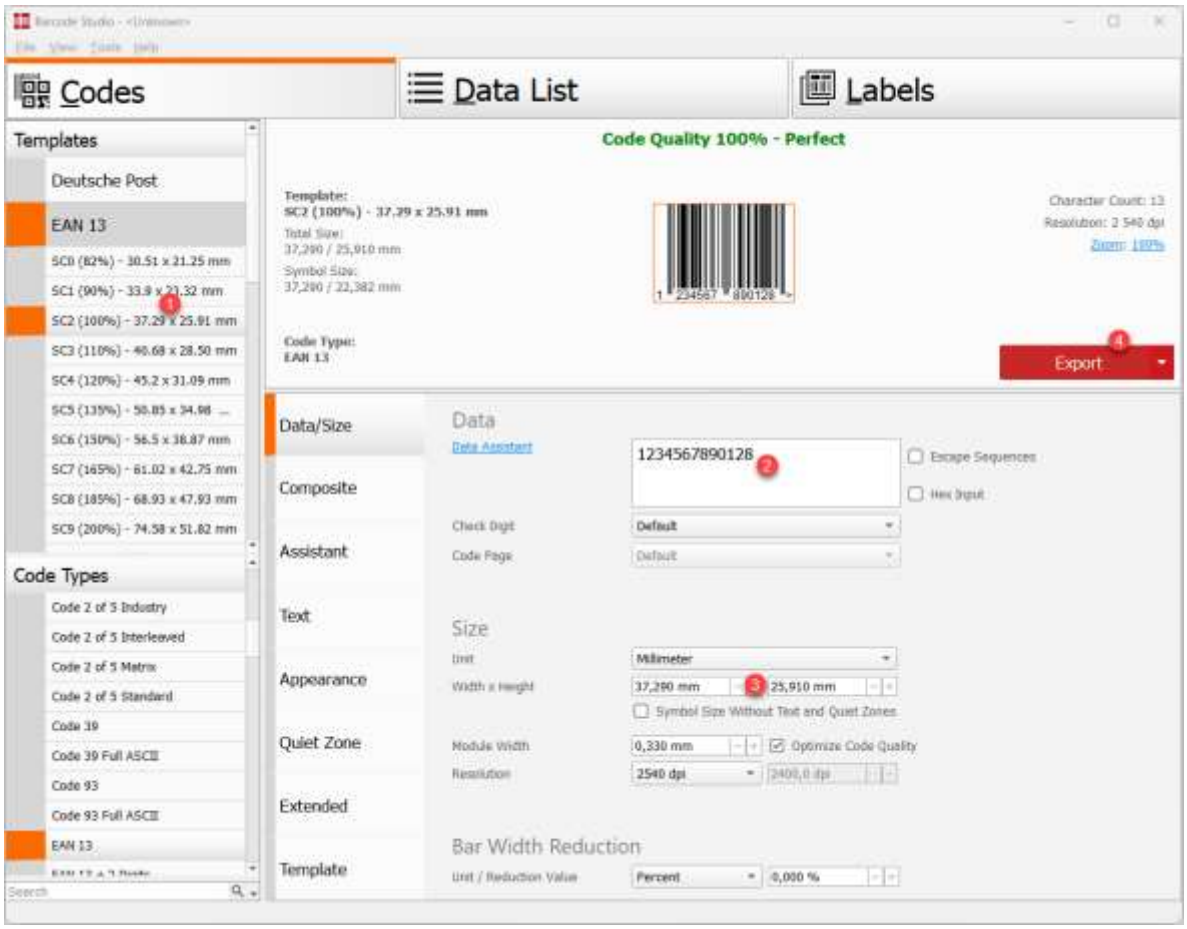
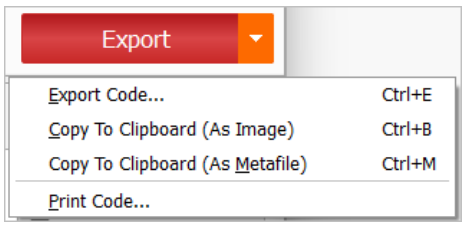


Figure 2: Quick-Start with Templates – Barcode Studio Main Window

5.4 Export/Print Codes



You can share the barcode in the following ways (see chapter *6 Generate Codes – Export and Print*):

- Export to File
  Generate an image file containing the code. **Barcode Studio** supports a variety of raster and vector image formats.
- Copy to Clipboard
  Copy the code to the clipboard and paste it into any application, such as Microsoft® Word or Microsoft® Excel.
- Print to a printer
  Print the code to the selected printer including information about code settings and quality.

Barcode Studio also supports:

- Export a series of codes from the data list:
  Export a series of codes as images from the data list (see chapter *8.5 Export Codes*).
- Print a series of codes as labels:
  Print a series of on selected predefined or custom label sheets (see chapter *9 Labels*).

6 Generate Codes – Export and Print

This chapter explains and gives hints how to export and print codes created with **Barcode Studio**.

6.1 General

You can share the generated codes with other applications in the following ways:

- Export a single code as an image file.
- Copy a single code to the clipboard as an *image* and paste it into your application.
- Copy a single code to the clipboard as a *metafile* and paste it into your application. (Available on Microsoft® Windows only.)
- Print a single code, including type, size, and quality information.
- Export a series of codes from the data list as image files.
- Print a series of codes from the data list as labels.
- Export one or multiple barcodes as image files using the command line.

Before exporting, adjust the data, code size, and any other necessary settings. Be sure to review the notes on resolution and readability (chapter *6.2 Resolution and Readability*, see also chapter *7.8.2 Size*).

6.2 Resolution and Readability

- ► Avoid resizing exported code images using image editing software (such as Photoshop®), as this may affect quality and readability.
- ► Whenever possible, allow **Barcode Studio** to generate the codes in the required sizes and resolutions to avoid the need for resizing them later.

6.2.1 Code Quality

When exporting a code to a raster image format (BMP, GIF, JPG, PNG, TIF, ...), it may be necessary to reduce the resolution to a lower graphic pixel resolution. This reduction can introduce rounding errors, resulting in varying module widths and potentially reduced readability.

Monitor the status information (see chapter *7 Barcode Studio User Interface*, ❹) located at the top right of the code preview. This information indicates the quality (0 to 100%) and the expected readability of the code. Code quality is affected by factors such as output resolution, size, and content.

6.2.2 Module Width

The module width is very important for the readability of a code.

- The module width should not be too small. In practice, for most linear barcodes, the module width should **not be less than 0.19 mm**. Larger module widths generally improve code readability, as long as the total size of the code remains within the recommended limits.
- The module width should be an integer multiple of a pixel. To ensure this, enable the "*Optimize Code Quality*" option.

6.3 Export to an Image File

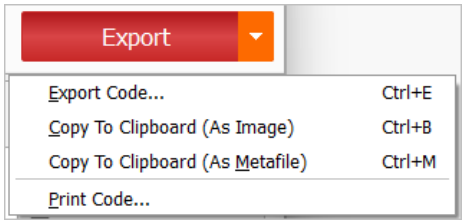


Figure 3: Export button in Codes view with dropdown menu

To export a single code as an image file, navigate to the *Codes* page and either click the *Export* button in the code preview or select *Export Code…* from the button's drop-down menu.

In the *Export Code* dialog, specify a filename, choose the desired image file type, e.g., 'Bitmap (*.bmp)', and then click the *Save* button to confirm.

- The exported code symbol matches exactly with the preview in **Barcode Studio** (taking into account the zoom factor).
- See *Appendix C: Image Types* for a complete list of available image formats.
- It is not recommended to resize images exported in raster format.

6.3.1 Print Settings Dialog Box

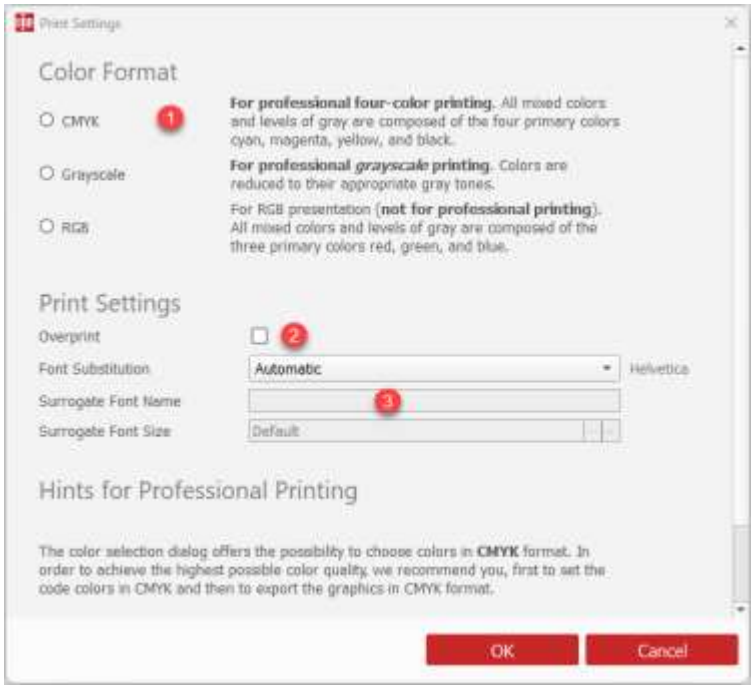


Figure 4: Print Settings Dialog Box

If you select any of the vector-based image formats EPS, PDF, or AI, you will need to specify the following settings. Please note that not all settings are available for all image formats.

- Color Format ❶
- Overprint ❷ (only EPS and PDF)
- Font Substitution ❸ (only EPS and AI)

.

6.3.1.1 Color Format

The color format (❶) defines how the colors are composed. For professional printing, it is recommended to select the *CMYK* or *Grayscale* format (see the explanations provided in the view).

► For professional printing, it is recommended to select the CMYK color format for the colors defined in **Barcode Studio**. When exporting, choose either CMYK or Grayscale as the export format in the dialog box.

6.3.1.2 Overprinting

Decide whether or not to enable *Overprinting* (❷). This setting determines how overlapping colors are handled.

6.3.1.3 Font Substitution

Font Substitution (❸) is applied to EPS and AI files. This is an advanced option and should only be modified if absolutely necessary.

Font Substitution	Description
None	The font is not replaced unless it is invalid.
Automatic (Default)	The font will be replaced with an appropriate alternative if necessary.
Fixed Font	The font specified in 'Surrogate Font Name' will replace the original font during export. The 'Surrogate Font Size' will also replace the original font size, unless it is left at the default setting.

Table 2: Font Substitution

6.4 Copy to Clipboard (As Image)

To copy a single code as an image to the clipboard, choose *Copy to Clipboard (As Image)* from the dropdown menu of the *Export* button.

Paste the copied code into your target application (e.g., Microsoft® Word).

► Resizing code images can lead to a loss of quality, or even worse, unreadable codes.

6.5 Copy to Clipboard (As Metafile)

To copy a single code as a vector-based metafile (EMF) to the clipboard, choose *Copy to Clipboard (As Metafile)* from the dropdown menu of the *Export* button.

Paste the copied code into your target application (e.g., Microsoft® Word).

► Metafiles can only be used in the Microsoft® Windows environment.
► Because of its vector-based format, you can resize the imported image within your target application without losing readability.
► The resulting codes may differ in size from the preview in **Barcode Studio**, depending on the chosen resolution and the target application.

6.6 Print a Code

To print a single code to a printer, choose *Print Code…* from the dropdown menu of the *Export* button.

In the *Print* dialog, choose the printer, adjust the print settings as needed, and then click *Print* to proceed. The printed page will include the code along with additional information such as code type, size, resolution, and quality.

6.7 Export a Series of Codes from the Data List

Figure 5: Export button in DataList view with dropdown menu

To export a series of codes from the Data List, follow these steps (see also chapter *8.5 Export*):

1. Navigate to the *Data List* page.
2. Add data to the data list.
3. In the Code Preview section, click the Export button or choose *Export Codes...* from the dropdown menu.
4. Adjust the export settings in the *Export Codes* dialog.
5. Confirm the settings by clicking the *Ok* button.

You can also export the data list as CSV file by selecting *Export Data...* from the button's dropdown menu (see *8.6 Export Data*).

6.8 Print Labels

Figure 6: Print button in Labels view with dropdown menu

To print a series of codes as labels, follow these steps (see also chapter *9 Label*):

1. Navigate to the *Data List* page.
2. Add data to the data list.
3. Proceed to the *Labels* page.
4. In the *Labels Preview*, click the *Print* button or select *Print Labels...* from the dropdown menu of the button.
5. In the *Print* dialog, choose the printer, adjust the print settings as needed, and then click *Print* to confirm.
6. The data will be validated, and the labels will be printed one page at a time.

- The output resolution and code quality can be influenced by the printer settings.
- Inkjet printers generally produce better results with **bar width reduction** (see chapter *7.8.3 Bar Width Reduction*).

6.9 Create Codes via Command Line

Barcode Studio allows users to generate single barcodes or batches of barcodes from the command line and export them as image files. Data input and output filenames can be specified using text files.

See *Appendix D: Command Line Parameters* for a description of the command line parameters.

7 Barcode Studio User Interface

7.1 Overview

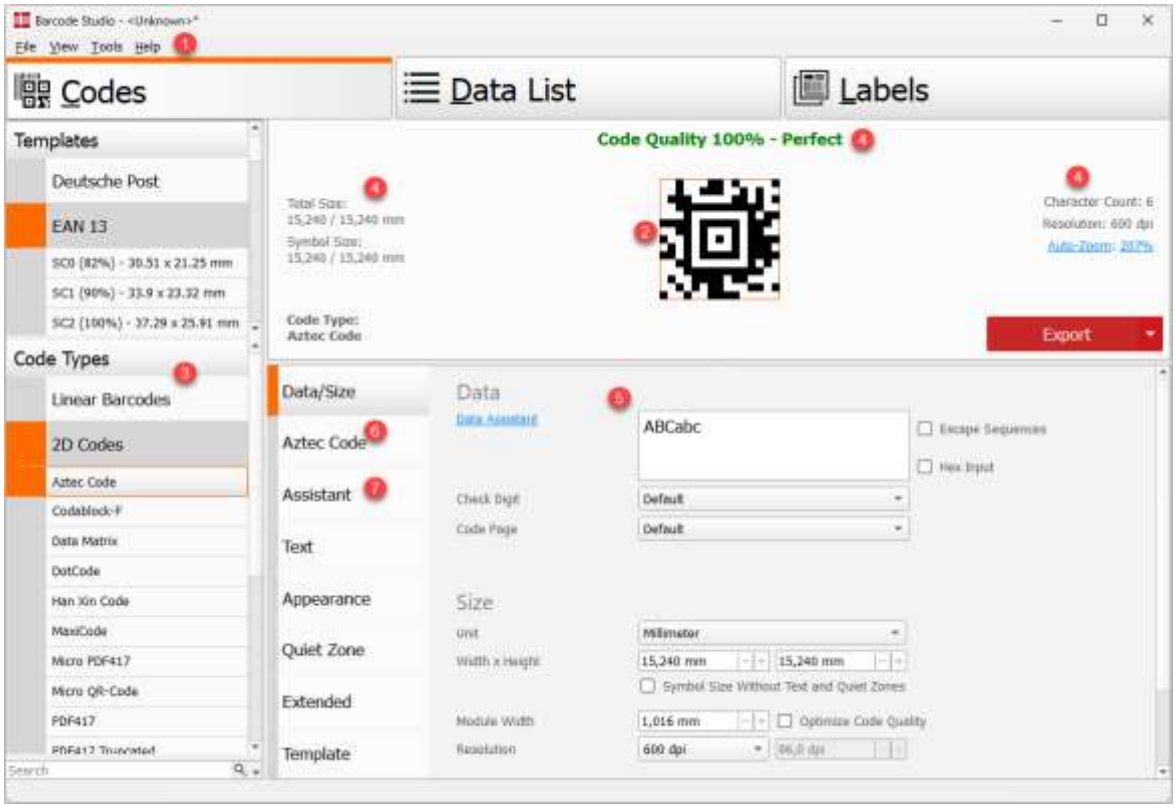


Figure 7: Main View

The main window is divided into the following sections:

❶ *Menu* – see chapter *7.2*.
❷ *Code View* – see *7.4*.
❸ *Templates and Code Types* – see chapter *7.5*.
❹ *Code Status* – see chapter *7.6*.
❺ *Settings Pages* – see chapters *7.7* ff.
❻ *Special Option* (available for code types with special options) – see *7.14*.
❼ *Data Assistant* (available for code types that support applications) – see *7.15*.

7.2 Menu

The menu gives you access to the following functions:

7.2.1 File

New	Create a new document (with initial code settings). Shortcut: *Ctrl+N*
Open…	Open an existing settings file (*.bc). Shortcut: *Ctrl+O*
Save Settings **Save Settings As…**	Save the current settings (symbology, dimensions, module width, etc.) into a file. Settings files have the extension **.bc**. Shortcut: *Ctrl+S (for Save)*

Save As Template…	Save the current settings as a template file in the *Templates* folder.
Print Code…	Print the code and the most important settings (size, resolution, etc.).
Exit	Exit **Barcode Studio**. Shortcut: *Alt+F4*

7.2.2 View

Zoom In	Increase the zoom factor in the preview. Shortcut: *Ctrl++*
Zoom Out	Decrease the zoom factor in the preview. Shortcut: *Ctrl+-*
Reset Zoom	Turn off zoom. The code is shown in its original size.
Zoom Dialog	Open the zoom dialog box.
	Hint: The zoom does not change the real code size, only the size in the preview.

7.2.3 Tools

Options…	Open the *Options* dialog box (see Chapter *10 Options*).
Refresh Templates	Refresh the *Templates* list. Shortcut: *Ctrl+R*

7.2.4 Help

Help	Open the documentation. Shortcut: *F1*
Command Line Usage	Open a dialog box that provides instructions on how to use the command line parameters.
Check For Update…	Check for updates for **Barcode Studio**.
Buy Barcode Studio…	Open the TEC-IT order form in the web browser.
License...	Open the license and activation dialog box.
Activation Key Management....	Check whether a license of a former version of Barcode Studio may be updated.
About Barcode Studio	Show information about the application, version number and copyright.
Online resources	
Visit tec-it.com	Open the TEC-IT homepage www.tec-it.com.
Barcode Reference	Open the barcode reference document.
Google Reviews	Add and show Google reviews for TEC-IT.
YouTube Videos	Open YouTube videos for **Barcode Studio**.
X (Twitter)	Open the TEC-IT page on X (Twitter).
More from TEC-IT	
	Learn more about TEC-IT and TEC-IT software products.

7.3 Dark and Light Mode

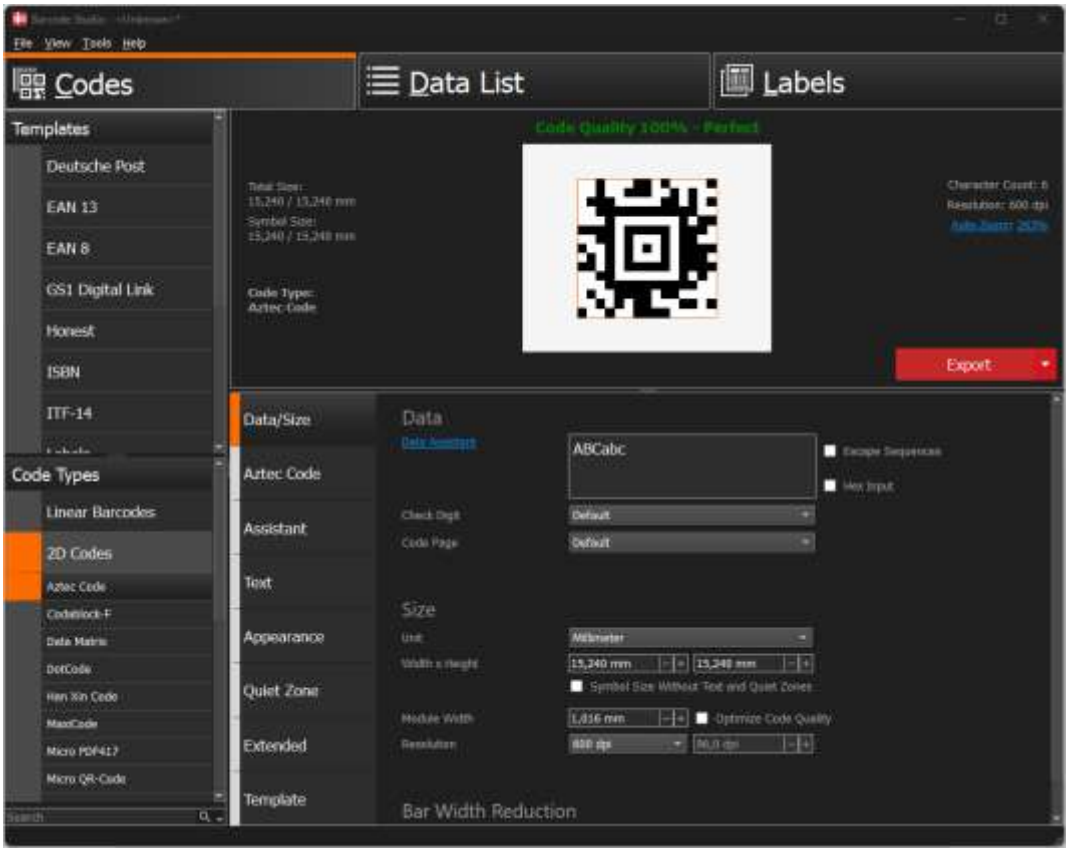


Figure 8: Dark Mode

Barcode Studio adapts to the operating system's settings for light and dark mode. If the system is in light mode, **Barcode Studio** will start in light mode. If the system is in dark mode, **Barcode Studio** will start in dark mode.

> ► Switching between modes is not currently supported.

7.4 Code View

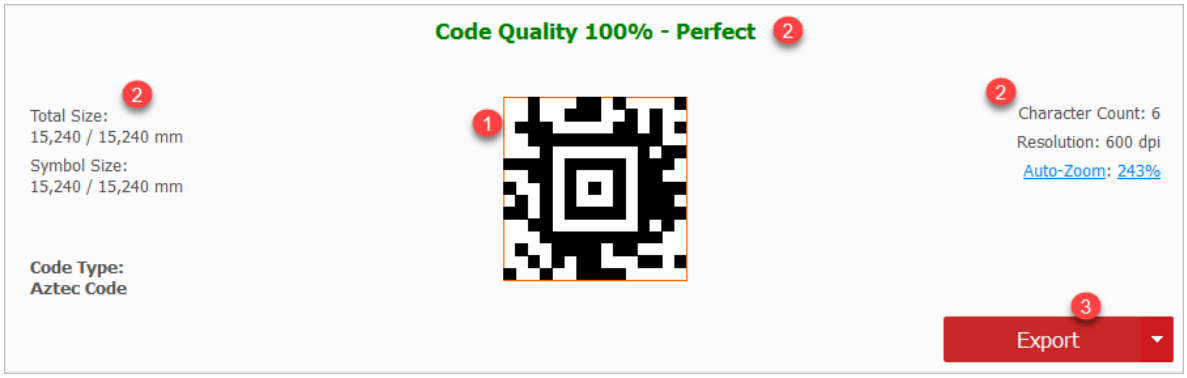
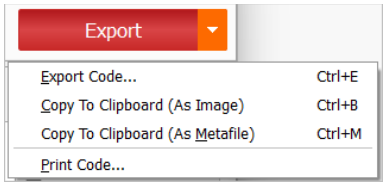


Figure 9: Code View

The *Code View* shows the code (❶) as it will be printed. The *Code Status* (❷, chapter *7.6*) informs the user about code quality, code size, and other attributes.

Use the *Export* button ❸ to export or print single codes, or copy them to the clipboard (see chapter *6 Generate Codes – Export and Print*).



7.5 Templates and Code Types

7.5.1 Code Type

Select the desired code type (= symbology) from the list (see *Figure 7*, ❸). The list is divided into several categories such as Linear Barcodes, 2D Codes, Postal Codes, GS1 Codes, and Health Care Codes.

To learn more about the symbologies, please examine the *Barcode Reference* (*Help ►Online Resources ► Barcode Reference*).

7.5.2 Template

Select the desired template from the *Templates* list. Any previously adjusted settings will be overwritten by the template settings.

You can add your current configuration to the *Templates* list by saving it with *Save As Template…* from the *File* menu or by clicking *Save Template* in the *Codes ►Template* view. To update the *Templates* list choose *Tools ►Refresh Templates*.

Template settings like *Title*, *Category*, and *Keywords* can be changed in the view *Codes ►Template* (see *7.13 Page Template*).

7.5.3 Filter

To easily find the desired code type or template, enter a filter text in the search box located below the *Codes/Templates* list. Filters narrow down the items displayed in the lists based on your search criteria.

The user can switch between two filter options (❸): *Search the list* or *Search by Allowed Content* via popup menu (hint: click on the *Search* button).

7.5.3.1 Search the List

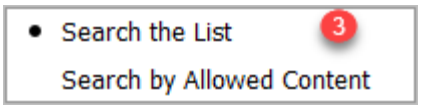
Show only the list entries that contain or are related to the filter text (❶). This option applies to both the Code Type list and the Templates list.

7.5.3.2 Search by Allowed Content

The *Code Type* list is limited to the code types that can encode the filter data (❷).

Figure 10: Search the List

Search by Allowed Content



Search Popup Menu (click on the search button)

7.6 Code Status

Figure 11: Quality Watch

7.6.1 Quality

The higher the quality, the better the readability of the code. Quality depends on the output resolution and the print ratio of the code, and it can be enhanced by adjusting the module width to an appropriate value. To achieve this, either enable the *Optimize Code Quality* option or select *Pixel* as the unit of measurement and set the module width to an integer value (e.g., 1, 2, 3, 25, etc.).

The field *Quality* will report one of the following values:

Tolerance	Quality	Description
Perfect	100%	Optimal output quality.
Acceptable	70-99%	Should be readable by most scanners.
Critical	50-69%	May be readable.
Unreadable!	1-49%	Unreadable in most cases.
Data Loss!	0%	Total or partial loss of information. Unreadable code.

Table 3: Code Quality

7.6.2 Code Template Name

The name of the code template if any is selected.

7.6.3 Total Size

The total size of the code, including text and quiet zones.

7.6.4 Symbol Size

The symbol size of the code without text and quiet zones.

7.6.5 Code Type

The name of the code type.

7.6.6 Application

The name of the application type if any is selected in the Data Assistant.

7.6.7 Character Count

Displays the number of characters in the code data.

7.6.8 Resolution

The current output resolution in dpi (dots per inch).

7.6.9 Zoom/Auto-Zoom

The current zoom factor in percent.

Click *Zoom* respectively *Auto-Zoom* to toggle between auto-zoom and a fixed zoom factor. Click the zoom value to open the *Zoom* dialog box.

7.7 Settings Pages

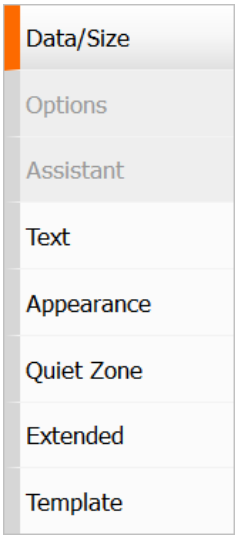


Figure 12: Tab Bar

Use the tab bar to switch between the following settings pages:

- *Page Data/Size* – see chapter *7.8*.
- *Page Text* – see chapter *7.9*.
- *Page Appearance* – see chapter *7.10*.
- *Page Quiet Zone* – see chapter *7.11*.
- *Page Extended* – see chapter *7.12*.
- *Page Template* – see chapter *7.13*.

The *Options* Page is only available for code types that handle special options, such as 2D codes like QR-Code, Date Matrix, … (see chapter *7.14 Special Option*).

The *Data Assistant* simplifies data entry for specific data formats, such as *GS1 Application Identifiers*, *vCard*, *Mobile Tagging*, etc. It is only available for code types that support these formats (see chapter *7.15 Data Assistant*).

7.7.1 Reset Settings

Reset links (❶) appear to the right of the section title when the user changes a value. Clicking *Reset* will revert the field values in the current section back to their default.

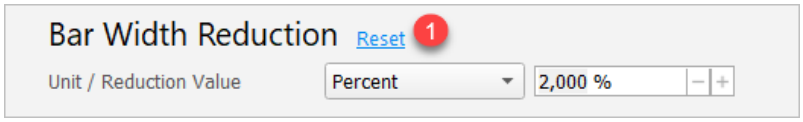


Figure 13: Reset Link

7.8 Page Data/Size

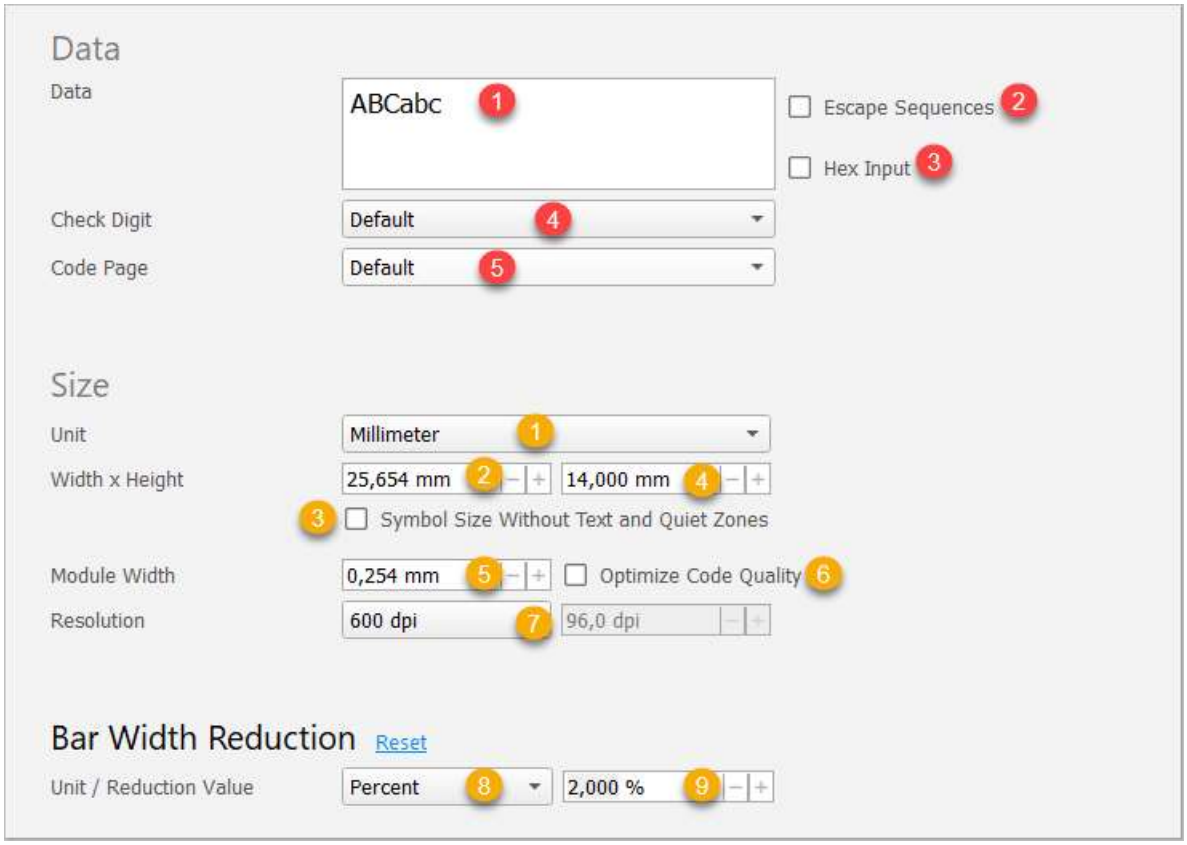


Figure 14: Page Data/Size

7.8.1 Section Data

7.8.1.1 Data

The *Data* ❶ is the code content. **Barcode Studio** provides default data for each code type. Reset the data by clicking the *Reset* link in the *Data* section.

> ► Each code type can encode a specific set of data characters. Some symbologies encode digits. Others encode alphanumeric characters (digits + letters + punctuation). Still others may contain the full ASCII character set (see also *7.5.3.2 Search by Allowed Content*).

7.8.1.1.1 Control Characters

Enter control characters directly in the edit field or use the context menu. To access the context menu, right-click in the edit field, and then select *Add Control Characters*. From the submenu, choose the control character you want to insert.

Since control characters are represented by special character combinations (such as '\F' for FNC1), **Barcode Studio** automatically enables the *Escape sequences* option (see section *7.8.1.2 Escape Sequences*). See also the "Barcode Reference" (*Help ► Barcode Reference*).

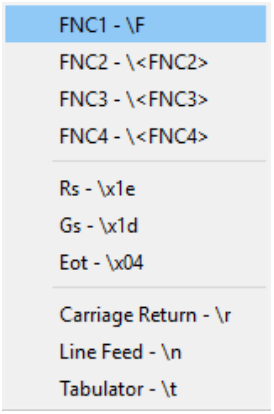


Figure 15: Add Control Character

7.8.1.2 Escape Sequences

The use of escape sequences is necessary when you need to encode control characters such as carriage return or FNC1 into the code.

This option (❷) specifies whether escape sequences (such as "\n") will be translated (default: no). Each escape sequence starts with a backslash ("\") and is followed by one or more characters. For a list of recognized escape sequences, please refer to the "Barcode Reference" document (*Help ► Barcode Reference*).

> ► If escape sequences are enabled, backslashes must also be escaped by replacing them with double backslashes "\\".

7.8.1.3 Hex Input

If this option is checked (❸), the data is treated as hexadecimal data. The setting applies to the *Data* fields in the *Data* section and in the *Composite* section.

Whenever *Hex Input* is enabled, **Barcode Studio** treats manually entered data as well as imported data as a hexadecimal character sequences. Hexadecimal sequences are converted to normal character sequences before they are encoded in the code.

7.8.1.4 Check Digit

The field *Check Digit* (❹) specifies the check digit calculation method. By adding check digits at the end of the usable data, scanners can ensure that the code is read correctly.

Scanners can verify whether the check digit matches the data content and can prompt the device to repeat or reject the scan if they do not match. The calculation methods for check digits are standardized for certain common code types. While using check digits is often optional, it is recommended for specific symbologies and standards.

Default means that the check digit is calculated according to the specification of the selected code type. Change the check digit method only for codes with a variable check digit method or for use with special applications.

See also the "Barcode Reference" (*Help ► Barcode Reference*).

7.8.1.4.1 Check Digit Override

Some code types with a predefined number of usable data characters (such as all *EAN*, *UPC*, *Postnet*, and *GS1 DataBar* codes) include a check digit at a fixed position within the data.

Example:

The EAN-13 code consists of 12 usable digits followed by 1 check digit at the last position.

If you enter 12 digits, the check digit will be calculated and inserted automatically. Alternatively, you can enter all 13 digits manually and override the calculated check digit.

- Note: If you manually enter a check digit along with the data, the barcode will be considered valid only if the entered check digit matches the automatically calculated check digit based on the rest of the data.
- The correctness of a check digit will remain unchecked only if the check digit method is set to *None*. Under normal circumstances, you should avoid using this setting – we recommend relying on **Barcode Studio**'s automatic calculation of the check digit(s).

7.8.1.5 Code Page

By changing the *Code Page* (❺), the user can influence how the input data is interpreted.

For a general overview about code pages, please refer to the “Barcode Reference” (*Help ► Barcode Reference*). See also chapter *7.12.1 Encoding*.

7.8.2 Size

7.8.2.1 Unit

Field ❶ specifies the unit of measurement for the code dimensions. Allowed units are:

- Millimeter
- Inch
- Mils (=1/1000 inch)
- Pixel
- Point
- Micrometer.

Unit	Description
Millimeter	The width and the height of the code in millimeters. The physical width (in pixels) can be calculated as follows: width (pixels) = width (mm) / 25.4 * resolution (dpi).
Inch	The width and the height of the code in inches. The physical width (in pixels) can be calculated as follows: width (pixels) = width (inch) * resolution (dpi).
Mils	The width and the height of the code in mils. The physical width (in pixels) can be calculated as follows: width (pixels) = width (mils) / 1000 * resolution (dpi).
Pixel	The width and the height of the code in pixels. The width and the height are specified in pixels. The resulting dimension on the screen depends on the adjusted resolution (dpi).
Point	The width and the height of the code in points. The physical width (in pixels) can be calculated as follows: width (pixels) = width (pt) / 72 * resolution (dpi).
Micrometer	The width and the height of the code in micrometers. The physical width (in pixels) can be calculated as follows: width (pixels) = width (µm) / 25400 * resolution (dpi)

Table 4: Units

7.8.2.2 Dimensions (Width and Height)

Width (❷) and *Height* (❹) define the size of the code. The user can choose to view and edit the code size with or without text and quiet zone sizes (❸).

7.8.2.3 Module Width

The module width (❺) defines the width of a minimum bar or space segment. **Barcode Studio** provides a default module width for each code type. Reset the size by clicking the *Reset* link in the *Size* section.

- Codes are composed of modules, which are the smallest segments representing bars or spaces. The module width serves as the fundamental unit of measurement for barcodes, as all other widths within the code are based on this value. Adjustments to the module width directly impact the overall code width, and vice versa.

Exceeding the lower limit of the module width results in unreadable codes (e.g., if the module width is smaller than 1 pixel). The minimum limit is determined by the resolution of the screen, printer, or image.

- The module width for codes with variable bar widths should be at least 0.19 mm or greater to ensure proper scanning by code readers.

7.8.2.4 Optimize Code Quality

When the "*Optimize Code Quality*" option (❻) is checked, the module width of the code is set to an integer pixel size (to the nearest lower value). This minimizes or eliminates the pixel deviation.

In other words: The code is printed using only integer pixel sizes. This avoids aliasing effects and ensures optimal scanning quality.

- If the "*Optimize Code Quality*" option is checked, the module width will be adjusted for optimal output quality.
- This setting can be very helpful especially at low resolutions.
- Note: The Code View is only a preview. It only provides optimal quality when the resolution is set to "Screen Resolution". Higher resolutions are not displayed exactly. Only output devices such as printers or bitmaps can handle higher resolutions correctly.

7.8.2.5 Resolution

Select the required output resolution in ❼:

Resolution (DPI)	Description
Screen Resolution	Codes are exported in exactly the same size and resolution as displayed in Barcode Studio. Depending on your operating system and the display settings within the operating system, this resolution may vary. Typical values are 72, 96 or 120 dpi.
Printer Resolution	Use the DPI setting of the currently selected printer.
72 dpi (BMP)	Create codes with 72 dpi.
96 dpi **:** **2540 dpi**	Create codes with the given dpi. Higher dpi values result in larger (and more detailed) image files.
Custom Resolution…	Create codes with a custom resolution.

Table 5: Resolution Settings

- To increase the accuracy of the code symbol, you can increase its resolution.

- Note that the resolution set in **Barcode Studio** must match the resolution of the output device or the target application.
- Image Export: Many applications still display images in screen resolution. If the resolution of the code image is higher, the output size will be larger than in **Barcode Studio**.
  For example: If your screen resolution is 96 dpi and the exported image has a resolution of 300 dpi, it will be displayed about 3 times larger than in **Barcode Studio**.
- The dpi (dots per inch) value can only be saved in certain image formats.
  For instance, GIF files do not support saving resolution (dpi). Only the BMP, JPG, PNG, and TIF formats can retain resolution information.

For detailed information, see *Appendix C: Image Types*.

7.8.3 Bar Width Reduction

Reduce the nominal bar widths by the specified value or factor (❽).

With inkjet printers, ink absorbed by the paper tends to diffuse. By adjusting the bar width reduction, you can compensate for this ink spreading. This feature is also beneficial for laser printers with high toner saturation.

The bar width reduction can be specified in various units of measurement (❾), including in *percent of the module width*, *millimeters*, *inches*, and *mils*. For instance, if the unit is set to percent and the value is set to 20, all bars will be reduced by 20 percent of the module width.

- Caution: When using this feature, it is advisable to perform multiple test scans to ensure the codes can be scanned accurately.
- A common starting value is 15%. However, setting the bar width reduction to more than 50% may render the code unreadable.
- If you require different horizontal and vertical bar width reductions, you can achieve this by adding the following option string to *Extended / Code Options*:

  ```
  DRAW_BarWidthReduction_Values={x=[double] y=[double] unit=[enum]}
  ```

  where ***x*** is the horizontal reduction, ***y*** is the vertical reduction,
  and ***unit*** is the reduction unit (1=pixel, 2=mm, 3=mils, 4=inch).

7.9 Page Text

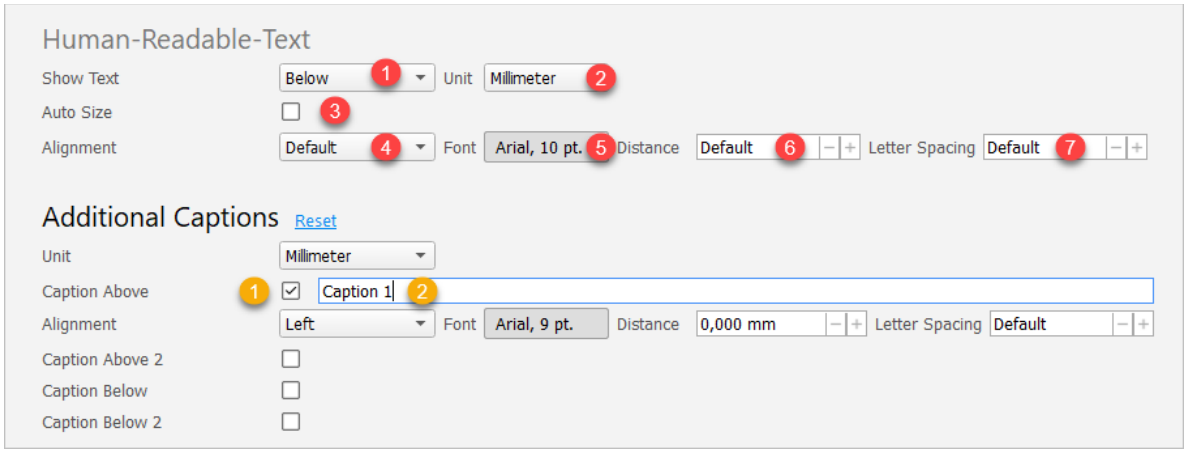


Figure 16: Human-Readable-Text and Captions

7.9.1 Human-Readable-Text

7.9.1.1 Show Text

Select (❶) whether and where to display the human-readable text. Valid values are *No Text*, *Below* (default), and *Above*.

Some code types, particularly 2D symbologies, do not support displaying human-readable text at all. For other code types like EAN-8, EAN-13, UPC-A, or UPC-E, it is not possible to display text *Above* the code.

7.9.1.2 Unit

Define the measure unit (❷) for the fields *Distance* and *Letter Spacing*. Valid units are:

- Modules (=the width of a single module)
- Millimeter
- Inch
- Mils (=1/1000 inch)
- Pixel
- Point
- Micrometer.

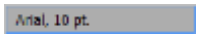
7.9.1.3 Auto Size

When enabled (❸), the font size is automatically calculated based on the module width of the code. If **not** set to *Default*, the *Letter Spacing* and *Text Distance* also adjust proportionally with the font size.

7.9.1.4 Alignment

Adjust the alignment of the human-readable text or caption (❹; options include *Default*, *Left*, *Right*, and *Center*).

7.9.1.5 Font

Click the 

button to modify the font for the human-readable text or caption. The selected font will be displayed in the font box (❺).

7.9.1.6 Distance

Specify the distance (❻) between the human-readable text and the code symbol in the selected unit. For a caption, this value could also represent the distance between the caption and the next inner caption.

7.9.1.7 Letter Spacing

Indicate the spacing (❼) between letters in the human-readable text or caption using the selected unit.

7.9.2 Additional Captions

Activate the caption by checking the box (❶) and inputting the caption text in the edit box (❷). All other settings behave identically to those described above.

7.10 Page Appearance

Figure 17: Color, Rotation, Bearer Bars, Notch Height, …

7.10.1 Color

Click the RGB(0, 0, 0) button corresponding to the color you wish to modify. Next, select the desired color from the dialog box that appears.

Fore Color ❶	The foreground color of the code, denoting the color of the bars.
Back Color ❷	The background color of the code, denoting also the color of the spaces.
Text Color ❸	The color of the “human-readable text”.
Back Style ❹	Mode for drawing the background of the code. The background can be set to either transparent (allowing the background to shine through) or opaque (default, the background is overwritten with the background color).

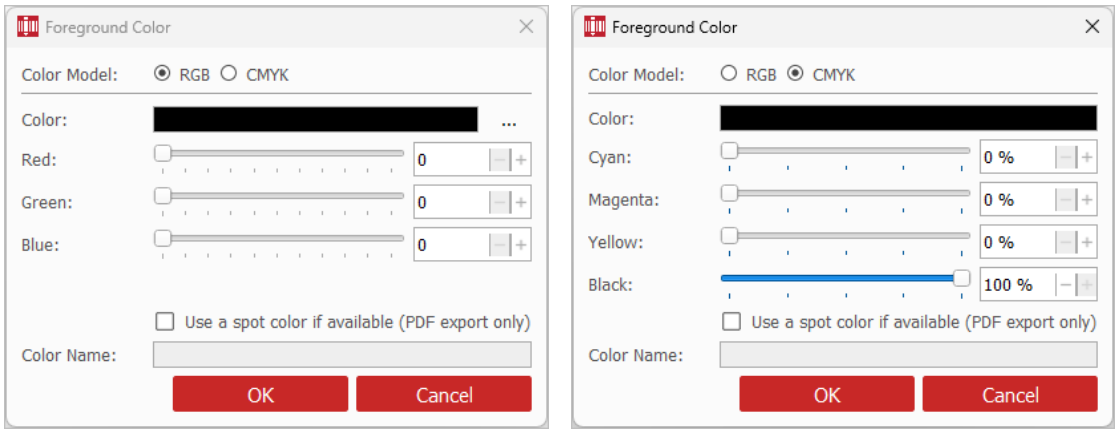


Figure 18: Color Selection Dialog

Select the desired color model. For professional printing, we recommend using **CMYK** or a **named spot color** (available for PDF export only).

7.10.2 Rotation

Specify the code *Rotation* (❺). Possible values are 0°, 90°, 180°, and 270°.

7.10.3 Bearer Bars and Notch Height

7.10.3.1 Bearer Bars

Choose the type of *Bearer Bars* (❷) you wish to employ (*None*, *Top and Bottom*, *Rectangle*, *Top*, *Bottom*). Adjust the unit of measurement (❶) and the width of the bearer bars (❸).

Bearer bars aid in assisting a decoder to detect the complete width of a code, thereby minimizing the likelihood of partial scans (where only a portion of the symbol is decoded).

7.10.3.2 Notch Height

Adjusts the height at which the synchronization bars (❹) extend beyond the standard barcode height.

Synchronization bars are the double lines found on the left, center, and right edges of EAN and UPC codes.

7.10.3.3 Show Quiet Zone Markers

Show or hide the *Quiet Zone Markers* (❺) for EAN and UPC codes.

7.10.4 Embedded Logo

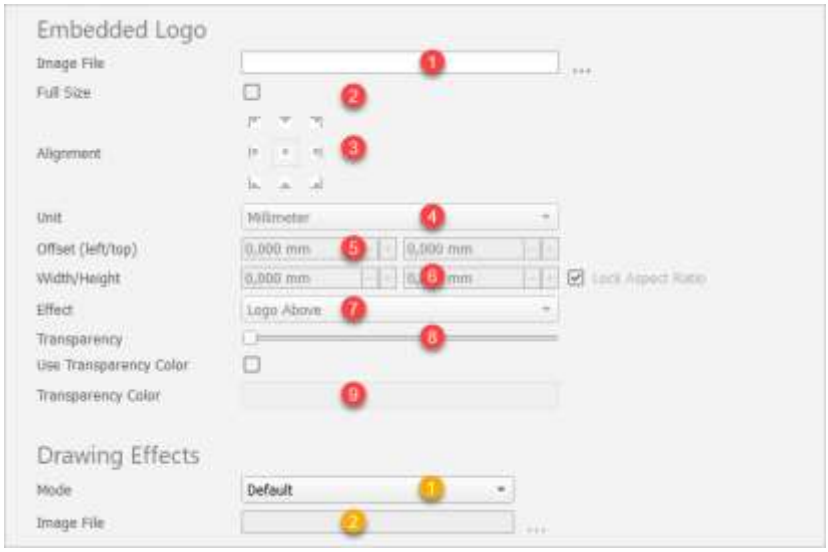


Figure 19: Embedded Logo and Drawing Effects

> ► **Attention:** Embedding logos can **compromise** readability. We strongly recommend that you check the readability of each code.
>
> ► Using logos only makes sense for special applications such as advertising, fun codes, etc., where the readability of the code is not enforced.

7.10.4.1 Image File

The filename of the logo image (❶). The further settings can only be modified if the user selects a valid logo file here.

7.10.4.2 Full Size

If checked (❷), the logo will be stretched to fit the full size of the code. For meaningful results, it's recommended to select an appropriate display mode as well.

7.10.4.3 Alignment

Specify the *Alignment* (❸) and so the position of the logo within the code.

7.10.4.4 Offset

The *Offset* (❺) represents the relative horizontal and vertical distance, measured in specified units, of the logo from the position designated by the alignment.

7.10.4.5 Width/Height, Lock Ratio

This setting defines the width and height (❻) of the logo in specified units (❹). If "*Lock Ratio*" is checked, the ratio between *Width* and *Height* remains constant when the size is adjusted..

7.10.4.6 Effect

Choose a graphical *Effect* (❼) that determines how the code image will incorporated the logo. You can select from a variety of options. Here are a few examples:

- Logo Above → The logo is drawn in front of the code.
- Logo Below → The logo is drawn behind the code.
- Mask → The logo is exclusively drawn over the bars.
- …

7.10.4.7 Transparency

Set the *Transparency* (❽) of the logo in the range between opaque and invisible.

7.10.4.8 Transparency Color

If you wish to utilize a specific *Transparency Color* (❾), select this option and choose a color. Any areas of the logo that match this color will be rendered transparent.

7.10.5 Drawing Effects

7.10.5.1 Mode

> ► **Attention**: For typical use, we highly recommend using **only** the **Default** drawing mode. Utilizing any other mode could potentially **impact** readability.
>
> ► Changing the *Draw Mode* only makes sense for special applications such as advertisements, fun codes, etc. where the readability of the codes is not enforced.

The Mode (❶) determines the shape of the bars. In most cases, bars are depicted as rectangles (exceptions are DotCode, MaxiCode, ...). For specialized applications like advertising, etc., alternative shapes can be selected.

Draw Mode	Description
Default	Standard. The bars are drawn according to the standards. *Caution: For typical use, we highly recommend sticking to this setting.*
Rectangles	Bars are drawn as rectangles.
Circles/Ellipses	Bars are drawn as circles or ellipses, depending on the outline of the bar.
Big Circles/Ellipses	As above, but the circles or ellipses are drawn slightly larger. In matrix codes, the circles overlap, which increases the readability of the code compared to the ***Circles/Ellipses*** setting.
Rounded Rectangles	Bars are drawn as rectangles with rounded corners.
Images	Bars are drawn as images. The name of the image file is specified in the **Image File** field.

Table 6: Draw Modes

7.10.5.2 Image File

The image file (❷) used when the user selects the *Image* drawing method. Images will be drawn instead of the bars.

7.11 Page Quiet Zone

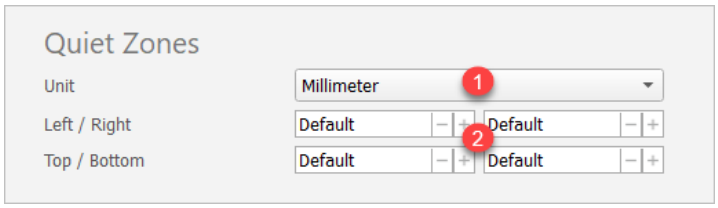


Figure 20: Quiet Zones

Set the sizes of the *Quiet Zones* (❷) around the code. You can specify the sizes in various units (❶) such as *number of modules*, *millimeters*, *inches*, *mils*, *pixels*, *point*, and *micrometers*.

You can configure the quiet zone independently for all four sides of the code.

7.12 Page Extended

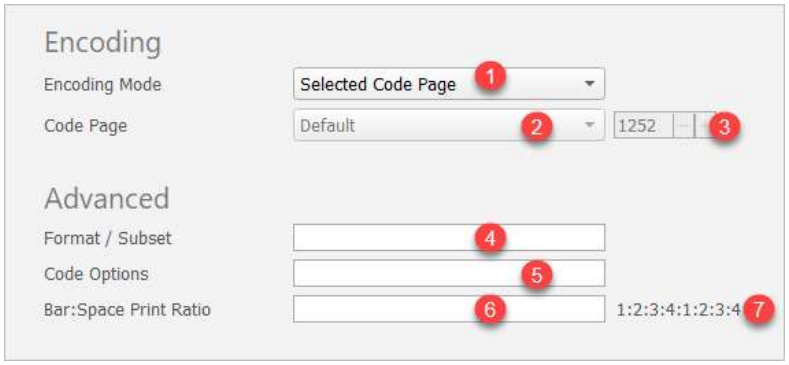


Figure 21: Extended Settings

7.12.1 Encoding

7.12.1.1 Encoding Mode

Through the Encoding Mode (❶), users can modify how the code generator interprets the data. See the “Barcode Reference” (*Help ► Barcode Reference*).

- ► Data is always passed to **Barcode Studio** as a multi-byte UNICODE stream, yet many code types only support single-byte character encoding. Therefore, the data must be converted, and the user can decide how this is executed.
- ► By default, the input data is converted to the selected *Code Page*, but there are situations where it may be beneficial to modify the encoding (see *Table 7: Encoding/Compression*).

You have following possibilities:

Value	Description
Selected Code Page	Convert the input data to the code page selected in *Code Page* (default).
Lo-Bytes only	Consider only the lower bytes of the input characters; the higher bytes are disregarded.
Lo- before Hi-Byte	The data is passed as-is, without any conversion. All bytes are considered, with the lower byte being passed before the higher byte.
Hi- before Lo-Byte	Consider both bytes but reverse the order of the lower and higher byte. This means that the higher byte is encoded before the lower byte.

Table 7: Encoding/Compression

7.12.1.2 Code Page

The *Code Page* (❷) is a sub-setting of the *Encoding Mode* *Selected Code Page*. You can select from a range of predefined code pages (e.g., *ANSI*, *ISO 8559-1 Latin I*, *UTF-8*, *Shift-JIS*, ...) or add the ID of a custom code page by selecting *Custom Encoding…* and entering it in the adjacent input field (❸).

7.12.2 Advanced

7.12.2.1 Format / Subset

The format string (❹) is utilized to format the usable data of the code based on predefined rules, employing replacement symbols to specify the data's structure. Additionally, the format string can insert constant characters into the code data. Control characters enable the modification of subsets for Code 128, EAN 128, and UCC 128, or the definition of the desired start/stop character for CODABAR.

See also the “Barcode Reference” (*Help ► Barcode Reference*).

7.12.2.2 Code Options

The *Code Options* (❺) field enables you to adjust advanced code settings. For a comprehensive list of valid code options, please refer to the description of the `BCSetOptionsA` function at

```
www.tec-it.com/Documentation/TBarCode11 Library Reference/group   options.html.
```

7.12.2.3 Bar:Space Print Ratio

The *Print Ratio* (❻) defines the correlation between the various widths of bars and spaces within a code (distinct from the width-to-height ratio of a symbol!). It is delineated by a sequence of colon-separated values. The sequence commences with 'n' bar widths, succeeded by 'm' space widths, where 'n' and 'm' are contingent on the code type. These values represent multiples of the module width:

- 1 → “Equal to the module width”,
- 2 → “Twice the module width”,
- …

You can employ the *Default Print Ratio* string (❼) as a template for a customized *Print Ratio* value. It indicates the number of distinct bar and space widths utilized for the chosen code type. The precise width of a bar (or space) is determined by multiplying the Print Ratio value by the module width.

For instance: If a barcode element comprises 4 distinct bar widths and 4 different space widths, the print ratio (e.g., for Code 128) might resemble this: *1:2:3:4:1:2:3:4*.

The initial 4 numbers ("1:2:3:4") denote the bar widths, while the final 4 numbers signify the space widths (in this instance, they are uniform). The narrowest bar is "1" module wide, followed by the "2" module-wide bar (twice as wide as the previous one), and so forth.

- Modifying the print ratio is an advanced feature suitable for specific applications. For instance, in the case of the 2OF5 interleaved code, the print ratio can range from 1:2 to 1:3.
- Exercise caution when utilizing this option. Incorrect printing ratios may render codes unreadable.

See also the “Barcode Reference” (*Help ► Barcode Reference*).

7.13 Page Template

Figure 22: Custom Template

Enter the template information (title, category, keywords, …) and save the new template in the template directory either by clicking *Save Template* (❺) or by selecting *File ►Save As Template* from the menu (see also *7.5.2 Template*).

7.13.1 Title and Comment

Enter a template title (❶) and comment (❷). Here you can add any user-relevant information about the template.

7.13.2 Template Category

The *Template Category* (❸) is used to categorize templates in the *Templates* tree.

7.13.3 Keywords

In the field *Keywords* (❹), enter additional information about the template that can be used for filtering (see *7.5.3 Filter*).

7.14 Special Options

7.14.1 Aztec Code

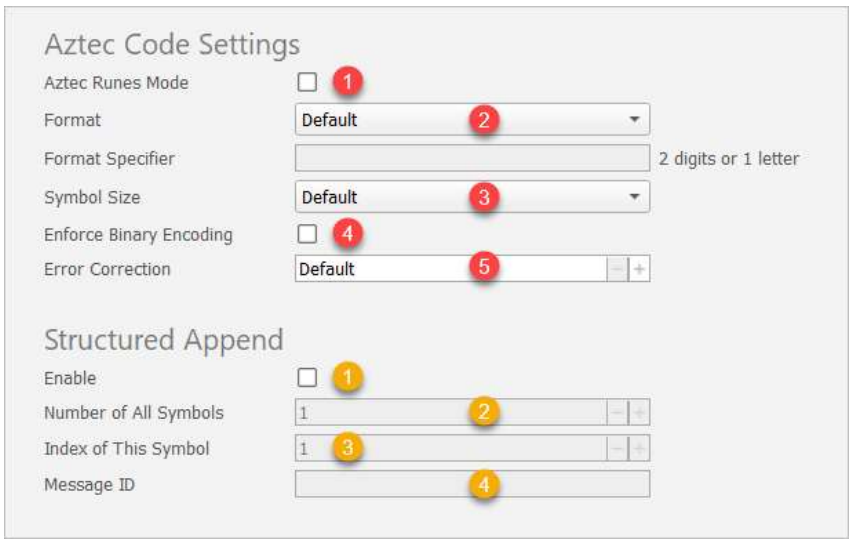


Figure 23: Aztec Code Settings

7.14.1.1 Aztec Runes Mode

Activate or deactivate the *Aztec Runes Mode* (❶).

Aztec Runes are a distinct type of Aztec Code symbol designed to generate small and highly readable markers quickly. They are limited to encoding integers from 0 to 255.

7.14.1.2 Format / Format Specifier

Choose the code *Format* (❷) for the Aztec Code. For the *Industry* format, you must also enter a *Format Specifier*.

Code Format (Enumeration)	Description
Default	Standard format.
GS1/UCC/EAN	Special format defined by GS1/UCC/EAN. Used to encode Application Identifiers. (FNC1 in 1st position).
Industry	Supports peculiar industry formats (FNC1 in second position). If you select the *Industry* format, you must additionally complete the *Format Specifier* field with either 2 digits or 1 letter. This field specifies the industry format of the input data.

Table 8: Aztec Code – Modes

7.14.1.3 Symbol Size

Specify the size of the symbol in a pair of rows and columns (❸). Possible values range from 15 x 15 modules to 151 x 151 modules. The size is automatically calculated based on the encoded data by default.

7.14.1.4 Enforce Binary Encoding

Check the box to enable binary encoding. In *Binary Encoding* mode (❹), the algorithm foregoes attempting to compute the smallest possible symbol, opting instead for binary encoding across all characters. This setting is advised for handling binary input data and for generating codes efficiently.

7.14.1.5 Error Correction

Specify the error correction level (❺) in percent. Enter a value from 1 to 89 percent. The *Default* is equivalent to 23 percent.

7.14.1.6 Structured Append

Structured Append (❶) is used to chain several Aztec Code symbols into one data block. Each symbol in the chain must have a unique index [1..26] (❷). The index defines the order in which the data will be appended after the reading/scanning process. The length of the chain must be entered in the *Number of All Symbols* field (❸). All symbols in the chain must have the same *Message ID* (❹).

7.14.2 Codablock-F

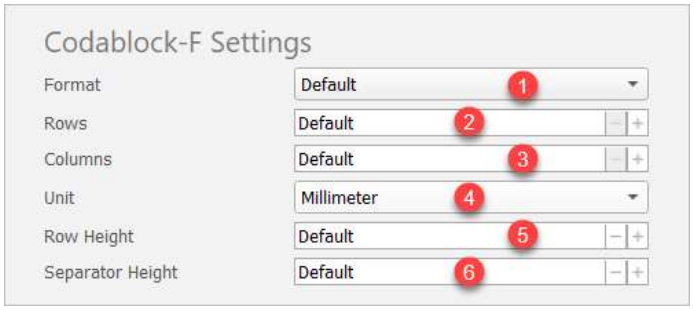


Figure 24: Codablock-F Settings

Codablock-F is a stacked symbology derived from Code 128. Internal checksums are employed to uphold data integrity. The data is organized into rows and columns. Single rows are encoded similarly to Code 128, with the inclusion of a row identifier and a column checksum.

► Use caution when changing these settings. Always perform test scans!

7.14.2.1 Format

Select the code format (❶) of the Codeblock-F code.

Code Format (Enumeration)	Description
Default	Standard format
GS1/UCC/EAN	Special format defined by GS1/UCC/EAN to be used in GS1/UCC/EAN applications.

Table 9: Codablock-F – Modes

7.14.2.2 Rows

The number of rows (❷) must be between 2 and 44. By default, this value is automatically calculated based on the number of characters entered.

7.14.2.3 Columns

The number of columns (❸) in data words must be between 4 and 62, excluding start, stop, and line indicator columns. By default, this value is automatically calculated based on the number of characters entered.

7.14.2.4 Row Height

The height of a row (❺) in the selected units (❹). A fixed row height locks the symbol height to a specific value. By default, the row height is automatically calculated based on the number of characters entered.

7.14.2.5 Separator Height

The width of the row separator line (❻) in the selected units (❹). By default, the line width is equal to the module width, which is automatically calculated based on the symbol size.

7.14.3 Composite

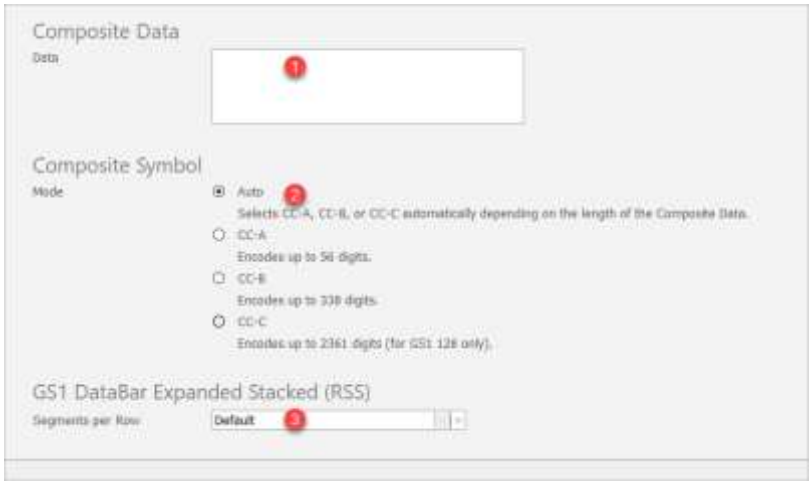


Figure 25: Composite Settings

Composite symbols consist of a main component and a composite component. The main component has one of the following types / these symbologies support composite components:

- EAN8
- EAN13
- GS1-DataBar / RSS14
- GS1 DataBar Stacked / RSS14 Stacked
- GS1 DataBar Stacked Omnidirectional / RSS14 Stacked Omnidirectional
- GS1 DataBar Truncated / RSS14 Truncated
- GS1 DataBar Expanded / RSS Expanded
- GS1 DataBar Expanded Stacked / RSS Expanded Stacked
- GS1 DataBar Limited / RSS Limited
- GS1-128 / UCC/EAN128
- UPC-A
- UPC-E

The composite part has either the type **Micro PDF417** (mode *CC-A*, *CC-B*) or **PDF417** (mode *CC-C*). The composite part is displayed when the composite data is set.

Figure 26: Composite Symbol

7.14.3.1 Data

The content of the composite symbol (❶). The composite part is displayed when the composite data is set.

7.14.3.2 Mode

The *Mode* (❷) determines which composite symbol is displayed. *Auto* is the default.

Auto	Automatically select *CC-A*, *CC-B*, or *CC-C* Symbology depending on the length of the composite data. CC-C is exclusively supported by GS1-128 Composite Symbols.
CC-A	*CC-A* is a variant of the MicroPDF417 Symbol distinguished by a unique combination of row address patterns (RAP). It is the smallest variant of the 2-dimensional composite component. It can encode up to 56 digits of alphanumeric data with 3 to 12 rows and 4 columns.
CC-B	*CC-B* is a variant of the MicroPDF417 Symbol identified by the code word 920. The encoding algorithm automatically selects CC-B when CC-A does not have sufficient capacity (Auto-mode). CC-B can encode up to 338 digits of alphanumeric data in 3 to 12 rows and 2 to 4 columns.
CC-C	The *CC-C* structure is a PDF417 Symbol identified by the internal code word 920 (920 is the first code word after the symbol length indicator). It serves as a 2-dimensional composite component of GS1-128 Composite Symbols. With the largest data capacity among EAN.UCC Composite Symbols, it can encode up to 2361 alphanumeric characters in 3 to 30 rows and up to 30 Data-Error-checking-Code-Columns.

Table 10: Composite Symbols – Modes

7.14.3.3 Segments per Row

Set the number of (graphical data) segments per row in the ***GS1 DataBar Expanded Stacked (RSS)*** Symbol. This parameter influences the width of the code. By *Default* each row contains at least 4 segments.

► This setting is used only by the GS1 DataBar Expanded Stacked (RSS) symbology.

7.14.4 Data Matrix

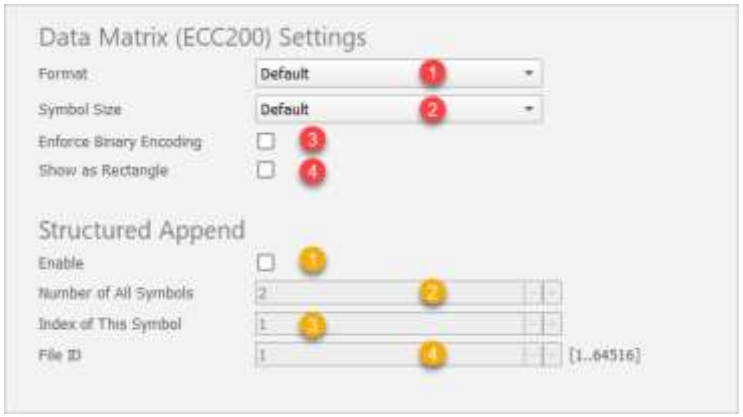


Figure 27: Data Matrix Settings

7.14.4.1 Format

Choose the code *Format* (❶) for the Data Matrix code.

Code Format (Enumeration)	Description
Default	Standard format.
GS1/UCC/EAN	Special format defined by GS1/UCC/EAN. Used to encode Application Identifiers. (FNC1 in 1st position).
Industry	Supports peculiar industry formats (FNC1 in 2nd position).
Format 05	[]>Rs05Gs is encoded at the beginning of the code.
Format 06	[]>Rs06Gs is encoded at the beginning of the code.
Reader Programming	Format utilized for programming code-reading devices.
DP Postmatrix	Special Format defined by „Deutsche Post“. It is used for mailing commercials.

Table 11: Data Matrix – Modes

7.14.4.2 Symbol Size

Specify the size of the symbol in pairs of rows and columns (❷). Possible values range from 10 x 10 modules to 144 x 144 modules for square symbols and from 8 x 18 to 26 x 64. By default, the size is automatically calculated based on the data being encoded.

7.14.4.3 Enforce Binary Encoding

Mark the box to enable binary encoding. In *Binary Encoding* mode (❸), the algorithm foregoes attempting to compute the smallest possible symbol, opting instead for binary encoding across all characters. This setting is advised for handling binary input data and for generating codes efficiently.

7.14.4.4 Show as Rectangle

Mark the box (❹) to present the Data Matrix symbol as a rectangle, or leave it unmarked to display it as a square (default).

7.14.4.5 Structured Append

Structured Append (❶) is used to chain several Data Matrix symbols into one data block. Each symbol in the chain must have a unique index (❷, [1..16]). The index defines the order in which the data will be appended after the reading/scanning process. The length of the chain must be entered in the "Number of All Symbols" field (❸, [2..16]). All symbols in the chain must have the same File ID (❹).

7.14.5 DotCode

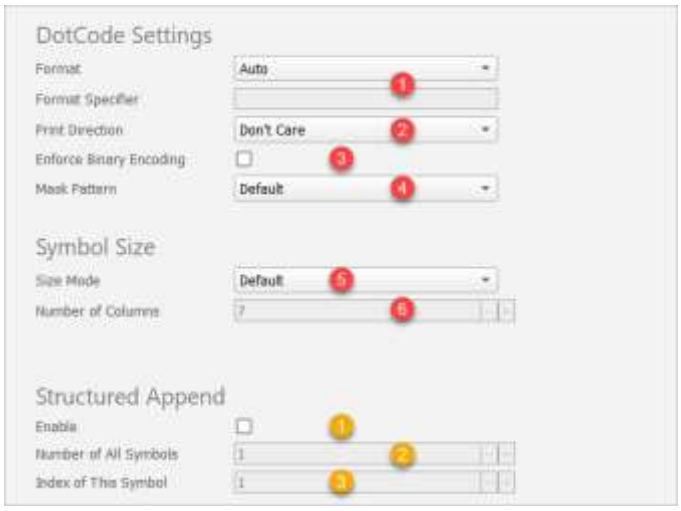


Figure 28: DotCode Settings

7.14.5.1 Format / Format Specifier

Choose the code *Format* (❶) for the DotCode. For the *Industry* format, you must also enter a *Format Specifier*.

Code Format (Enumeration)	Description
Auto	Auto discriminate the format based on the code data. If the data starts with 2 digits, code is in GS1 format, otherwise generic data is assumed.
Generic	Generic data format, which means that data with no special format is used. If the data starts with 2 digits, FNC1 is automatically inserted in first position.
GS1	Special format defined by GS1 that is used for encoding Application Identifiers. Code data must start with 2 digits (the Application Identifier number).
Industry	Supports peculiar industry formats (FNC1 in second position). If you select the *Industry* format, you must additionally complete the *Format Specifier* field with either 2 digits or 1 letter. This field specifies the industry format of the input data.

Macro 05	Data is surrounded by an []>Rs05Gs … RsEot envelop.
Macro 06	Data is surrounded by an []>Rs06Gs … RsEot envelop.
Macro 12	Data is surrounded by an []>Rs12Gs … RsEot envelop.
Custom Macro	Data is surrounded by an []>Rs … Eot envelop. If you choose this option, you also have to fill in the field Application Indicator (2 digits). It determines which envelop ID is to be used. Missing digits and control characters (like Gs, Eot, etc.) have to be added by the user.
Reader Programming	Format utilized for programming code-reading devices.

Table 12: DotCode – Modes

7.14.5.2 Print Direction

The generation of a DotCode symbol can be optimized based on the print direction (❷) of a printer, considering the movement direction of the print head. This feature is beneficial in anticipation of poor print quality. Otherwise, we recommend users to set this field to *Don't Care*.

> ► This setting is applicable only when neither the number of columns nor the number of rows of the symbol is fixed.

7.14.5.3 Enforce Binary Encoding

Mark the box to enable binary encoding. In *Binary Encoding* mode (❸), the algorithm foregoes attempting to compute the smallest possible symbol, opting instead for binary encoding across all characters. This setting is advised for handling binary input data and for generating codes efficiently.

7.14.5.4 Mask Pattern

The mask (❹) enhances code readability. By default, the mask pattern is automatically calculated. If you wish to expedite the calculation process, you can either opt for the *Fast Calculation* mode or directly select a mask value.

Appropriate mask patterns are *Mask 0* to *Mask 3*, and *Mask 0 Prime* to *Mask 3 Prime*. Select the *Prime* Patterns only if your scanner supports *DotCode Rev. 4.0* or later.

7.14.5.5 Size Mode / Size

The combination of the fields *Size Mode* (❺) and *Size* (❻) specifies the number of rows and columns of the code. The name of the *Size* field changes in dependence of the size mode.

Size Mode	Size Field Name	Description
Default	--	Maintain the default aspect ratio between the number of columns and rows, which is set at 3:2 by default.
Ratio Width/Height	Ratio Width/Height	Specify the aspect ratio between the number of columns and rows.
Fixed Width	Number of Columns	Set a fixed number of columns.
Fixed Height	Number of Rows	Set a fixed number of rows.
Fixed Width/Height	Number of Rows and Columns	Set a fixed number of rows and column.

Table 13: DotCode – Symbol Size

7.14.5.6 Structured Append

Structured Append (❶) is used to chain several DotCode symbols into one data block. Each symbol in the chain must have a unique index (❷, [1..35]). The index defines the order in which the data will be appended after the reading/scanning process. The length of the chain must be entered in the "Number of All Symbols" field (❸, [1..35]).

7.14.6 Han Xin Code

Figure 29: Han Xin Code Settings

7.14.6.1 Symbol Size (Version)

Specify the size (=version) of the symbol in pairs of rows and columns (❶). Possible values range from 23 x 23 modules to 189 x 189 modules. By default, the size is automatically calculated based on the data being encoded.

7.14.6.2 Enforce Binary Encoding

Mark the box to enable binary encoding. In *Binary Encoding* mode (❷), the algorithm foregoes attempting to compute the smallest possible symbol, opting instead for binary encoding across all characters. This setting is advised for handling binary input data and for generating codes efficiently.

7.14.6.3 Error Correction Level

Define the error correction level (❸). Choose from the following values:

Error Correction Level (Enumeration)	Description
L1	Lowest level. Data recovery capacity is up to approximately 8%.
L2	Up to 15%.
L3	Up to 23%.
L4	Highest level. Up to 30%.

Table 14: QR-Code – Error Correction Levels

7.14.6.4 Mask Pattern

The mask (❹) enhances code readability. By default, the mask pattern is automatically calculated. If you wish to expedite the calculation process, you can directly select a mask value.

Appropriate mask patterns are *Mask 0* to *Mask 3*.

7.14.7 MaxiCode

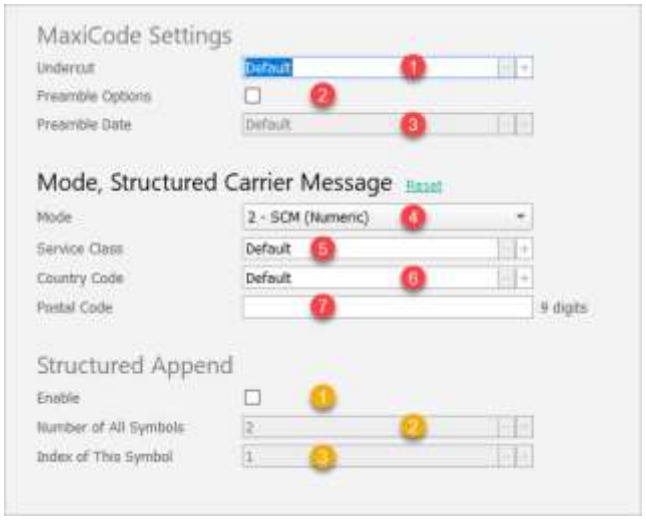


Figure 30: MaxiCode Settings

MaxiCode symbols have a circular center (called a "bull's eye") surrounded by hexagonal dots that contain the code information. MaxiCode supports several different modes and 2 different error correction levels (SEC / EEC = Standard / Enhanced Error Correction). The *Structured Carrier Message* mode is specified by the UPS® package delivery service.

7.14.7.1 Undercut

The undercut (❶) affects the diameter of the hexagonal code elements. According to the AIM standard, it is recommended to use an undercut setting of 75% (*Default*).

7.14.7.2 Preamble

The preamble (❷) is especially used in *Open System Standards*. If enabled, the user can either use the default preamble or customize the *Preamble Date* (❸).

7.14.7.3 Mode

The mode (❹) specifies the structure of the symbol.

Mode (Enumeration)	Description
4 – Standard Symbol	For unstructured numeric and alphanumeric character strings, including standard error correction.
2 – SCM (Numeric)	Structured Carrier Message with digital postal code (up to 9 digits).
3 – SCM (Alphanumeric)	Structured Carrier Message with alphanumeric postal code (up to 6 characters).
5 – Full EEC	Similar to Mode 4, but with extended error correction for increased safety, albeit at the expense of reduced usable data.

Table 15: MaxiCode – Modes

7.14.7.4 Structured Carrier Message

MaxiCode was originally developed by UPS® (United Parcel Service). The *Mode* (❹) "Structured Carrier Message" (Mode 2 and 3) provides special data fields for UPS®-purposes: Service Class (❺), *Country Code* (❻), and *Postal Code* (❼).

Utilize escape sequences to directly insert the date, preamble, service class, country code, and postal code values into the code data. To learn more about this feature, see the “Barcode Reference” (*Help ► Barcode Reference*).

7.14.7.5 Structured Append

Structured Append (❶) is used to chain several MaxiCode symbols into one data block. Each symbol in the chain must have a unique index (❷ – [1..8]). The index defines the order in which the data will be appended after the reading/scanning process. The length of the chain must be entered in the "Number of All Symbols" field (❸ – [2..8]).

7.14.8 PDF417 / Micro PDF417

The code types PDF417 and Micro PDF417 are 2-dimensional stacked symbologies with error correction capabilities. The data is divided into rows and columns of code words. The settings of PDF417 and Micro PDF417 are similar, but some settings differ.

► Use caution when changing these settings. Always perform test scans!

7.14.8.1 PDF417 Settings

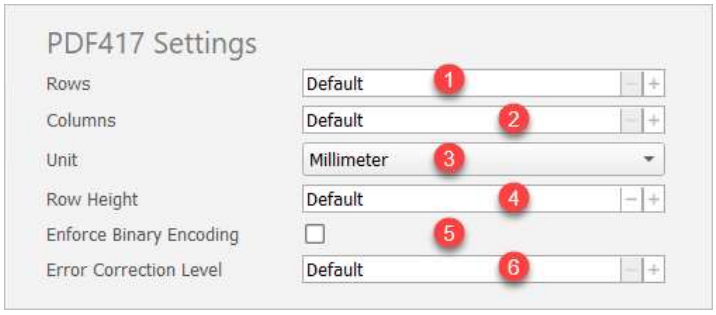


Figure 31: PDF417 Settings

7.14.8.1.1 Rows

The number of rows (❶) must be between of 3 and 90. By default, this value is automatically calculated based on the number of characters entered.

► This feature is **not** available for the **Micro PDF417** symbology.

7.14.8.1.2 Columns

The number of columns (❷) in data words must be between 1 and 30, excluding start, stop, and line indicator columns. By default, this value is automatically calculated based on the number of characters entered.

► This feature is **not** available for the **Micro PDF417** symbology.

7.14.8.1.3 Row Height

The height of a row (❹) in selected units (❸). A fixed row height locks the symbol height to a fixed value. By default, the row height is automatically calculated based on the number of characters entered.

7.14.8.1.4 Enforce Binary Encoding

Mark the box to enable binary encoding. In *Binary Encoding* mode (❺), the algorithm foregoes attempting to compute the smallest possible symbol, opting instead for binary encoding across all characters. This setting is advised for handling binary input data and for generating codes efficiently.

7.14.8.1.5 Error Correction Level

The error correction level (❻) can range from 0 (error detection only) to 8 (maximum correction). By default, the level is automatically calculated according to the number of characters entered (minimum 2, maximum 5).

► This feature is **not** available for the **Micro PDF417** symbology.

7.14.8.2 Micro PDF417 Settings

Figure 32: Micro PDF417 Settings

The *Micro PDF417 Settings* are **only** available for **Micro PDF417** and derived symbologies.

7.14.8.2.1 Format

Select the code format (❶) of the Micro PDF417 code. In most cases the *Normal (Default)* format is the best choice.

► Note: Scanner hardware may not always support all of the listed modes below. Please verify compatibility with your scanner beforehand.

Mode (Enumeration)	Description
Normal (Default)	The input data is parsed. The text, numeric, or binary compression mode is used accordingly to generate the smallest symbol.
GS1/UCC/EAN-128 Emulation	GS1/UCC/EAN-128 Emulation mode. Transmit]C1 or]L3 as symbology prefix. Use compaction for Application Identifier (AI) "01" + 14 digits.
Code128 Emulation	Code-128 Emulation mode. Symbology prefix :]C0 or]L5.
Code128 FNC1 2nd position	Code-128 with FNC1 on second position will be emulated. Prefix:]C2 or]L4.
Linked GS1/UCC/EAN-128	Linked GS1/UCC/EAN-128 emulation. Transmit]C1 or]L3. Indicates that the Micro PDF symbol is linked with a linear symbol (which may be required for a successful scan). Using this mode, the following AIs can be encoded with better compaction in the given order: ▪ Date (AI 11, 13, 15 or 17) + lot number (AI 10) + other AI's (optional). ▪ Date (AI 11, 13, 15 or 17) + serial number (AI 21) + other AI's (optional). ▪ Date (AI 11, 13, 15 or 17) + other AI's (optional). Note: This mode is not used with UCC EAN Composite Symbology (which uses linked symbols, too).
05 Macro	The preamble [(> RS 0 5 GS precedes the encoded data. The postamble RS EOT follows the bar code data.
06 Macro	The preamble [(> RS 0 6 GS precedes the encoded data. The postamble RS EOT follows the bar code data.
CC-A Data Mode	For special applications only: use Base-928 compaction and process input data as byte array.
CC-B Prefix	For special applications only: use binary compaction (Base-900), prefix symbol data with reserved symbology code word.

Table 16: Micro PDF417 – Modes

7.14.8.2.2 Symbol Size (Version)

Specify the size (=version) of the symbol in pairs of rows and columns (❷). Possible values range from 1 x 11 modules to 4 x 44 modules. By default, the size is automatically calculated based on the data being encoded.

7.14.8.2.3 Columns

The number of columns (❸) in data words must be between 1 and 30, excluding start, stop, and line indicator columns. By default, this value is automatically calculated based on the number of characters entered. This setting is disabled when the symbol size is set explicitly.

7.14.8.3 Structured Append / Macro PDF417

Figure 33: Macro PDF417 Settings

7.14.8.3.1 Structured Append

Structured Append (❶) is used to chain several Data Matrix symbols into one data block. Each symbol in the chain must have a unique index (❷, [1..99999]). The index defines the order in which the data will be appended after the reading/scanning process.

Either the length of the chain must be entered in the "Number of All Symbols" field (❸, 1.. 99999]) or the last symbol in the chain must be marked by checking the “Is Last Symbol” field (❹). All symbols in the chain must have the same File ID (❺).

7.14.8.3.2 Extended Settings

The extended settings are all optional. The following fields can be configured:

Filename (alphanumeric) (❶)	Filename (variable length field).
Time stamp (numeric) (❷)	Time stamp. (elapsed time in seconds since January 1, 1970 00:00 GMT)
Sender (alphanumeric) (❸)	Sender (variable length field).
Addressee (alphanumeric) (❹)	Addressee (variable length field).
File size (numeric) (❺)	Total number of bytes encoded (variable length field).
Checksum (numeric) (❻)	16-Bit CRC checksum. (using CCITT-16 polynomial x16 + x12 + x5 + 1 over the entire data)

Table 17: Macro PDF417 – Extended Settings

7.14.9 QR-Code® / QR-Code (JIS) / Micro QR-Code / Swiss QR-Code

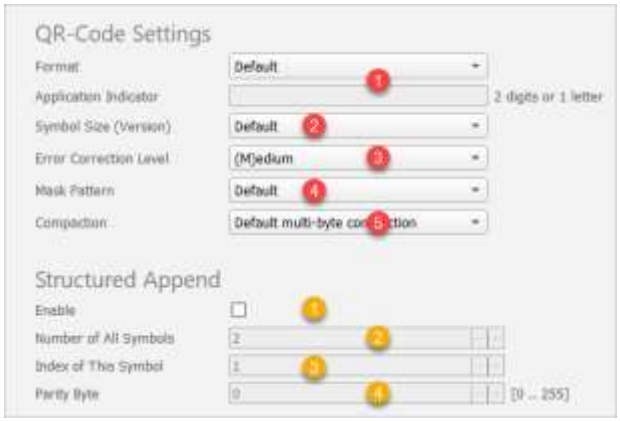


Figure 34: QR-Code Settings

7.14.9.1 Format / Application Indicator

Choose the code *Format* (❶) for the QR-Code. For the *Industry* format, you must also enter an *Application Indicator*.

Format (Enumeration)	Description
Default	Standard format.
GS1/UCC/EAN	Special format defined by GS1/UCC/EAN. Used for encoding Application Identifiers. (FNC1 in 1st position).
Industry	Supports peculiar industry formats (FNC1 in second position). If you select the *Industry* format, you must additionally complete the *Application Indicator* field with either 2 digits or 1 letter. This field specifies the industry format of the input data.

Table 18: QR-Code – Modes

► This feature is **not** available for the **Micro QR-Code** symbology.

7.14.9.2 Symbol Size (Version)

Specify the size (=version) of the symbol in pairs of rows and columns (❷). Possible values range from *(1) 21 x 21* modules to *(40) 177 x 177* modules for QR-Code and from *(M1) 11 x 11* to *(M4) 17 x 17* for Micro QR-Code. By default, the size is automatically calculated based on the data being encoded.

7.14.9.3 Error Correction Level

Define the error correction level (❸). You can choose from the following values:

Error Correction Level (Enumeration)	Description
(L)ow	Lowest level. Data recovery capacity is up to approximately 7%.
(M)edium (default)	Up to 15%.
(Q)uartil	Up to 25%.
(H)igh	Highest level. Up to 30%.

Table 19: QR-Code – Error Correction Levels

► Not all versions (sizes) of Micro QR-Code support every error correction level. If you choose a level that exceeds the capacity of the selected symbol version, **Barcode Studio** will automatically default to the highest feasible level.

7.14.9.4 Mask Pattern

The mask (❹) enhances code readability. By default, the mask pattern is automatically calculated. If you wish to expedite the calculation process, you can directly select a mask value.

Mask Pattern (Enumeration)	Description
Default	The mask pattern is calculated automatically.
0..7 (QR-Code only)	Select *Mask 0* to *Mask 7*. The mask calculation algorithm is quite complex and very resource intensive.
0..3 (Micro QR-Code only)	Select *Mask 0* to *Mask 3*. Compared to QR-Code, the mask calculation algorithm for Micro QR-Code is quite fast and simple.

Table 20: QR-Code – Mask Patterns

7.14.9.5 Compaction

Choose the multi-byte compaction mode (❺). QR-Code offers code compression for different multi-byte character encodings such as Kanji and Chinese. This compression feature aids in generating smaller codes.

Compaction (Enumeration)	Description
Default multi-byte compaction (Default)	Automatically chooses the appropriate compaction method.
No multi-byte compaction	Disables the multi-byte compaction.
Kanji character compaction	Enables the compaction for Kanji characters.
Chinese character compaction	Enables the compaction for Chinese characters.

Table 21: QR-Code – Compaction

► Do not enable compaction for binary data.

7.14.9.6 Structured Append

Structured Append (❶) is used to chain several QR-Code symbols into one data block. Each symbol in the chain must have a unique index [1..26] (❷). The index defines the order in which the data will be appended after the reading/scanning process. The length of the chain must be entered in the "Number of All Symbols" field (❸, [2..26]). All symbols in the chain must have the same *Parity Byte* (❹).

Chained QR-Code symbols are identified by the parity byte. The parity byte must be identical in all symbols and can be calculated using the method „*QR_StructAppParity()*“, which is part of **TBarCode** API.

► This feature is **not** available for the **Micro QR-Code** symbology.

7.15 Data Assistant

The *Data Assistant* provides user-friendly data input masks for various predefined applications and data formats, including contact information for business cards, GS1 Application Identifiers, and more. The availability of specific applications depends on the selected code type.

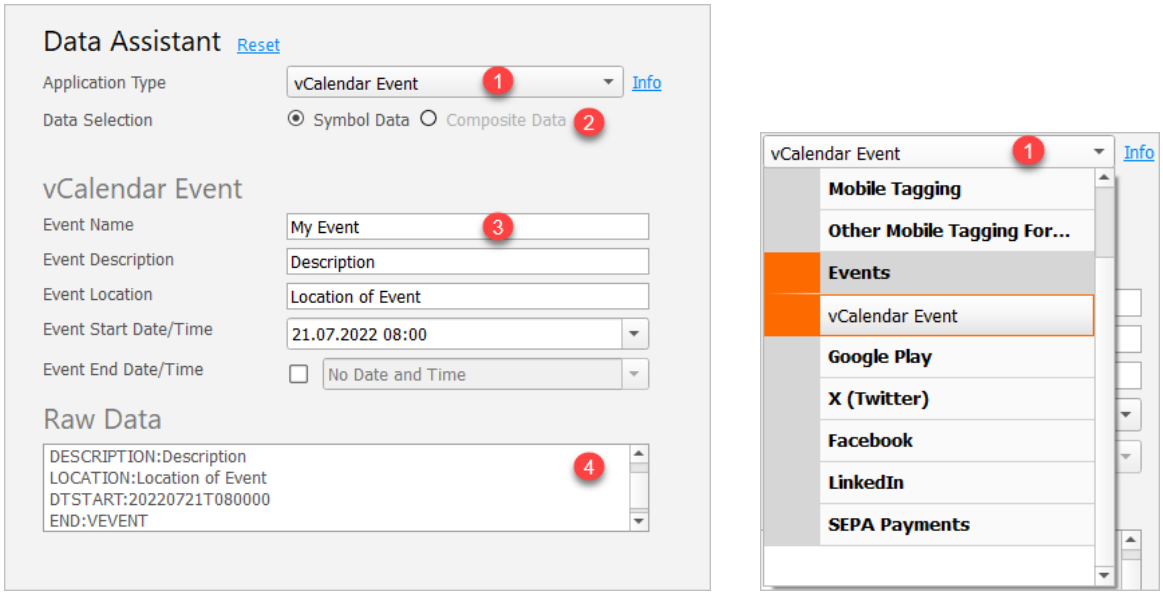


Figure 35: Data Assistant – vCalendar

Select the *Application Type* (❶) and *Data Selection* (❷). The layout of the input mask (❸) varies depending on the chosen application type (e.g., *vCalendar Event*).

Data Selection is only relevant for Composite codes. Users have the option to allocate the data to either the *Symbol Data* field or the *Composite Data* field.

The read-only field *Raw Data* (❹) displays the translation of the user input into the code-compatible raw data, which will be encoded into the barcode.

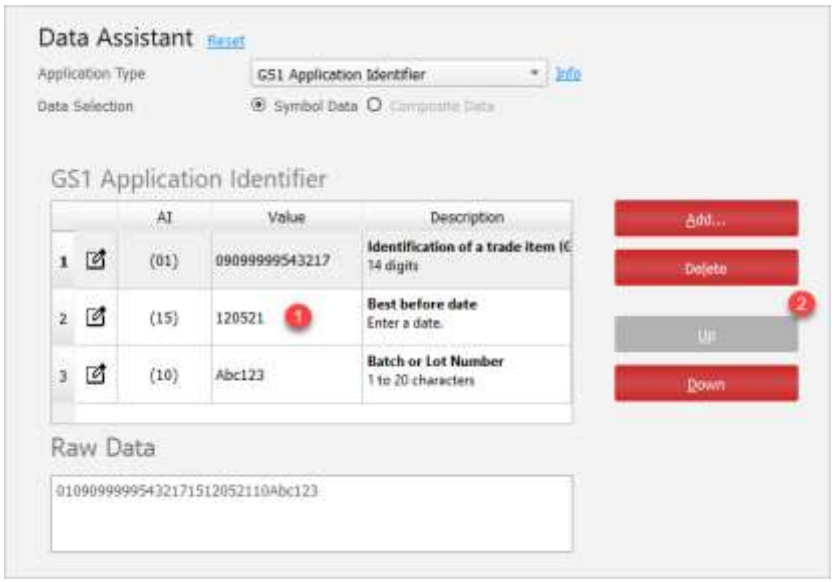


Figure 36: Data Assistant – GS1 Application Identifier

GS1 Application Identifiers (**AI**s) are presented in a list format (❶) with columns for “AI” (AI number), “Value”, and “Description”. The list is interactive and can be managed using buttons (❷) such as *Add*, *Delete*, *Up*, and *Down*. *Values* can be edited directly.

8 Data List

8.1 Overview

The *Data List* enables you to organize a series of code data. First, adjust the code settings in the Codes tab (see chapter *7 Barcode Studio User Interface*). Then, proceed to the *Data List* tab to populate it, either for generating a series of code images (see chapter *8.5 Export*) or for printing code labels (see chapter *9 Label*).

Fill the data list by

- importing data from a file – chapter *8.3 File Import*.
- generating a serial number with a sequence generator – chapter ***Error! Reference source not found. Error! Reference source not found.***.
- editing it manually – chapter *8.2 Data List View*.

8.2 Data List View

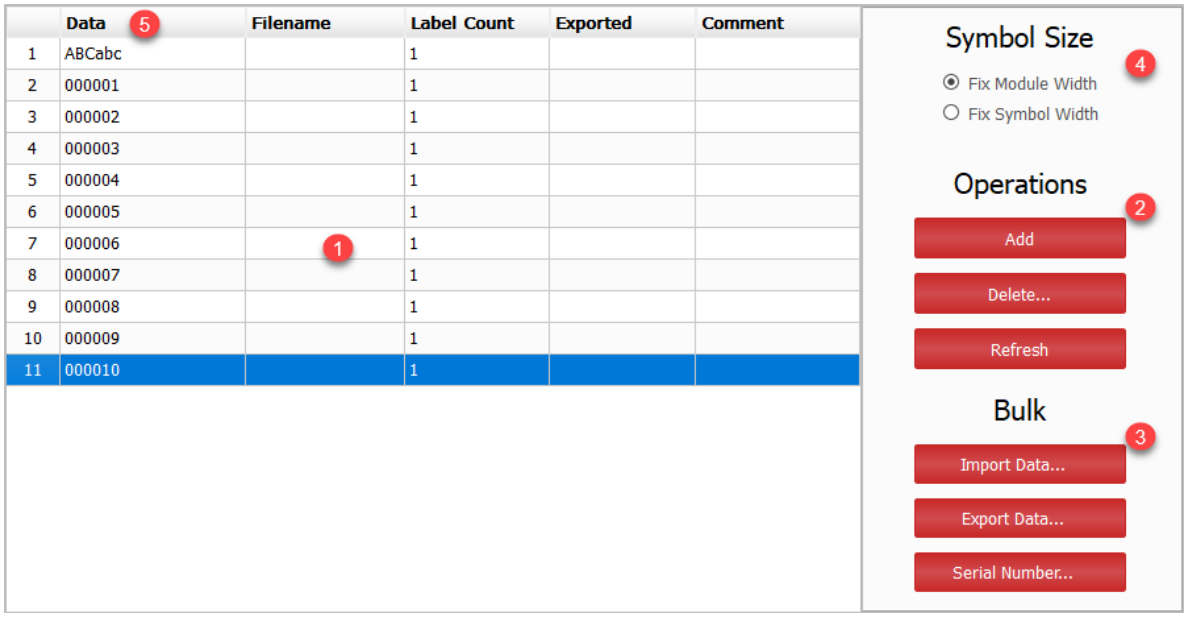


Figure 37: Data List View

The data list (❶) contains code records that can be used to export a series of code images or to print labels.

The data can be inserted, updated or deleted (❷), imported from a text file, automatically generated by the sequence generator, and exported as a CSV file (❸).

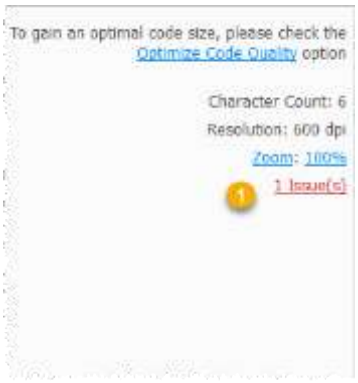
To initiate the image export, click the *Export* button as described in Chapter *6.7 Export a Series of Codes from the Data List*. For additional details on *Label*, please refer to Chapter *9*.

8.2.1 List Content

8.2.1.1 General

The first column contains the **row number**. It makes it easier for the user to navigate through the list.

Records containing errors are highlighted in **red text**. To view the detailed error message, simply select the corresponding data row. The error message will be displayed in the **Code View** above. Erroneous records usually mean that the code cannot be created successfully.



The status view looks like the code page status view, except that the number of issues (❶) is displayed at the bottom of the view.

In the lower right corner of the window, in the status bar, you can see the total number of records and the number of records with an issue (❷).

8.2.1.2 Data

The *Data* column contains the code data necessary for code generation. Values in this column can be edited manually, imported from a file, or generated using the sequence generator.

8.2.1.3 Composite Data

The *Composite Data* column contains the contents of the composite code component.

This column is only visible for symbologies that support composite codes. Values can be edited manually or imported from a file.

8.2.1.4 Captions

The four *Caption* columns contain the caption texts that will be printed above or below the code.

The caption columns are only visible if the corresponding caption fields are enabled in the Settings view (see *7.9.2 Additional Captions*). The columns have the following names: **Caption Above**, **Caption Above 2**, **Caption Below**, **and Caption Below 2**. Values can be edited manually or imported from a file.

8.2.1.5 Filename

The *Filename* column contains the names of the exported image files.

The filenames are typically generated automatically during export but can also be edited manually or imported from a file.

8.2.1.6 Label Count

The *Label Count* column indicates the number of labels printed for each record for each record, but does not affect image export. Values can be edited manually or imported from a file.

8.2.1.7 Exported

The *Exported* column is read-only and displays the timestamp of the last image export. This timestamp is automatically generated during the export process and cannot be modified by the user.

8.2.1.8 Comment

The user may enter comments for the data list in the *Comment* column. Values can be edited manually or imported from a file.

8.2.2 Module Width / Symbol Size

Users must decide whether to *fix* the *Module Width* (default) or the *Symbol Width* (❹).

In the first scenario, codes can vary in size while maintaining a constant module width. Conversely, if the symbol width is fixed, all codes will be uniform in size, although the module width and code quality may differ.

8.2.3 Order

The user can rearrange the list order by clicking on the column headers (❺):

- first click → ascending
- second click → descending
- third click → reset order to original

Clicking on the first (unnamed) column also resets the sort order.

The current order is also utilized for label printing. See also Chapter *9 Label Printing*.

8.2.4 Editing

8.2.4.1 Add New Entries

Add new data entries by clicking the *Add* button. The columns will be filled with the actual settings from the *Codes* page.

8.2.4.2 Edit the Data List

Edit the data directly by **double-clicking** (or by pressing *F2*). The user can modify the contents of the columns *Data*, *Composite Data*, *Caption Above*/*Below* (*1*/*2*), *Filename*, *Label Count*, and *Comment*. The *Data* column is mandatory, all others are optional.

8.2.4.3 Delete Entries

Delete one or more selected list entries by clicking the *Delete* button. Confirm the Dialog Box to proceed.

8.2.4.4 Refresh

The *Refresh* button updates the error status of the whole list.

8.2.4.5 Populate Columns

To populate a single column, select a cell, enter the desired value, confirm the edit by pressing Enter, right-click the cell and choose *Populate Column* from the **popup menu**.

8.3 File Import

Users can import data from a CSV file, configuring the import settings to specify how the data is imported based on the selected application type.

8.3.1 Import Dialog

8.3.1.1 File

Enter or select the name of the import file in ❶. The file must be a text file with character-separated values (*.csv, *.tsv, etc.). Each data row must be separated by line breaks, with columns within each row delimited by a specified separator character.

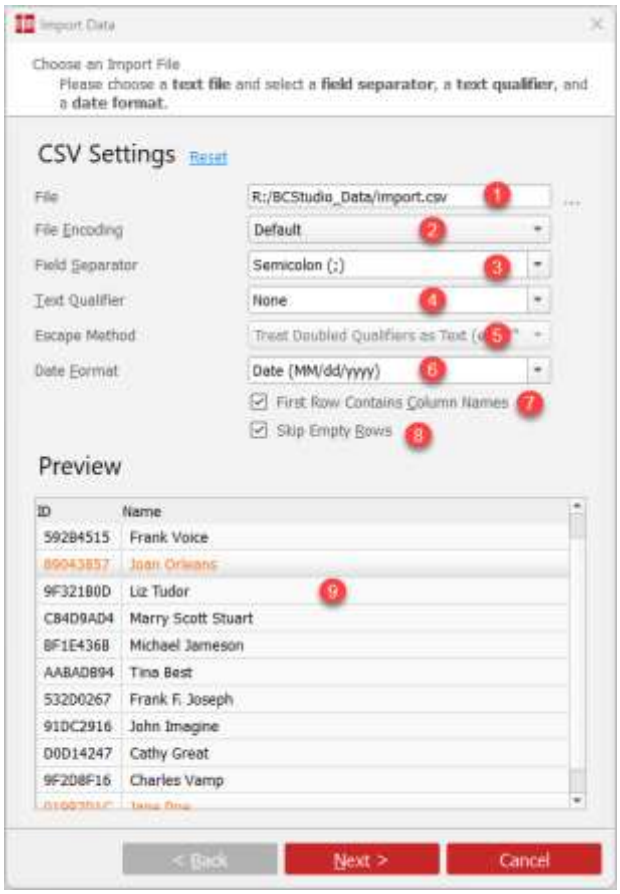


Figure 38: File Import

8.3.1.2 File Encoding

The file import supports 3 types of *File Encodings* (❷): *ANSI*, *UTF-8*, and *UTF-16*. The *Default* for Microsoft® Windows is *ANSI*, the *Default* for Linux and macOS® is *UTF-8*.

8.3.1.3 Field Separator and Text Qualifier

The *Field Separator* (❸) divides rows of data into columns. Commonly used delimiters are **comma** (**,**), **semicolon** (**;**), and **tab**, but the user can specify any character.

If the imported data contains delimiters within a single column, the user has the option to specify a *text qualifier* (❹). All characters between two text qualifiers are treated as a unit. Common text qualifiers include **single quotation marks** (**'**) and **double quotation marks** (**"**). If the imported data contains text qualifiers, they must be escaped (❺) by either doubling them ("" → ") or by preceding them with a backslash (\" → “).

8.3.1.4 Date Format

To import date fields correctly, the user can select a *date format* ❻ to use in the import file. Only one date format can be specified per file.

8.3.1.5 First Row Contains Column Names

If ❼ is checked, the first row is assumed to contain the column names.

8.3.1.6 Skip Empty Rows

Specify whether to skip empty rows in the data file (❽).

8.3.1.7 Preview and Status

Once a valid filename is specified, the preview (❾) displays the first few lines of the import file.

8.3.2 Data Mapping

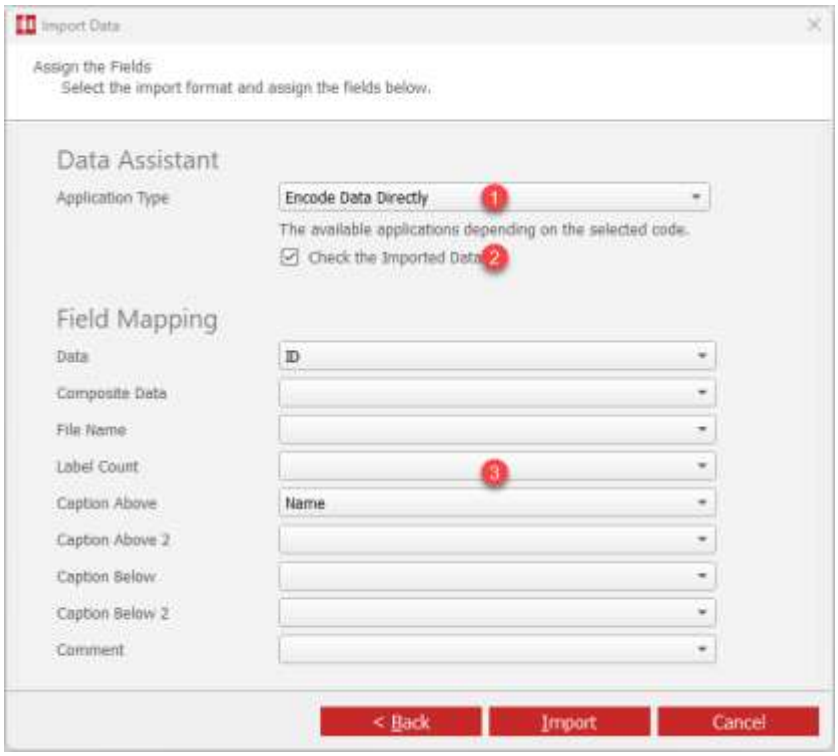


Figure 39: Data Mapping

The user can either import the data directly without further conversion or select one of the provided *application types* / *data formats* (❶). By default, data is imported directly in the raw format.

It is recommended to check the box *Check the Imported Data* (❷) in order to keep data consistency as high as possible. You should only uncheck it if you are experiencing performance problems.

Each column of the import file can be mapped to one of the columns *Data, Composite Data, Filename, Caption Above/Below (1/2)* in the data list (❸). The selection of a *Data* column is mandatory, all other columns are optional.

Below you find examples for importing various data formats.

Example vCard:

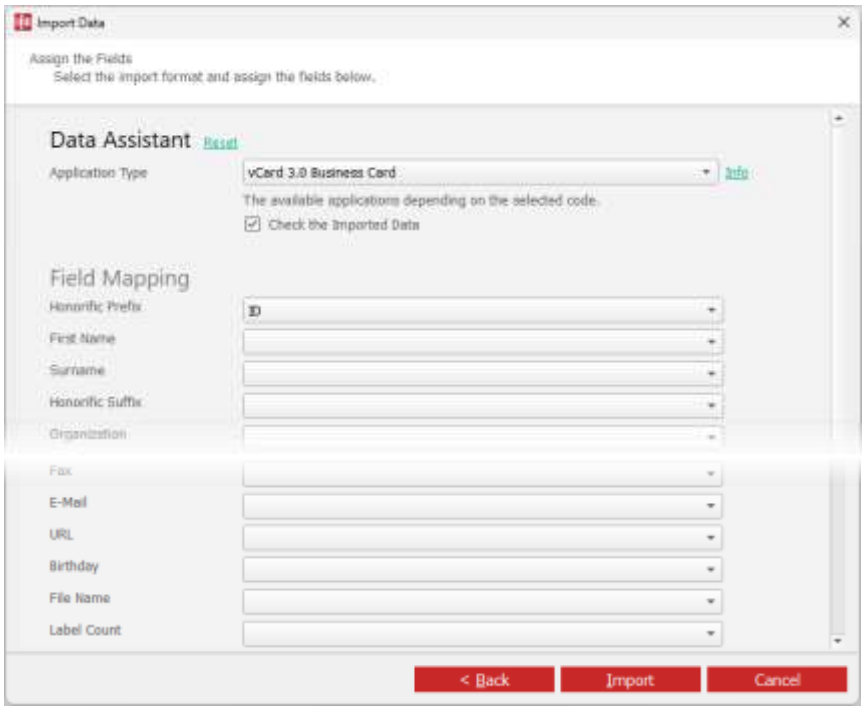


Figure 40: Import Assistant – vCard

Example Email:

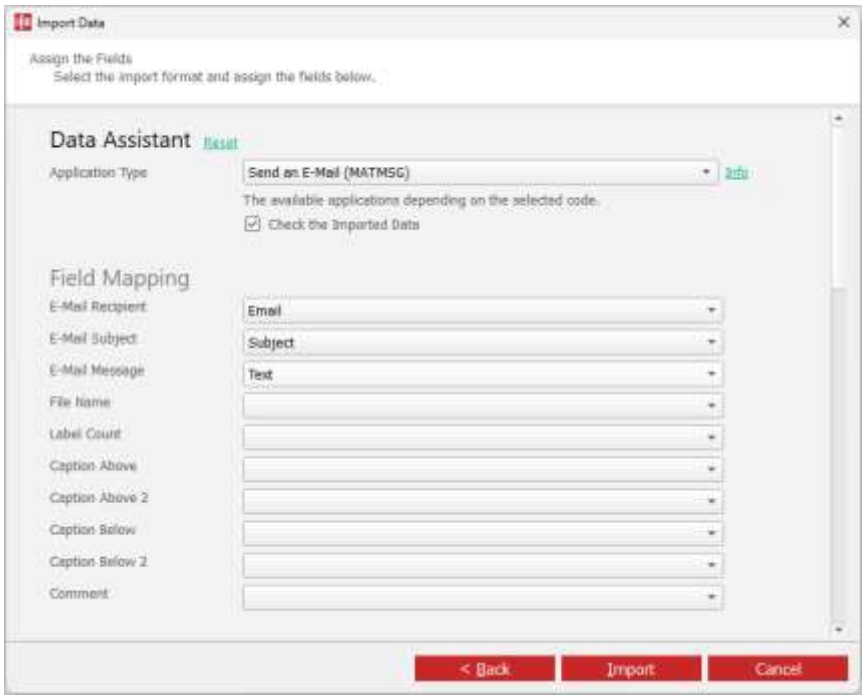


Figure 41: Import Assistant – Email

8.4 Serial Numbers

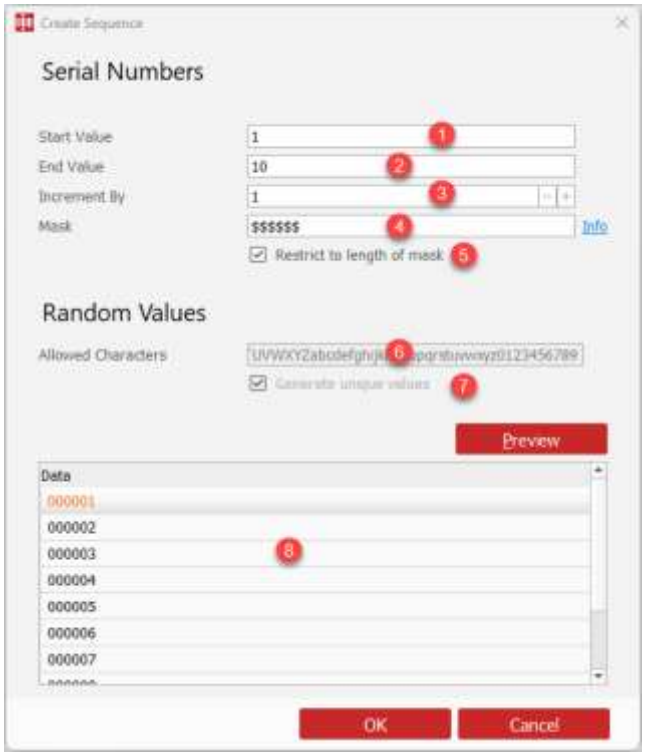


Figure 42: Serial Numbers Generator

The Serial Numbers Generator enables the user to populate the data column in the data list with formatted serial numbers.

8.4.1 Start Value / End Value / Increment

The range of a sequence is defined by the *Start Value* (❶), the *End Value* (❷), and the *Increment* (❸). The start value sets the lower limit of the range, the end value sets the upper limit, and the increment specifies the step size between consecutive elements.

► The last value of the sequence will never exceed the number specified in *End Value*.

8.4.2 Mask

The *Mask* ❹ defines the pattern to be applied to the sequence elements. The mask string can contain 4 types of placeholders that will be replaced by the digits of the serial or random number during generation. All other characters are left as they are.

Character	Description
#	Substitute the mask string with the generated sequence numbers, padding with leading spaces if the serial number has fewer digits than the mask string.
$	Like "#" but with leading zeros instead of spaces.
*****	Like "#" but with leading asterisks instead of spaces.
?	Each '?' is replaced by a random character (see *8.4.5 Random Value Generation*).
	All other characters are used literally.

Table 22: Format – Characters

Examples for serial numbers:

Format String	Sample Output	Description
$$	01, 02, 03, …, 10, 11	Leading zeroes
##	1, 2, 3, …, 10, 11	Leading spaces
**	*1, *2, *3, …, 10, 11	Leading asterisks
00$$	0001, 0002, 0003, …, 0099	Like “$$” but with “00” as **constant** prefix
A$$	A01, A02, A03, etc.	Like “$$” but with “A” as **constant** prefix

Table 23: Format – Examples

► If you enter “$$$” and create a sequence containing numeric values greater than 999, the code data will be truncated. In such instances, adjust the mask string accordingly.

8.4.3 Info Link

Click the *Info* link to display a brief description of the mask pattern.

8.4.4 Restrict Serial Numbers to Placeholders

This option (❺) allows you to control whether the generated serial number adheres strictly to the specified number of placeholders. When enabled, any extra digits at the beginning will be truncated. When disabled, the full serial number is displayed, regardless of the placeholder count.

Use this option to generate serial values without leading padding characters.

8.4.5 Random Value Generation

If the mask contains at least one ‘***?***’, random value generation is enabled. Each ‘?’ is replaced by a random value taken from the list of *Allowed Characters* (❻).

If you check *Generate unique values* (❼), the resulting sequence will be checked for multiple identical values. If so, an error message is displayed, when Ok is clicked. The user can either try again or change the sequence parameters to be more successful.

8.4.6 Preview

Click *Preview* to create the sequence based on the settings above. The list (❽) shows a brief preview of the generated data.

Click *OK* to append the sequence to the data list.

8.5 Export Codes

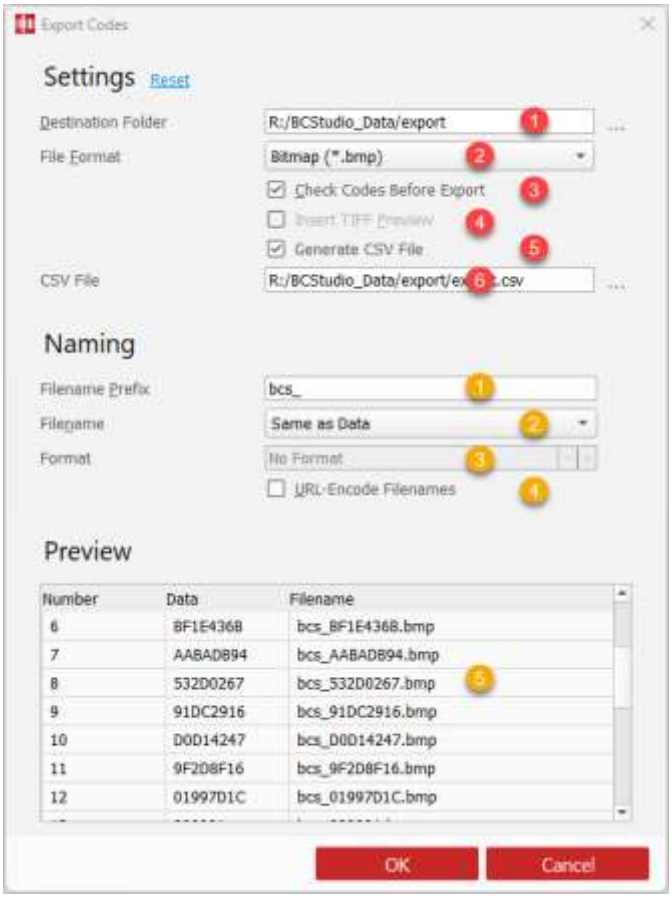


Figure 43: Export Dialog

Open the *Export Codes* dialog box by clicking the *Export* button.

The user can specify the export destination folder, the image format, filename generation parameters, and other export parameters.

8.5.1 Settings

8.5.1.1 Destination Folder

The code images are exported to the destination folder (❶).

8.5.1.2 File Format

Select the file format of the export file (❷). For a description of the different file formats, please refer to *Appendix C: Image Types*.

8.5.1.3 Check Codes before Export

If checked (❸), the code data will be checked before export. If the data list contains invalid data, the user is notified.

8.5.1.4 Insert TIFF Preview

If checked (❹), a TIFF preview is added to the generated EPS files. The check box is enabled only when the *EPS* file formats is selected.

8.5.1.5 Generate CSV File

If checked (❺), the data from the data list will be exported to the designated CSV file. You can specify the CSV file in the Export Data dialog box, accessible by clicking ... (see *8.6 Export Data*).

8.5.2 Naming

8.5.2.1 Filename Prefix

The generated filename is preceded by the *Filename Prefix* (❶).

8.5.2.2 Filename

The filenames are generated in one of the following ways (❷):

Generation Method	Description
Same as Data	The filenames are generated based on the code data. For instance, if the data is '12345', the corresponding filename will be 'bcs_12345.bmp' (where 'bcs_' is the prefix and 'bmp' is the file format). Note: If the sequence data includes characters not valid for filenames, it's advisable to use URL encoding.
Serial Number	The image filenames are generated using serial numbers starting from 1. Users can specify the desired number of digits for the generated numbers (❸).
Same as Data, but Prefer Existing	Same as '**Same as Data**' but already defines file names are not overwritten.
Serial Number, but Prefer Existing	Same as '**Serial Number**' but already defines file names are not overwritten.

Table 24: Sequence – Filename Generation

8.5.2.3 URL-Encode Filenames

If checked (❹), the filenames will be URL-encoded. This means that characters not permitted in filenames will be escaped by a percent sign followed by the hexadecimal ASCII code.

Character	HEX-Code	Character	HEX-Code	Character	HEX-Code
\	%5C	/	%2F	:	%3A
*	%2A	?	%3F	"	%22
<	%3C	>	%3E	|	%7C
%	%25				

8.5.3 Preview

The preview ❺ shows how the generated filenames will look like.

8.5.4 Export

Click *OK* to export the code images. If errors occur, up to 200 error messages will be displayed in an error list. Errors occur when:

- The item data cannot be encoded with the currently selected code type
  (e.g., ***2of5 IL*** symbology is used, but the data contains not only digits, but also letters).
- Invalid filenames were generated
  (e.g., a filename contains a backslash, but the "URL Encode the Filenames" option was not selected).

Keep in mind that exporting many data list items can take *a lot* of time and disk space.

8.6 Export Data

Open the *Export Codes* dialog box either directly by selecting *Export Data...* from the *Export* button's dropdown menu or indirectly via *Export* Dialog (see *8.5.1.5 Generate CSV File*).

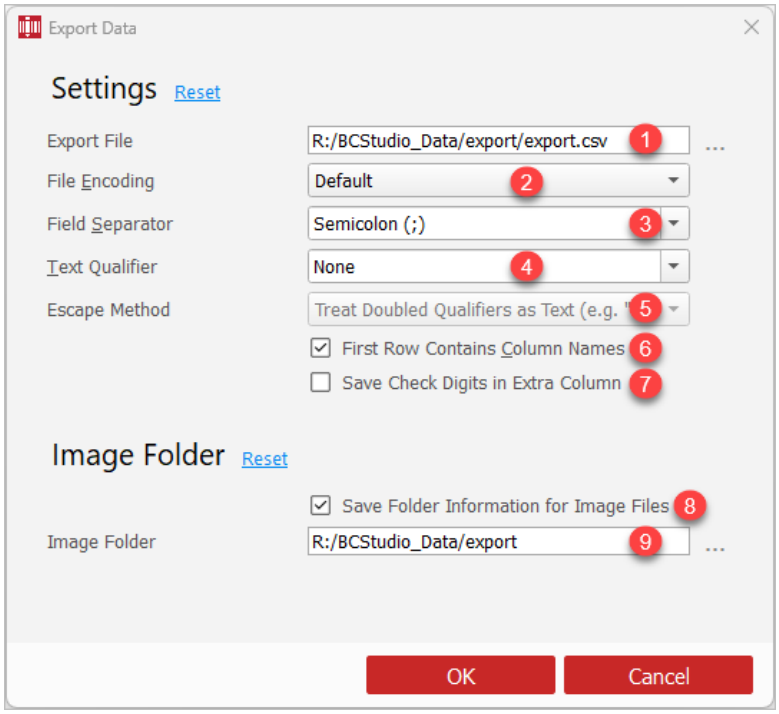


Figure 44: Export Data Dialog

8.6.1 Export File

The CSV file is exported to the filename in ❶.

8.6.2 File Encoding

Data export supports the following types of *File Encodings* (❷): *ANSI*, *UTF-8*, and *UTF-16*. The *Default* for Microsoft® Windows is *ANSI*, for Linux and macOS® it is *UTF-8*.

8.6.3 Field Separator and Text Qualifier

The *Field Separator* (❸) divides rows of data into columns. Commonly used delimiters are **comma** (**,**), **semicolon** (**;**), and **tab**, but the user can specify any character.

If the exported data contains delimiters within a single column, the user has the option to specify a *text qualifier* (❹). All characters between two specified text qualifiers are treated as a unit. Common text qualifiers include **single quotation marks** (**'**) and **double quotation marks** (**"**). If the exported data contains text qualifiers, they must be escaped (❺) by either doubling them ("" → ") or by preceding them with a backslash (\" → “).

8.6.4 First Row Contains Column Names

When ❻ is checked, the first row contains the column names.

8.6.5 Save Check Digits in Extra Column

When ❼ is checked, an extra column of calculated check digits is added.

8.6.6 Save Folder Information for Image Files

If checked (❽), the filenames of the code images are exported with their full path. You can find the export folder at ❾.

8.6.7 Image Folder

If the Export Data dialog was opened from the Export dialog, the destination folder (❾) for the code images is already defined there and cannot be changed here. Otherwise, if the Export Data dialog was opened directly from the dropdown menu, the path is not predefined and can be changed.

Click *OK* to save the CSV file.

9 Labels

9.1 Overview

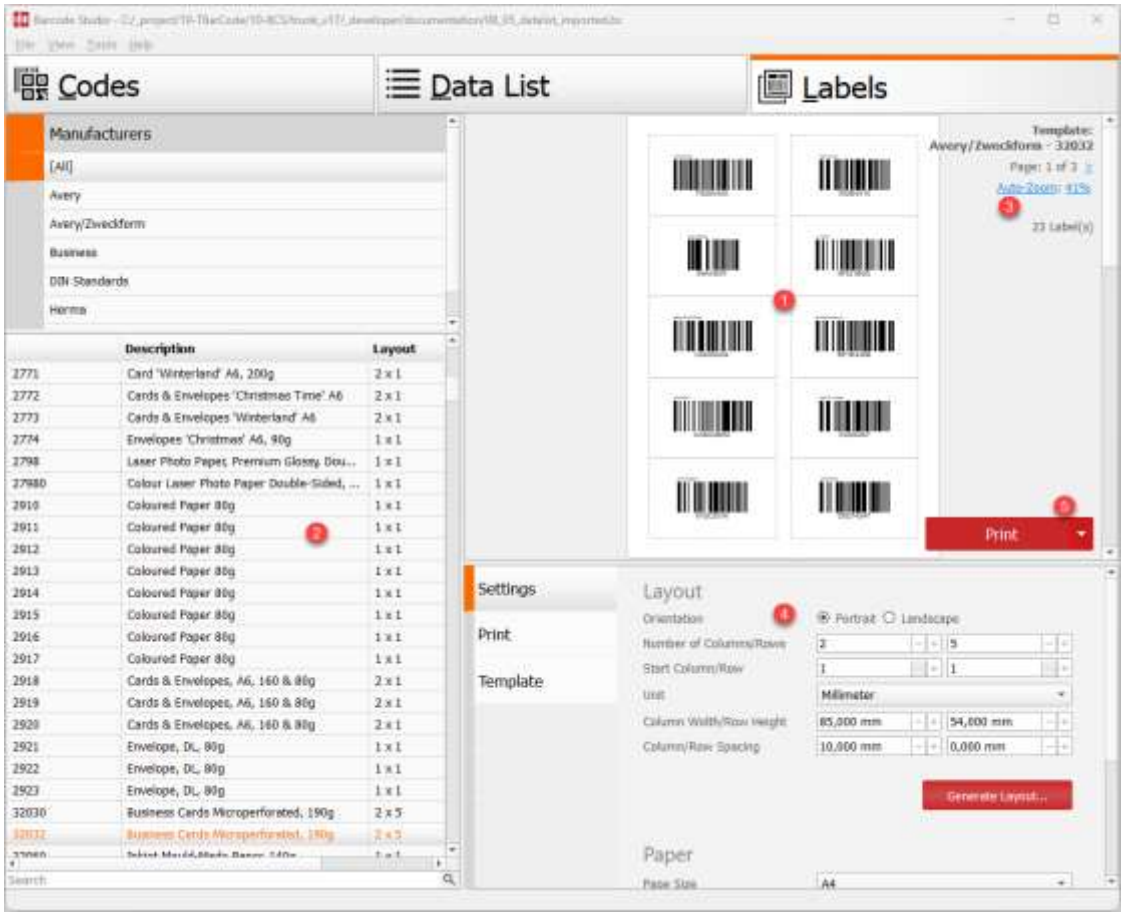


Figure 45: Labels View

The *Labels* window is divided into the following sections:

❶ *Layout Preview* – see chapter *9.2*
❷ *Manufacturers and Label Templates* – see chapter *9.3*
❸ *Labels Status* – see chapter *9.4*
❹ *Settings Pages* – see chapter *9.5*
❺ *Printing* – see chapter *9.9*

Barcode Studio provides an easy solution for printing codes on labels. Follow the steps below:

- Configure the code settings – see chapter *7 Barcode Studio User Interface*
- Generate or import the code data – see chapter *8 Data List*
- Choose a suitable label template (❷) or define the labels yourself (❸)
- Preview the layout (❶), view the labels status (❹) and print your labels (❺)

9.2 Layout Preview

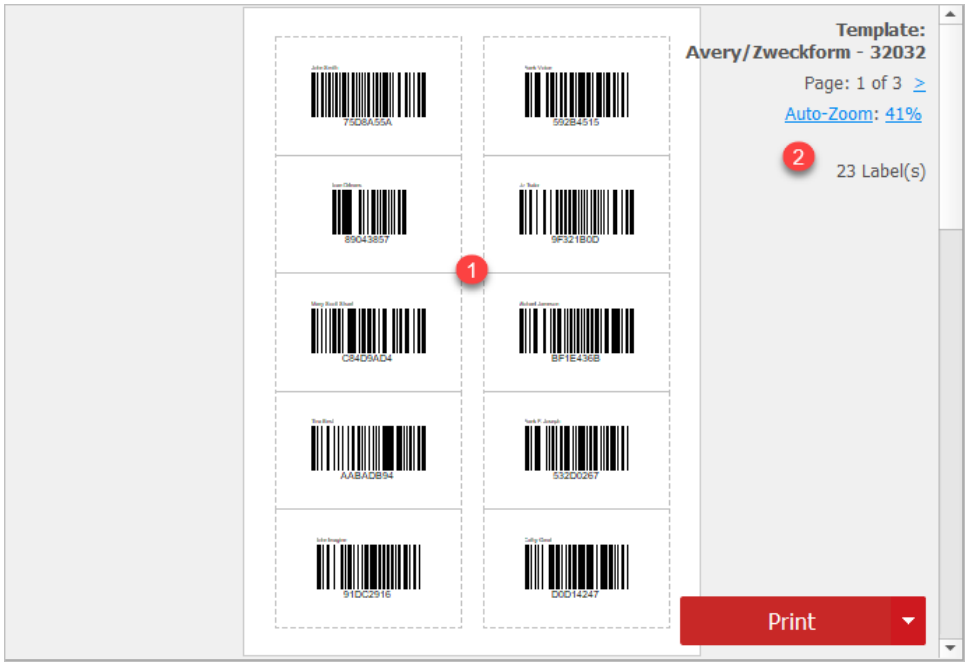


Figure 46: Labels Preview

The Layout Preview shows the layout of the selected single page (❶). You can navigate and zoom through the pages using the *Labels Status* view links (❷, chapter *9.4*), the scrollbar, and the keyboard.

Use the *Print* button ❸ to print the labels (see also chapters *9.9 Printing* and *6.8 Print Labels*).

If codes are invalid or too large, error messages notify the user about the issues.

Figure 47: Clipped Code in Label

Invalid Code in Label

9.3 Manufacturers and Label Templates

Choose a manufacturer (❶) and a label template (❷) from the list. Enter filter text (such as keywords) in ❸ to assist in finding the appropriate template.

The settings of the template are displayed and can be changed if necessary in the *Settings Pages* (chapter *9.5*).

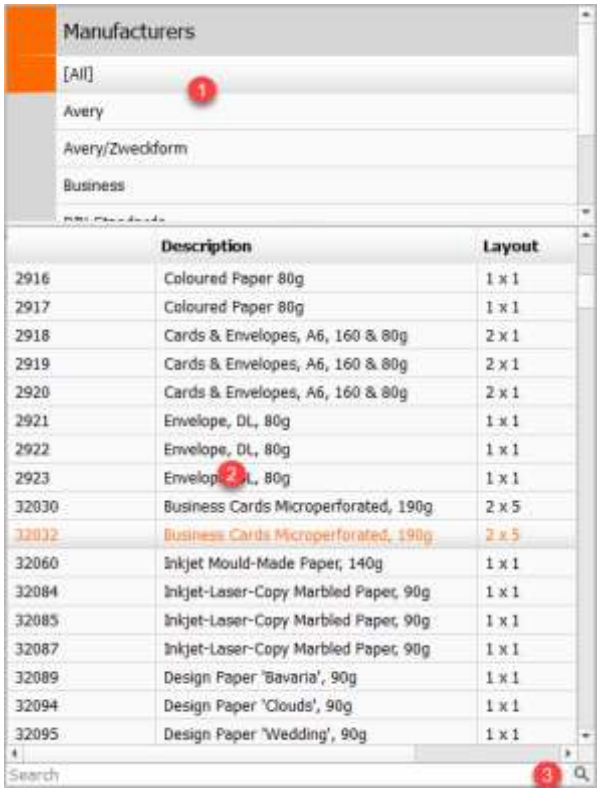


Figure 48: Manufacturers and Label Templates

9.4 Labels Status

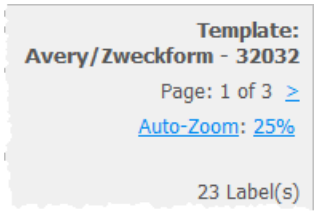


Figure 49: Labels Status View

9.4.1 Page Information

The *Label Status* view informs the user of the **total number** of pages and the number of the **current page**. The user can **navigate** through the pages by clicking "<" and ">" or by using scrollbar and cursor keys.

9.4.2 Zoom Information

The zoom information shows the current zoom factor in percent.

Click *Zoom* respectively *Auto-Zoom* to toggle between auto-zoom and a fixed zoom factor. Click the zoom value link to open the *Zoom* dialog box.

9.4.3 Labels Information

The label information informs the user of the number of labels to be printed and the number of issues that occurred.

9.5 Settings Pages

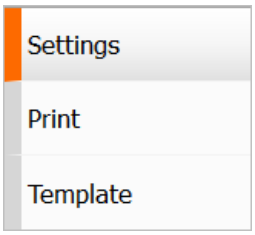


Figure 50: Tab Bar

Use the tab bar to switch between the following settings pages:

- *Page Settings* – see chapter *9.6*.
- *Page Print* – see chapter *9.7*.
- *Page Template* – see chapter *9.8*.

9.6 Page Settings

Figure 51: Page Settings

9.6.1 Layout

9.6.1.1 Orientation

Select the paper *Orientation* (❶). It can be either *Portrait* or *Landscape*.

9.6.1.2 Number of Columns/Rows

Specify the number of label columns and rows per page (❷).

9.6.1.3 Start Column/Row

Specify the position of the first label to be printed on the first page (❸). Labels positioned before the specified position will be skipped. The outcome of this setting varies depending on the print order.

9.6.1.4 Column Width / Row Height

Specify the *Column Width* and the *Row Height* (❺) of a label in the current unit of measure (❹). All labels on the sheet will be the same size.

9.6.1.5 Column / Row Spacing

Specify the *Horizontal* and *Vertical Spacing* between two labels (❻) in the current unit of measure (❹).

9.6.1.6 Generate Layout

Generate a simple custom format on the fly by defining the requested number of labels and optionally of columns (❼).

Figure 52: Generate a Label Layout

9.6.2 Paper

9.6.2.1 Page Size / Width / Height

Select either a predefined *Page Size* (❶, e.g., *A3*, *A4*, *Letter*, etc.) or choose a custom page size by specifying the *Width* (❷) and *Height* (❸) in the current unit of measure (❹).

9.6.3 Margins

Specify the Page *Margins* for the *Top* (❹), *Bottom* (❺), *Left* (❻), and *Right* (❼) sides using the current unit of measure (❹).

9.7 Page Print

Figure 53: Page Print

9.7.1 Print Sequence

The *Print Sequence* (❶) refers to the order in which the labels are printed (*Down, then Across* or *Across, then Down*).

9.8 Page Template

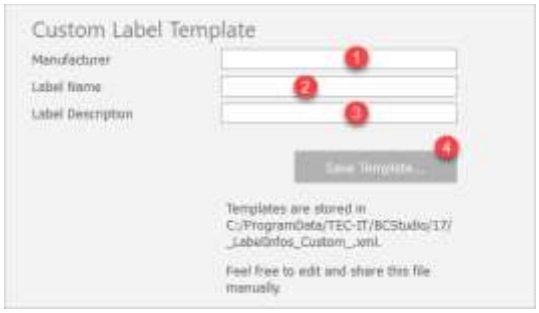


Figure 54: Page Template

Enter the template information (manufacturer, name, and description), then save the new template in the displayed file by clicking *Save Template* (❹).

9.8.1 Manufacturer

Enter the label *Manufacturer* (❶). If the manufacturer is not already in the list, it will be added; otherwise, the label will be updated.

► Keep in mind that you cannot overwrite predefined labels.

9.8.2 Label Name and Label Description

Enter *Label Name* (❷) and *Description* (❸). The label's name is typically its unique ID, and the description is a brief text that describes the label.

9.9 Printing

Click the *Print* button to initiate label printing, adjust printer settings as needed, and confirm by clicking *Print* in the dialog. If errors occur, up to 200 error messages will be displayed in an error list. Errors occur when:

- Items in the data list cannot be encoded with the currently selected code type (e.g., ***2of5 IL*** symbology is used, but the data contains not only digits, but also letters).
- Codes do not fit on the label, because they are too large.

Please note that printing many labels may require a significant amount of time.

10 Options

The Options settings are general settings for **Barcode Studio**. They are saved to the file bcstudio.bcopt after each change and are loaded at program startup. The file is located in the 'Application Data/Program Data' directory, e.g.:

- `C:\ProgramData\TEC-IT\BCStudio\17.0`
- `<user_dir>\AppData\Local\TEC-IT\BCStudio\17.0`

The options are managed in the Options dialog, accessible via the *Tools* ► *Options…* menu.

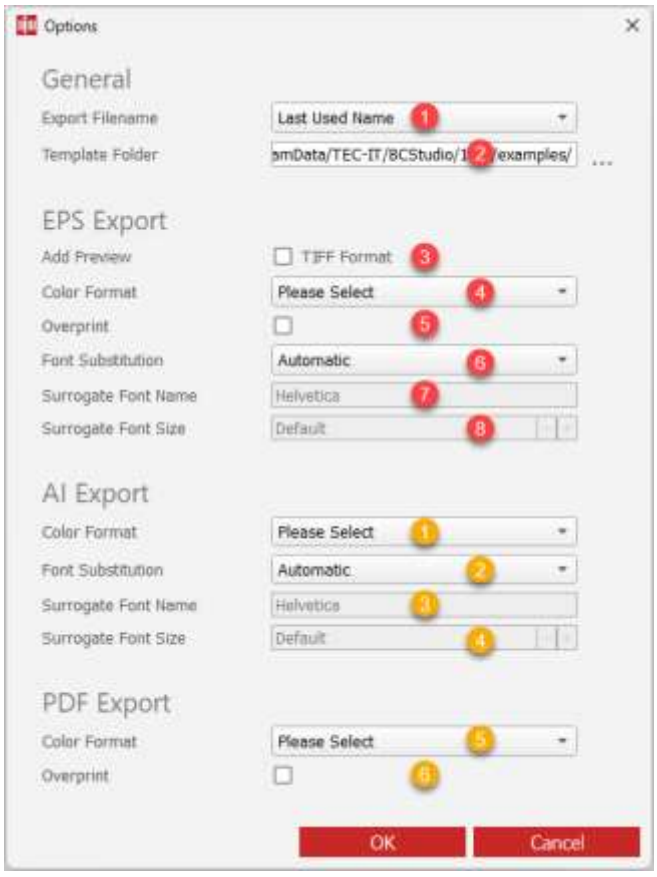


Figure 55: Options Dialog

10.1 General

10.1.1 Export Filename

The user may choose which filename (❶) is proposed for code export. Possible options are *Last used name* and *Like data*.

10.1.2 Template Folder

The *Template Folder* (❷) contains the code template files that appear in the *Template List* on the *Codes* page. The path to the template folder is initially set by the setup routine but can be customized.

10.2 EPS Export

10.2.1 Add Preview (TIFF Format)

Many applications do not support Postscript. In this case, the embedded EPS documents are displayed only as placeholders. To allow the user to see the content of the embedded document in preview or design mode, the EPS generator offers the option to add a TIFF preview image to the EPS file (❸).

► When *Add Preview* is enabled, the resulting EPS files are larger than the original file size. Therefore, it is recommended to disable *"Add Preview"* when printing a large number of documents.

10.2.2 Color Format

Select the default Color Format (❹) for EPS exports. Valid formats are *CMYK*, *Grayscale*, and *RGB*.

10.2.3 Overprint

Enable or disable the default overprint settings (❺) for EPS exports.

10.2.4 Font Substitution

Select the default *Font Substitution* (❻) mode for EPS exports. This is an advanced option and should only be changed if absolutely necessary.

Font Substitution	Description
None	The font is not replaced unless it is invalid.
Automatic (Default)	The font will be replaced with an appropriate alternative if necessary.
Fixed Font	The font specified in 'Surrogate Font Name' will replace the original font during export. The 'Surrogate Font Size' will also replace the original font size unless it is left at the default setting.

Table 25: Font Substitution

10.2.5 Surrogate Font

If *Font Substitution* (❻) is set to *Fixed Font*, the *Surrogate Font Name* (❼) overrides the original *Font* property. If not set to *Default*, the *Surrogate Font Size* (❽) overrides the original font size.

► Note that you must specify a *PostScript* font name that does not contain spaces.

10.3 AI Export

10.3.1 Color Format

Select the default Color Format (❶) for AI exports. Valid formats are *CMYK*, *Grayscale*, and *RGB*.

10.3.2 Font Substitution

Select the default *Font Substitution* (❷) mode for AI exports. This is an advanced option and should only be changed if absolutely necessary.

Font Substitution	Description
None	The font is not replaced unless it is invalid.
Automatic (Default)	The font will be replaced with an appropriate alternative if necessary.

Fixed Font	The font specified in 'Surrogate Font Name' will replace the original font during export. The 'Surrogate Font Size' will also replace the original font size unless it is left at the default setting.

Table 26: Font Substitution

10.3.3 Surrogate Font

If *Font Substitution* (❷) is set to *Fixed Font*, the *Surrogate Font Name* (❸) overrides the original font property. If not set to *Default*, the *Surrogate Font Size* (❹) overrides the original font size.

► Note that you must specify a *AI* format font name that does not contain spaces.

10.4 PDF Export

10.4.1 Color Format

Select the default Color Format (❺) for PDF exports. Valid formats are *CMYK* and *RGB*.

10.4.2 Overprint

Enable or disable the default overprint settings (❻) for PDF exports.

11 FAQ

- For more information on codes and their parameters, please refer to the **TEC-IT Barcode Reference** at www.tec-it.com/download/PDF/Barcode_Reference_EN.pdf
- More FAQs for **Barcode Studio** can be found on our website www.tec-it.com/support/faq/Default.aspx
- For more information on barcode issues, see www.tec-it.com/support/faq/barcode/bar-code-config/Default.aspx
- In case of problems, please contact our support team: support@tec-it.com.

12 Licensing

12.1 Product Variants

Barcode Studio is currently available in following product variants:

- The *Barcode Studio Lite* Edition
  is a restricted version of **Barcode Studio**, limited to a few, but important barcode types and does not allow bulk export of code images.
- The *Barcode Studio Pro* Edition
  is the full, unlimited, and full featured version of **Barcode Studio**.

Both editions are available as subscription. Please visit www.tec-it.com/order/Default.aspx to view and compare the available license types and prices.

12.2 Entering your License Data

To enter the license information, select *Help* ► *License…* from the menu. The dialog shown below appears. **Barcode Studio** offers three ways to enter the license data:

- *Online Activation using an Activation Key* (see *12.2.1*).
  This is the default method. Internet access is required.
- *Renew your Activation* (see *12.2.2*).
  If your license key is expired, you can renew your activation here. Internet access is required.
- *Manual Licensing* (see *12.2.3*).
  Please contact us if your system does not have Internet access or if you prefer to use manual activation for other reasons. We will be happy to send you the appropriate license data for manual licensing.

12.2.1 Online Activation using an Activation Key

If you received an *Activation Key* from TEC-IT, please proceed with *Online* product activation ❶. Internet access is required.

Figure 56: License Dialog – Online Activation

1. Enter the activation key In field ❷.
2. Enter your email address in field ❸. A license certificate will be automatically sent to this address.

3. Fill in the rest of the information (company, country, postal code and city). All fields are required for a successful activation.
4. Confirm the information entered by agreeing to the Privacy Policy and click *Send*.

You will receive a confirmation message upon a successful activation. If you encounter any issues or errors, please contact TEC-IT for assistance.

12.2.2 Renew your Activation

If you have successfully activated your license before, but your license has expired, you can renew your activation. The fields in the dialogue box will already be filled in with your activation details.

► Important: Please buy an update **before** renewing your activation!

Figure 57: License Dialog - Renew Activation

1. Please update your data to the latest status.
2. If your activation key changed, press *X* (❷) to clear the field and enter the new activation key.
3. Confirm the information entered by agreeing to the Privacy Policy and click *Send*.

You will receive a confirmation message upon a successful activation. If you encounter any issues or errors, please contact TEC-IT for assistance.

12.2.3 Manual Licensing

Manual licensing is available as an alternative for systems that are not connected to the Internet. Select *Manual Licensing* ❹ and enter the license information provided by TEC-IT.

► Important: Please enter the license data **exactly** as it was provided by TEC-IT!
Respect spacing and capitalization. To avoid typos, please copy and paste the information from the email containing your license information whenever possible.

► Single licenses
If you have purchased a single (workstation) license, you will need to provide the '*System ID*' (or *hostname* on Linux and macOS®) of the target computer. You can locate the *System ID* (or *hostname*) in the licensing dialog (refer to the screenshot below).

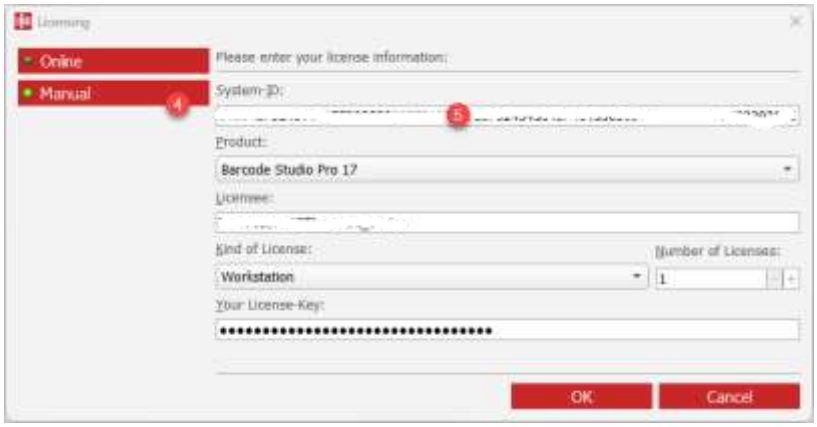


Figure 58: License Dialog – Manual Licensing

The following steps are required to license the product:

1. On the top you find the *System ID*[1] (❺) of the computer.
   For *"Single"* licenses, we will request the System ID of the target computer from you.
2. In the field *"Product"* please choose either "*Barcode Studio Lite*" or "*Barcode Studio Pro*".
3. In the *"Licensee" field,* please enter the name of the license holder.
4. In the *"Kind of License"* field*,* select the kind of license that you purchased. Possible license types are:
   - *Workstation*
   - *Enterprise*
5. In the *"Number of Licenses"* field*,* enter the number of licenses that you purchased.
6. In the *"Your License-Key"* field*, e*nter the license key exactly as received from TEC-IT (no space characters, …).
7. Confirm with *OK*.

A message will inform you about the successful activation/licensing. If you encounter any issues or errors, please contact TEC-IT.

12.3 Temporary Trial Licenses

You have the option to obtain a temporary trial license that allows you to use **Barcode Studio** without restrictions for a limited period of time.

When you open **Barcode Studio**, you will see a splash dialog with the option to request a trial license online.

[1] On UNIX, Linux, or macOS®, the hostname of the system is used as System ID (relevant only for Single licenses)

Figure 59: Request Trial License

If you click *YES*, the license information will be submitted automatically. In case of any problems, please contact TEC-IT.

Figure 60: Demo Version Unlocked

If you have successfully obtained a trial license, you will be notified of the remaining trial period each time you start the application.

12.4 Subscriptions

As soon as the subscription nears its end or has expired, you are notified about the expiration date and are given the option to extend your subscription.

When you open **Barcode Studio**, a splash dialog will appear, offering the option to request an extension of your subscription online. As soon this is done, you can renew your activation (see *12.2.2 Renew your Activation*).

Figure 61: Subscription Expired

13 Contact and Support Information

TEC-IT Datenverarbeitung GmbH

Address:	Hans-Wagner-Strasse 6 AT-4400 Steyr Austria/Europe
Phone:	+43 / (0)7252 / 72 72 0
Fax:	+43 / (0)7252 / 72 72 0 – 77
Email:	office@tec-it.com
Web:	www.tec-it.com

Appendix A: Barcodes

A.1 Supported Barcodes

Please refer to the additional “Barcode Reference” available from www.tec-it.com/download/PDF/Barcode_Reference_EN.pdf.

This document provides an overview of supported barcodes, enumeration values (for developers using **TBarCode OCX**), standard print ratios, and standard check digit methods.

A.1.1 MaxiCode

13.1.1.1 Setting SCM parameters

Please refer to the additional “Barcode Reference” available from www.tec-it.com/download/PDF/Barcode_Reference_EN.pdf.

A.2 Check Digits

The available check digit calculation methods depend on the code type. Each type provides a default method.

See section *7.8.1.4 Check Digit* to set the check digit in **Barcode Studio.**

Please refer to the additional “Barcode Reference” available from www.tec-it.com/download/PDF/Barcode_Reference_EN.pdf.

A.3 Print Ratio

The *Print Ratio* is the relationship between the widths of the bars and the widths of the spaces in a code. Each symbology specifies a fixed print ratio.

See section *7.12.2.3 Bar:Space Print Ratio* to set the print ratio in **Barcode Studio.**

Please refer to the additional “Barcode Reference” available from www.tec-it.com/download/PDF/Barcode_Reference_EN.pdf.

A.4 Format

The *Format* acts like a mask used for formatting code data prior to encoding. Placeholders within the specified format string can be combined with constant data characters to create the final code data string. Additionally, control characters are supported.

See section *7.12.2.1 Format / Subset* to set the format string in **Barcode Studio.**

Please refer to the additional “Barcode Reference” available from www.tec-it.com/download/PDF/Barcode_Reference_EN.pdf.

A.5 Escape Sequences

To encode unprintable or special characters in a code, use *Escape Sequences*. These sequences always begin with a backslash ("\") followed by the specific escape sequence. Additionally, escape sequences can be used to encode binary data (bytes) if the symbology used supports this feature, such as PDF417 or Data Matrix.

See section *7.8.1.2 Escape Sequences* to enable or disable the translation of escape sequences in **Barcode Studio.**

Please refer to the additional “Barcode Reference” available from www.tec-it.com/download/PDF/Barcode_Reference_EN.pdf.

Appendix B: Error Messages

Error	Recommended Action
Wrong character 'x' at position y!	You have entered a character that cannot be encoded with the selected symbology. Please use another symbology or change your input data.
Wrong character 'x' at position y in the Format string!	You entered a character in the format string that is not a valid Format command and cannot be encoded with the selected symbology. Please change or remove the invalid character from the Format string or use a different symbology.
Wrong check digit 'x' at position y!	You have entered a check digit that is incorrect for the selected symbology. Please correct or remove the check digit so that the application can calculate it automatically.
Wrong number of input characters (needs x chars)!	You have entered too many or too few characters for the selected symbology. Please use a different symbology or correct the length of the input data.
Input string too long	The input string is too long**.** Please reduce the length of the data string.
Barcode does not fit into bounding rectangle	The output resolution is too low to produce readable codes with the given symbol size. You must either increase the width/height of the code, the module width, or the output resolution.
No input characters	No code data is specified. Please enter data in the *Data* field.
Invalid code page or invalid characters!	You have entered characters that cannot be encoded in the selected code page. Please use another code page or change the input data.
Not implemented (reserved for future use)	The selected symbology is not yet supported. Please contact TEC-IT for an implementation request.
Invalid data format. Please examine the specification	The input data does not match the definition of the given code type. Please check the code type specification for the required data format. This error message is mainly displayed when a detailed error description would exceed the available space.
Quiet zone too small for vertical bearer bars	You have selected *Rectangle* bearer bars, which require a sufficient horizontal quiet zone. Please increase the left and right quiet zones to at least 12 modules.
Error in linear Component: …	You intend to create a composite symbol but encounter issues due to incorrect linear symbol data. Refer to the following error message for detailed information about the problem.
Error in 2D Composite Component: …	You intend to create a composite symbol but encounter issues with the data for the 2D component. Refer to the following error message for a detailed description of the problem.
Invalid number of input characters in combination with composite!	When using GS1-128 as part of a composite symbology, a minimum number of symbol characters must be entered (which may differ from the number of input characters). Please increase the number of input characters in the Data field.

Table 27: Error Descriptions

Appendix C: Image Types

The following image formats can be used. Please note that if the resolution of the generated bitmap is too low, the code may become unreadable (This does not apply to vector-based image formats).

Image Format	Description	Internal Format	Color	Printing Resolution included?
BMP	Microsoft® Windows Bitmap.	Bitmap	color (RGB)	Yes
AI	Adobe Illustrator v7 Offers the following sub-formats: ▪ CMYK uses CMYK color space (4 color channels). ▪ RGB uses RGB color space (3 color channels). ▪ Grayscale uses one channel (luminance) only.	Vector	color (CMYK, RGB) or grayscale	
EPS	Encapsulated PostScript. Offers the following sub-formats: ▪ CMYK ▪ RGB ▪ Grayscale See AI format. Please note that the EPS export may require additional font adjustments (see section *10.2.4 Font Substitution*). Color overprinting is supported.	Vector	color (CMYK, RGB) or grayscale	No
GIF	Graphics Interchange Format.	Bitmap	color (RGB)	No
JPG	Joint Photographic Expert Group image file format.	Bitmap	color (RGB)	Yes
PDF	Portable Data Format. Offers the following sub-formats: ▪ CMYK ▪ RGB ▪ Spot Colors See AI format. Color overprinting is supported.	Vector	color (CMYK, RGB)	No
PNG	Portable Network Graphics.	Bitmap	color (RGB)	Yes
SVG	Scalable Vector Graphics	Vector	color (RGB)	No
TIF	Tagged Image File. No compression is supported.	Bitmap	color (RGB)	Yes

Table 28: Image Formats

Appendix D: Command Line Parameters

Barcode Studio can also be started from the command line. It supports various parameters that allow the user to automate the code generation process.

D.1 Syntax

```
1) BCStudio -d=<data> -out=<outputfile>
            [[-s=]<settingsfile>] [barcode options] [image options] [remaining options]
2) BCStudio -datafile=<datafile> -out=<outputfile>
            [[-s=]<settingsfile>] [barcode options] [image options] [remaining options]
3) BCStudio -datalist=<datalistfile> -outlist=<outputlistfile>
            [-skipheader=<numlines>] [-parseempty]
            [[-s=]<settingsfile>] [barcode options] [image options] [remaining options]
4) BCStudio -datalist=<datalistfile> -out=<outputfile>
            [-skipheader=<numlines>] [-parseempty]
            [[-s=]<settingsfile>] [barcode options] [image options] [remaining options]
5) BCStudio <settingsfile>
6) BCStudio -g [-d=<data> | -datafile=<datafile>]
            [[-s=]<settingsfile>] [barcode options]
7) BCStudio [-? | -help]
```

The Code Generation Arguments

-d=<data>	Pass the code data directly via command line. Is used alternatively to *-datafile=<datafile>* and *-datalist=<datalistfile>.*
-datafile=<datafile>	Pass the code data via data file. Is used alternatively to *-d=<data>* and *-datalist=<datalistfile>.*
-datalist=<datalistfile>	Pass a list of code data via a data list file separated by line breaks. Is used alternatively to *-d=<data>* and *-datafile=<datafile>*. You can use *-datalist* together with: ▪ *-outlist*: each code is created in a separate output file. The number of entries in the data list and the output list should be equal. ▪ *-out*: all codes are created in a single output file (PDF only). **Exception:** If the output list file contains only one filename, all codes will be exported to that file. (equivalent to parameter *-out*, PDF only).
-out=<outputfile>	The name of the output file (image file). The filename extension may be one of the following: JPG, GIF, TIF, BMP, PNG, EPS, AI, SVG, and PDF. The filename extension must be specified! Can be used alternatively to *-outlist=<outputlistfile>*.
-outlist=<outputlistfile>	The output list filename. It contains the list of output filenames, separated by line breaks. Can be used alternatively to *-out=<outputfile>*. If this argument is used, you must also specify *-datalist=<datalistfile>*. The number of entries in both lists should be equal.
-skipheader=<numlines>	Specifies the number of lines on top of the date list file that should be handled as header and therefore be skipped. This argument can only be used in conjunction with *-datalist=<datalistfile>.*
-parseempty	If set, empty lines are handled as data lines, otherwise they are skipped. This argument can only be used in conjunction with *-datalist=<datalistfile>.*
[-s=]<settingsfile>	Settings-Files are *.bc files which have been created by **Barcode Studio**. You can override values of the settings file by passing command line arguments. If you pass only the settings file without an argument name and no further argument, the application is opened in graphics mode.
-g	Start the application in graphics mode. This setting is needed to pass arguments to **Barcode Studio** and start it in graphics mode, though.
-? -help	Display the help screen.

The Barcode Options

`[-w=<width> |-m=<modulewidth>] [-h=<height>] [-u=<unit>] [-o] [-rot=<rotation>]`	
-w=<width> -m=<modulewidth>	Specify either the total width of the symbol or the module width (dimension X or narrow bar width); given in units (default: µm). E.g.: -w=50250 … code is 50.25 mm wide
-h=<height>	Height of symbol, given in units (default: µm)
-u=<unit>	The measure unit used for the code sizes (width, height, and module width). Possible values: ▪ MICM (Micrometer) ▪ MM (Millimeter) ▪ MIL (Mils) ▪ IN (Inch) ▪ PX (Pixel) ▪ PT (Point) See also usage dialog.
-o	Optimal resolution: adapt module width to an integral pixel width. Recommended for low resolution devices.
-rot=<rotation>	The code rotation in degrees. Valid values are 0, 90, 180, and 270. (Default = 0).

The Image Options

`[-it=<imagetype>] [-cs=<colorspace>] [-op] [-r=<dpi>]` `[-spotfg=<spot_color>] [-spotbg=<spot_color>] [-spottxt=<spot_color>]`	
-it=<imagetype>	The image type. The image types are equal to the file extensions.
-cs=<colorspace>	The color space to be used to generate the output file. If not set, the default color space of the image type is used (EPS, AI, and PDF only). See also usage dialog.
-op	Activates color overprinting (EPS and PDF only).
-r=<dpi>	The resolution of the output file in dpi (default = 600 dpi).
[-spotfg=<spot_color>]	Override the foreground color with a spot color (PDF only). See also usage dialog.
-spotbg=<spot_color>	Override the background color with a spot color (PDF only). See also usage dialog.
-spottxt=<spot_color>	Override the text color with a spot color (PDF only). See also usage dialog.

The Remaining Options

`[-logformat=<logformat>] [-log=<logfile>] [-q] [-fs=<fileseparator>]`	
-logformat=<logformat>	The log file format. Allowed formats: TEXT, JSON. (Default = TEXT).
-log=<logfile>	The name of the log file. If this argument is not set, no log file is generated.
-q	Activate the “Quiet Mode”. The program does not interact with the user. If an error occurs, a message is logged to file *error.log*.
-fs=<fileseparator>	A custom file separator, that can be used for separating filenames of command line arguments and for the list of filenames in the output filenames list.

D.2 Examples

Generate Code128 (default), content=98765, size 50x30mm:

```
bcstudio -d="98765" -out=barcode.bmp -w=50000 -h=30000 -o
```

Generate a code image (JPG) with 300 dpi and 0.330 mm module width:

```
bcstudio -d="12345" -out=barcode.jpg -h=30000 -m=330 -r=300
```

Generate a code (symbology defined in bc file) 70x30mm:

```
bcstudio -d="123" -out="c:\pictures\bc.jpg" -s=bcsettings.bc -w=70000 -h=30000 -o
```

Generate a code with data from txt file:

```
bcstudio -datafile=data.txt -out=barcode.tif -s=bcsettings.bc -w=70000 -h=30000 -o
```

Generate a list of codes using the arguments datalist and outlist:

```
bcstudio -datalist=datalist.txt -outlist=outputfilenames.txt -s=bcsettings.bc -o
```

Generate a code with optimized readability and fixed module width:

```
bcstudio -d="ABCDE" -out=barcode.bmp -s=bcsettings.bc -h=30000 -m=265 -o
```

Open a settings file in graphics mode:

```
bcstudio bcsettings.bc
  or
bcstudio -g -s=bcsettings.bc
```

Opens a settings file in graphics mode with given width and height: 50x30mm:

```
bcstudio -g -s=bcsettings.bc -w=50000 -h=30000 -u=MICM
```