User Guide: ExportPDF for CorelDraw

1. Introduction

Welcome to the guide for the ExportPDF script for CorelDraw. Its main goal is to automate the generation of personalized PDFs for a **wide range of products**, such as hoodies, caps, sports jerseys, mugs, and more. The script allows you to flexibly define **up to two different product types** and fully customize their names and properties.

Main features of the script:

- Dynamic loading of sizes from CSV and the ability to permanently save custom atypical sizes.
- Automatic and manual CSV column mapping with a clear interface.
- Personalization of text fields, including intelligent preservation of formatting for multi-line text.
- Advanced graphic numbering: Inserting digits 0–9 as individual graphic symbols.
- Option to insert a logo directly into numbers with dynamic placement.
- Advanced manual diacritics support with per-character placeholders: Visual setup of precise position and size of diacritical symbols for each letter individually. Support for 10 diacritics types across 4 languages (CZ, SK, DE, PL).
- Automatic checking and warning about missing characters (glyphs) in the font used, with the option to skip the check.
- Flexible configuration of exported file and folder names using templates with live preview.
- Flexible processing of up to two fully customizable product types (e.g., hoodies and caps).
- Display of optional graphics based on CSV data (e.g., for sponsor logos).
- Multilingual interface (Czech, English, German, Slovak, Polish).
- Profile management: Ability to save, load, import, and export complete sets of settings for different types of orders.
- Demo mode with a limited number of launches and processed rows.
- Two working modes: page mode (each product = separate page) and group mode (products as named groups on a single page).
- Generation of laser contours around selected objects for cutting/engraving purposes.
- Focus Mode detection in CorelDraw with automatic user warning.

This guide will walk you through installation, configuration, and usage of the script.

1.1. What's New in the Current Version

The following features and improvements have been added compared to previous versions:

New features:

- **Laser contours (Chapter 8.8):** A completely new feature for automatic generation of laser/cutting contours around objects named "Laser". The script supports two modes — page and group — and automatically detects which one to use. The contour offset is configurable in millimeters.
- **Group mode (Layout Mode, Chapter 9.1.1):** The script now supports, in addition to page mode, a group mode where products are organized as named groups on a single page. The mode is configured in the settings file.
- **Focus Mode detection (FAQ, Chapter 10):** The script automatically recognizes whether CorelDraw is in group editing mode (Focus Mode) and warns the user that this mode must be exited before running the script.
- **Skip glyph check:** A new `SkipGlyphCheck` option allows skipping the automatic font character validation if you know your font is fine or if the check causes issues.

Improvements:

- **Enhanced multi-line text style cache:** The script now intelligently remembers formatting styles (font, size, bold, italic, underline) for both lines of a multi-line text field, ensuring more accurate formatting preservation during personalization.
- **Enhanced glyph validation:** The automatic missing character check now uses an advanced fingerprinting comparison method and includes special rules for problematic fonts (e.g., "FaceOff").
- **Duplicate page detection:** The script checks whether the document contains duplicate page names and warns the user if found.

2. Requirements

To successfully use the script, you need:

Software:

- The script is tested and fully functional on CorelDraw X6, CorelDraw 2024, and CorelDraw 2025. Compatibility with other versions is possible but may require personal customization of the script.

Input files:

- **CorelDraw template (.cdr):** A prepared CorelDraw document containing:
  - Correctly named text fields for personalization.
  - Optionally: Symbols for diacritics, graphic numbers, logo — placed in the "Symboly" layer.
  - Optionally: Reference elements for dynamic logo placement.
  - Pages for individual sizes and product types (in page mode), or named groups on a single page (in group mode).
- **CSV file (.csv):** A comma-separated values file containing data for personalization (e.g., number, name, size, code). The first row must contain column names. UTF-8 encoding is required.

3. Script Installation and Launching

The plugin for CorelDRAW must be compiled specifically for your version of the program — each version uses different internal libraries. Installation and updates are therefore performed by the author directly on your computer, either in person or remotely via TeamViewer.

Launching the script in CorelDraw:

1. Open CorelDraw and the document (template) you want to work with.
2. Go to menu Tools > Scripts > Run Script...
3. In the dialog box, find and select the script file.
4. Click "Open". The script will start.

Tip: Adding a custom keyboard shortcut for quick script launch:

1. Go to menu Tools > Options > Customization > Commands > Macros.
2. Select ExportPDF_Corel.
3. Click on the "Keyboard Shortcuts" panel and enter your custom keyboard shortcut for launching the script (e.g., Ctrl+10) in the "New Keyboard Shortcut" field.

3.1. First Launch and Security Check

During the **first launch of the ExportPDF-Corel script** after installation (or after updating to a new version), you may notice that **startup takes longer than usual**. During this time, an information window titled "Script Initialization" may appear and CorelDraw itself may temporarily appear as "not responding" (the title bar may display "Not Responding").

Reason for longer startup: The script performs a **one-time security integrity check of its files**. This process is crucial for ensuring that the script has not been tampered with or corrupted, and for verifying your license. It includes, among other things:

- **Comprehensive integrity check:** The script calculates and verifies checksums (hashes) of its internal data to ensure no unauthorized tampering has occurred.
- **License mechanism initialization:** Sets up basic data for the licensing system (e.g., for the demo version or full activation).

What to expect:

- **"Script Initialization" window:** A dialog box will appear with information about the ongoing action.
- **Possible temporary "not responding" application:** Depending on your computer speed and CorelDraw version, the application may appear "not responding" for several tens of seconds to a few minutes. This is normal behavior because the script is intensively working with data storage and performing complex calculations during this operation.
- **Patience is key:** Please **do not attempt to close CorelDraw** or forcefully terminate the script. Wait until the process completes and the dialog window disappears.

Subsequent launches: All subsequent script launches will be **significantly faster** because the one-time deep check will no longer be needed. Only a quick check of current data and timestamps will be performed, which takes only a fraction of a second.

4. First Launch and Basic Setup

4.1. Language Selection

On the first launch of the script, you will be prompted to select the user interface language. The script supports five language versions:

- **Czech (CZ)** — default language.
- **English (EN)**
- **German (DE)**
- **Slovak (SK)**
- **Polish (PL)**

Your selection is automatically saved to the settings file and will be used for each subsequent launch. You can change the language at any time later using the "Change language..." button in the main script dialog (in the "Language & Info" panel).

After selecting the language, the script will start and display the main dialog in the selected language.

4.2. Licensing System

The script contains a three-tier licensing system:

Demo version:

- On the first run (or if no valid license is found), the script runs in demo mode.
- **Limitations:** Max 30 launches and max 20 CSV rows.
- Demo status is displayed in the "About the Script" dialog.
- The demo can be extended with a special key ("Extend demo version..." button in the "About the Script" dialog).

Subscription (monthly/yearly):

- Activation with a time-limited license key.
- Monthly (30 days) or yearly (365 days) license.
- Expiration date is displayed in the "About the Script" dialog and in the log.

- 7 days before expiration, the script shows a warning with the number of remaining days.
- After expiration, the script shows a dialog prompting you to enter a new key and Machine ID for ordering.
- **Renewal:** Enter a new key via the "Extend license..." button in the "About the Script" dialog. Remaining days are automatically added to the new period.

Permanent license:

- One-time activation without time limitation.
- Displayed as "Permanent license" in the "About the Script" dialog.

"About the Script" dialog:

- Displays script version, license status, subscription type, and expiration date.
- **Machine ID** — a unique identifier for your PC (copyable). Required for generating a license key — send it to the seller when ordering.
- Buttons: "Activate full version..." / "Extend license..." / "Extend demo version..."

Activation and renewal procedure:

1. Open the "About the Script" dialog ("About the Script..." button in the main dialog).
2. Copy the Machine ID and send it to the seller along with your order.
3. You will receive a license key.
4. Click "Activate full version..." (or "Extend license...").
5. Paste the key and confirm.
6. The script activates immediately and shows a confirmation.

4.3. Settings File

To remember your choices between individual runs (for example, the path to the export folder or the name of the PDF preset), the script saves them to the configuration file **ExportPDF-Corel.settings.ini**.

This is a text file in INI format that the script manages automatically. You do not need to edit this file manually — all settings are made through the script's dialogs.

File location:

- **Windows:** The file is located in the user data folder. The full path typically looks like this:
`C:\Users\YourName\AppData\Roaming\AlesUlrychScripts\CorelDraw`

What settings are saved?

The *ExportPDF-Corel.settings.ini* file saves, for example:

- Path to the main export folder.
- Name of the preferred PDF preset.
- Templates for automatic file and folder naming.
- Reference field selection.
- Your symbol mappings for manual diacritics and graphic numbers.
- Custom column names for your CSV file.
- Last selected user interface language.
- Layout mode (page or group).
- Laser contour settings (offset in mm).
- Glyph check skip option.

4.4. Profile Management: Saving and Loading Complex Settings

Profile management allows you to save a complete set of advanced settings for different types of orders and easily switch between them. You no longer need to reconfigure everything for each product type (e.g., "hoodies and caps" vs. "towels with logo").

All profile management tools can be found in the **"Advanced Parameter Settings"** dialog at the top of the window.

Profile management features:

- **Save profile...:** Saves the current configuration from all tabs of the dialog (product names, CSV mapping, graphics settings, etc.) to a file with a name of your choice.
- **Load profile...:** Displays a list of saved profiles. After selecting one, all values in the "Advanced Parameter Settings" dialog are immediately updated according to the saved profile.
- **Import...:** Allows you to import profile files (ending in .json) that you may have transferred from another computer.
- **Export...:** Allows you to select one of the saved profiles and export it as a *.json* file that you can back up or share with colleagues.

Quick Start for Experienced Users

This section serves as an accelerated guide for users who are already familiar with the basic principles of the script. For a detailed description of individual features, please refer to the following chapters.

Step 1: Preparing the Template in CorelDraw

1. **Open the .cdr file** with your product template.
2. **Check page names:** Make sure they are correctly named for automatic recognition, e.g., *mikina-XL*. Remember that prefixes (e.g., *mikina-*, *dresy-*, *trenky-*, ...) can be changed in advanced settings.
3. **Check text field names:** Verify that dynamic text fields have correct names, e.g., *NameText_M*, *NumberBack_L*, *NicknameText*. The script is case-sensitive.

Step 2: Preparing Data in the CSV File

1. **Prepare your .csv file** with personalization data. Make sure the first row contains column names (header).
2. **Check column names:** Default expected names are *cislo*, *jmeno*, *prezdivka*, *velikost-mikiny*, etc. Custom names can be permanently set in advanced settings.
3. **For multi-line text**, use the separator || in the cell (e.g., Jan||Novák).

Step 3: Launching the Script

1. In CorelDraw, go to menu **Tools > Scripts > Run Script...**
2. Find and open the script file (e.g., ExportPDF_Corel_spustit.ExportPDF).

Step 4: Main Process Settings

1. After launching, the **"Script Processing Preferences"** dialog will appear.
2. Choose key options for this export:
   - Do you want to use **graphic numbers**? (i.e., replace text digits 0–9 with provided images/symbols)
   - Does your font support **diacritics**, or should the script create them manually?
   - Do you want to insert a **logo into numbers**?
   - Will you be processing the **nickname** column?
   - Do you need to display **specific graphics** for some products (e.g., sponsor logo)?

Step 5: Export Settings and Launch

1. After confirming preferences, the **export settings dialog** will appear (in page mode).
2. If necessary, **map pages** to sizes.
3. **Set the target folder** for export and select the **PDF preset**.
4. Check and optionally adjust **file and folder name templates**.
5. Click **"OK"** to start the automatic export.

5. Main Preferences Dialog

After launching the script and the optional language selection dialog, the **"Script Processing Preferences"** dialog will appear. Here you set the key options for the current script run.

Options in this dialog:

Panel: Basic Settings

- **Use graphic symbols for numbers:** When checked, the following buttons become available:
  - **"1. Select reference field...":** This button opens a dialog where you define the main text fields whose font size will serve as the 100% reference for scaling graphic numbers (e.g., *NumberBack_M*).
  - **"Map symbols for graphic numbers...":** This button, active only after selecting a reference field, opens the familiar dialog for assigning symbols to individual digits.
- **Force manual CSV column mapping:** When checked, the CSV column mapping dialog will always appear, even if automatic mapping finds all required fields.
- **Use separate numbers for products:** When checked, the script will offer mapping of separate numbers for your chosen products.
- **Process nickname column:** When selected, the script allows inserting both a name and nickname on a single product.
- **Process optional graphics column:** When checked, the script will look for corresponding graphic elements based on values in the CSV optional graphics column.
- **Convert names/nicknames to UPPERCASE:** The content of the name/nickname text field will be converted to uppercase. **If manual diacritics is activated (see below), the option for names will be automatically checked and disabled.**

Panel: Diacritics settings for names

- **The font used for names supports diacritics:**
  - **Checked (default):** The script assumes the font correctly displays diacritics. The option to convert names to uppercase is fully editable.
  - **Unchecked:** The script will use the manual method of placing diacritics using per-character placeholders. The name “Novák” will become “NOVAK” in the text field and symbols (caron, acute, etc.) will be placed above the respective letters according to positions defined in placeholders. **When unchecked, the option “Convert names to UPPERCASE” is automatically enabled and locked, because manual diacritics is optimized for capital letters.**

When unchecked, the following buttons become available:

- **“1. Select reference field...”:** This button opens a dialog where you define the main text fields whose font size will serve as the 100% reference for diacritics scaling (e.g. *NameText_M*).
- **“Map diacritics symbols”:** Opens a dialog for assigning symbols to diacritics types. Now supports 10 symbol types (caron, acute, ring, mäkčeň, circumflex, umlaut, ogonek, dot above, stroke, eszett). Set unneeded types to “Not used”.
- **“2. Create/Update placeholders”:** Creates reference letters with diacritical symbols on a special layer **Diacritics_Placeholders**. The user then visually adjusts the position and size of each symbol. The script uses these positions during export.

Panel: Logo in Numbers Settings (on back)

- **Insert logo into numbers:** Checking this option activates the function that inserts the selected logo symbol directly into the numbers on the product.
- *Note: This function cannot be used simultaneously with the "Use graphic symbols for numbers" option.*

- When checked, the **"1. Set symbol and reference..."** button becomes available. Clicking it opens a dialog where you select the **logo symbol** and also specify the **reference text field** (e.g., *NumberBack_M*), whose font size the script will use as the 100% reference for proper logo scaling.

Panel: Advanced Settings

- **Edit advanced parameters...:** Opens the "Advanced Parameter Settings" dialog (see below).

Panel: Language & Info

- **Change language...:** Opens the dialog for changing the script language.
- **About script...:** Displays the dialog with script and license information.

Note: Your reference field selection is **automatically saved** for the next script run. You don't need to set them each time, only when you want to use a different reference.

5.1. Dialog: Symbol Mapping for Graphic Numbers

- **Purpose:** Assign symbols from the CorelDraw document to digits (0–9) for front and back numbers.
- **Usage:** For each digit and number type, select a symbol name from the dropdown list. If you choose "Do not use", the graphic number will not be generated.
- Settings are saved to *ExportPDF-Corel.settings.ini*.

Important: All symbols must be placed in a layer named "Symboly". It is not enough to have them only in the CorelDraw symbols docker — the script searches directly in the "Symboly" layer in the document.

5.2. Dialog: Diacritics Symbol Mapping

- **Purpose:** Assign symbols from the CorelDraw document for diacritical marks. The new version supports 10 symbol types for 4 languages.
- **Usage:** For each diacritics type, select the corresponding symbol from the document. Set unneeded types to “Not used”. Default symbol names are listed in the table below.
- The dialog automatically assigns symbols based on their names in the document. On subsequent runs, the mapping is loaded from saved settings.
- Settings are saved to *ExportPDF-Corel.settings.ini*.

Supported diacritical symbol types:

Symbol type	Default name	Letters	Languages	Placement
Caron (ˇ)	hacek	Š, Č, Ř, Ž, Ň, Ď, Ť, Ě	CZ, SK	Above letter
Acute (´)	carka	Á, É, Í, Ó, Ú, Ý, Ĺ, Ŕ, Ś, Ć, Ń, Ź	CZ, SK, PL	Above letter
Ring (˚)	krouzek	Ů	CZ	Above letter
Mäkčeň (SK caron)	makcen	Ľ, (Ď, Ť in SK)	SK	Upper right (apostrophe)
Circumflex (ˆ)	circumflex	Ô	SK	Above letter
Umlaut (¨)	umlaut	Ä, Ö, Ü	DE	Above letter
Ogonek (˛)	ogonek	Ą, Ę	PL	Below letter
Dot above	dot-above	Ż	PL	Above letter
Stroke	stroke	Ł	PL	Through center
Eszett	eszett	ß	DE	Replaces entire letter

- **Intelligent Ď/Ť assignment:** Letters Ď and Ť are automatically assigned to the correct symbol based on script language — caron in Czech, mäkčeň in Slovak.

- **Special handling of ß (eszett):** Eszett is not a classic diacritical mark but an entire letter. The script replaces ß with an invisible B (to maintain text proportions) and places the eszett symbol at its position.

5.3. Per-Character Diacritics Placeholders

The new placeholder system allows you to visually set the exact position and size of a diacritical symbol for each letter individually, directly in CorelDraw.

How it works:

- 1. On the first run with manual diacritics active, the script offers to create placeholders.
- 2. On a special non-printable layer Diacritics_Placeholders, reference letters (C, S, A, U, ...) with diacritical symbols are created.
- 3. The user visually adjusts the position and size of each symbol to match the desired appearance on the jersey.
- 4. On subsequent runs, the script reads positions and sizes from placeholders and uses them during export.
- 5. Positions and sizes are automatically scaled based on text size and rotation.

Advantages over the old system:

- Precise placement for each letter individually (Š vs Í have different offsets).
- Visual control directly in CorelDraw — WYSIWYG.
- One-time setup — then works automatically for all exports.
- Support for rotated text (0°, 90°, -90°, 180°).
- Automatic scaling for different sizes (adult/children).
- Symbol size from placeholders is transferred — smaller caron for narrow letters (I), larger for wide ones (Š).

Setting up placeholders — step by step:

- 1. Uncheck “The font used for names supports diacritics” (= enable manual diacritics).
- 2. Click “Map diacritics symbols” and assign symbols to diacritics types.
- 3. Select the reference text field.
- 4. Click “Create/Update placeholders”.
- 5. The script creates the Diacritics_Placeholders layer with reference letters and symbols.
- 6. Find the Diacritics_Placeholders layer and adjust the POSITION and optionally SIZE of each diacritical symbol for each letter.
- 7. Save the document. On the next export, the script will automatically use the positions.

Position and size from placeholders: The script uses both position and size of the symbol from placeholders during export. Size is automatically scaled proportionally based on text size (current/reference font ratio). This also applies to logo in number.

Placeholder layout:

Symbol groups are arranged side by side in rows with labels (HACEK, CARKA, UMLAUT, ...). Automatic wrapping when exceeding page width. Ogonek is placed below the letter, stroke and eszett through the center.

Warnings and checks:

- If placeholder layer doesn’t exist → offers to create it.
- If some placeholders are missing → warning with list of missing letters.
- If mapped symbols have changed → mismatch warning.
- User can continue (missing letters won’t have diacritics) or stop the script.

Default positions:

When creating placeholders, the vertical and horizontal offset values from the advanced dialog are used as default positions. The user then fine-tunes individual letters.

Supported Languages and Characters

- **Czech (CZ):** Á, Č, Ď, É, Ě, Í, Ň, Ó, Ř, Š, Ť, Ú, Ů, Ý, Ž
  - → *hacek, carka, krouzek*
- **Slovak (SK):** Á, Č, Ď, É, Í, Ĺ, Ľ, Ň, Ó, Ô, Ŕ, Š, Ť, Ú, Ý, Ž
  - → *hacek, carka, makcen, circumflex*
- **German (DE):** Ä, Ö, Ü, ß
  - → *umlaut, eszett*
- **Polish (PL):** Ą, Ć, Ę, Ł, Ń, Ó, Ś, Ź, Ż
  - → *carka, ogonek, stroke, dot-above*

5.4. Dialog: Advanced Parameter Settings

This dialog is the center for detailed configuration of script behavior.

Tab: Products and Sizes

- **Product Identification:** Here you define how the script recognizes your products. For each of the two products, you configure:
  - **Page prefix:** Technical identifier by which the script assigns a page to a product (e.g., *mikina-*).
  - **Display name:** Product name that you will see in the script interface (e.g., "Hoodies").
- **Custom sizes:** In this field, you can enter any atypical sizes separated by commas (e.g., *50x30*, *taska*, *1-2 roky*). These sizes will then appear in the selection when mapping pages and will be saved for the next launch.

Tab: CSV Mapping

In this tab, you can permanently change the expected column names in your CSV file. The script will remember this setting. It is divided into three sections:

- **Common fields:** For columns used in "simple" mode (when you don't have the "Use separate fields..." option checked).
- **Product 1 / Product 2:** For specific column names for each product, used in "separate fields" mode.

Tab: Graphic Functions

- **Use graphic numbers:** Allows you to enable or disable the graphic numbers function separately for each product. This option is active only when the global "Use graphic symbols for numbers" option is checked in the main dialog.
- **Use logo in number:** Just like graphic numbers, you can also enable or disable this function for each product separately.

- **Basic diacritics offsets:** Allows setting default vertical and horizontal offset for diacritical marks. These values are used as default positions when creating per-character placeholders. After creating placeholders, these values are replaced by the precise positions from the placeholders.
- **Logo reference element spacing:** Setting the horizontal spacing between automatically generated digits for dynamic logo placement.
- **Skip glyph check (SkipGlyphCheck):** When checked, the script will skip the automatic validation of whether the font contains all necessary characters. This is useful if you know your font is fine or if the check causes slowdowns with large batches.

For the detailed procedure on how to fine-tune logo positions, see Chapter **8.5 "Dynamic Logo Placement in Numbers"**.

6. Custom Product Configuration

One of the most powerful features of the script is the ability to fully customize which products you will process. You are no longer limited to just "jerseys and shorts." You can configure the script for any combination of two products, for example **hoodies and caps**, **t-shirts and shorts**, or even **mugs and coasters**.

This chapter guides you through the process of telling the script exactly what you want to produce.

Where to find the settings?

All options for defining custom products can be found in the main dialog under the **Edit advanced parameters...** button in the **Product Identification** section.

Parameter explanation

For each of the two products, you configure two key properties:

1. Page prefix (technical identifier)

- **Purpose:** This is technical text by which the script recognizes which page belongs to which product. The script simply checks whether the page name starts with this text.
- **Example:** If you set *mikina-* here, the script will consider all pages with names starting with "mikina-" (e.g., *mikina-M*, *mikina-L*) as "product 1".

2. Display name (interface label)

- **Purpose:** This is the name you will see throughout the script's user interface — in dialogs, error messages, and the final summary. It serves for your clarity.
- **Example:** If you set *Hoodies* here, the CSV mapping dialog will show text like "Size (Hoodies)".

Complete Example: From Jerseys to Hoodies and Caps

Let's show how to completely reconfigure the script from default jerseys and shorts to hoodies and caps.

Step 1: Goal. We want to export personalized hoodies (product 1) and caps (product 2).

Step 2: Script settings. Open **Advanced Parameter Settings** and fill in the **Product Identification** section as follows:

- *Product 1 Page Prefix*: **mikina-**
- *Product 1 Display Name*: **Hoodie**
- *Product 2 Page Prefix*: **cepice-**
- *Product 2 Display Name*: **Cap**

Step 3: Preparation in CorelDraw. Now we need to ensure our pages match the newly set prefixes. Rename them, for example:

- *mikina-M*

- *mikina-L*
- *mikina-XL*
- *cepice-univerzalni*

Step 4: Result. Done! From now on, the entire script will work with your new products:

- The CSV column mapping dialog will display labels like **"Size (Hoodie)"** or **"Code (Cap)"**.
- The final summary report will inform about the number of exported products, e.g.: **"Number of exported files (Hoodie): 5 of 5"**.
- The script will automatically know that page *mikina-L* belongs to the "Hoodie" product.

Important tips and notes:

- **Hyphen in prefix:** We recommend that the prefix always ends with a hyphen (-) for better readability of page names (e.g., *mikina-M*). The script will automatically add it if missing.
- **Consistency is key:** The names you set must match how you name the pages.
- **CSV columns:** Don't forget to rename the expected CSV column names in the same part of the dialog, e.g., *velikost-dresy* to *velikost-mikiny*.

7. CSV File Preparation

- **Format:** Standard CSV (comma **,** as separator).
- **Encoding:** UTF-8.
- **First row (header):** Must contain column names.
- **Columns:**
  - **cislo**: Player/product number. Can be empty if no number is needed for the given product.
  - **cislo-mikiny**: Numbers for hoodies, if different from caps.
  - **cislo-cepice**: Numbers for caps, if different from hoodies.
  - **jmeno**: Player name.
  - **prezdivka**: Player nickname or other text.
  - **volitelna-grafika**: Display of optional (conditional) graphics.
  - **velikost-mikiny / velikost-cepice**: Product size. At least one of these sizes must be present for processing a row in complex mode.
  - **kod-mikiny / kod-cepice**: Product code (optional).
  - *Note: The actual expected column names depend on your mapping settings (see Chapter 6).*

Note: If the CSV file is missing a value for name, nickname, or code, the corresponding text fields in the exported PDF will be empty.

7.1. CSV Column Mapping

- **Automatic mapping:** The script will attempt to find matches between CSV column names and expected names (from *ExportPDF-Corel.settings.ini* or defaults).
- **"CSV Column Mapping" dialog:** Appears when mapping is forced or when key columns were not automatically mapped. Allows manual assignment.
- **Permanent custom column names:** Can be set in the "Advanced Parameter Settings" dialog (see 5.3).

7.2. Inserting Multi-Line Text (Names, Nicknames)

The script supports inserting multi-line text into a single text field. This is ideal, for example, when you need to place a first name and below it a surname on a product.

How to do it?

Very simply. In the CSV file, insert **two vertical separators ||** between words you want to break to a new line in the name (or nickname) cell.

Example:

- **Entry in CSV file in the jmeno column:** `Jan||Novák`
- **Result in the NameText text field on the product:**

JAN

NOVÁK

This procedure works for both the *jmeno* and *prezdivka* columns.

The vertical separator |, also called a pipe, is typed using the following keyboard shortcuts depending on your keyboard layout:

- **On Czech keyboard (QWERTZ):** The most common shortcut is **AltGr + W**. AltGr is the key to the right of the spacebar (sometimes labeled as right Alt).
- **On English keyboard (QWERTY):** Press **Shift + **. The backslash key \ is usually located above the Enter key.
- **Using the numeric keypad (Windows):** Hold the left **Alt** and type the number **124** on the numeric keypad (on the right). After releasing the Alt key, the character will appear.

Intelligent formatting preservation: The script contains an advanced formatting style cache (TwoLineStyle) that remembers font, size, bold, italic, and underline separately for the first and second line of each text field. If your template has a text field with multiple lines and each has a different style (e.g., first line bold, second normal), the script will precisely apply this formatting to the newly inserted text from CSV. If the template has only one line, its style is used for all newly created lines.

7.3. Using Separate vs. Shared Numbers

In the script's introductory dialog, you will find the option "Use separate numbers for products...". This setting changes how the script works with the number column in CSV:

- **If the option is NOT checked (default):** The script expects one shared column for the number (default name *cislo*) and uses its value for both products.
- **If the option IS CHECKED:** The script will ignore the shared *cislo* column and will look for specific columns for each product (e.g., *cislo-mikiny* and *cislo-cepice*). This allows having different numbers for hoodie and cap on the same CSV row.

7.3.1. Using Superscript, Subscript, and Escaping

Where superscript or subscript is needed, insert an underscore or caret into the CSV.

- For water (H_2O) write: **H_2O**
- For square meters (m^2) write: **m^2**
- For CO_2 write: **CO_2**

The script automatically converts these to the correct typographic characters on import.

If a square or plain digit still appears in CorelDraw, it means only one thing: **the font doesn't support these characters** and a different font must be chosen (or use the optional graphics method).

The script supports **"escaping"** (also called "escape sequences"). If you write a **backslash ** before the _ or ^ character, the script will ignore it and output the character as-is.

The rules are as follows:

- **H_2O** → H_2O (Normal behavior: creates subscript)
- **Part_2** → Part_2 (Backslash "protects" the underscore, no subscript created)
- **m^2** → m^2 (Normal behavior: creates superscript)
- **\^** → ^ (Prints the caret itself)

- **C:\\Folder** → C:\Folder (To write a single backslash, type two \)

Note: Typing the caret ^ can be done **using the numeric keypad (Windows):** Hold the left **Alt** and type the number **94** on the numeric keypad (on the right). After releasing the Alt key, the character **^** will appear.

Typing the backslash \ can be done:

- **On Czech keyboard (QWERTZ):** The most common shortcut is **AltGr + Q**. AltGr is the key to the right of the spacebar (sometimes labeled as right Alt).
- **Using the numeric keypad (Windows):** Hold the left Alt and type the number **92** on the numeric keypad (on the right). After releasing the Alt key, the character \ will appear.

7.4. Displaying Optional (Conditional) Graphics

The script allows showing or hiding a specific graphic element (e.g., sponsor logo, anniversary mark, certification) based on a text value in your CSV file. This feature significantly increases personalization flexibility.

How does it work?

The principle is simple: If you fill in the name of a graphic (e.g., "ŠKODA") in a special column for a given product in the CSV file, the script will try to find the corresponding named object in CorelDraw and make it visible before export.

Step 1: CSV File Preparation

Add a new column to your CSV file. The default expected name is **volitelna-grafika**, but you can rename it in advanced settings.

- For products that should **not** have any extra graphics, leave the cell in this column **empty**.
- For products that **should** have specific graphics, enter its **base name** in the cell (e.g., ŠKODA or Hyundai).

Step 2: Template Preparation in CorelDraw

All optional graphic variants must be prepared in a single layer.

1. **Create a new layer** and name it exactly **Volitelna_Grafika**.
2. Place all graphic variants (sponsor logos, marks, etc.) into this layer.
3. **Group (Ctrl+G)** each graphic element separately.
4. **Name each group** according to the rule: **NAME_SIZE**.
   - The name must exactly match the text from CSV.
   - The size must match the product size.
   - **Example:** For sponsor "ŠKODA" and size "M", the group name must be *ŠKODA_M*. For "Hyundai" and size "L", it will be *Hyundai_L*.
5. Initially, all these groups in the *Volitelna_Grafika* layer must be **hidden**. The script manages their visibility on its own — it shows the needed graphic before export and hides it again afterward.

Step 3: Activation in the Script

When launching the script in the main **"Script Processing Preferences"** dialog, check the new option:

- **"Process optional graphics column"**

This tells the script to look for the volitelna-grafika column (or your renamed version) and try to make the corresponding graphic visible based on its content. If the script doesn't find the corresponding object (e.g., *ŠKODA_M*), it will log a warning but will continue normally with the export without that graphic.

8. CorelDraw Template Preparation

8.1. Text Fields

The script identifies text fields by their **names**.

These text field names are case-sensitive — the name must match exactly.

Important text fields (optional, the script will create them on a hidden layer if missing):

- *NumberText*: General field for number.
- *NameText*: For name.
- *NicknameText*: For nickname.
- *NumberBack*, *NumberFront*: For back and front numbers (if not using graphic numbers).
- *VelikostDresyText*, *KodDresyText*, *VelikostTrenkyText*, *KodTrenkyText*.

Size-specific fields (take priority):

- E.g., *NameText_XL*, *NumberBack_L*.

Location: The script primarily works with text fields on the currently active page.

Rotated text support: The script reliably supports correct placement of manual diacritics and logos in numbers even on text fields rotated by 90 or 180 degrees. This gives you greater freedom in template design.

8.1.1. Proper CorelDraw Preparation for the Script — Basic Setup

- Text fields to be updated according to the **cislo** column should be named ***NumberText***.
- Text fields to be updated according to the **jmeno** column should be named ***NameText***.
- Text fields to be updated according to the **prezdivka** column should be named ***NicknameText***.

8.1.2. Proper CorelDraw Preparation for the Script — Advanced Setup

Advanced script features such as inserting graphic numbers, logos, or manual diacritics use a "reference field" system. The principle is simple: the script needs to know what the 100% size of a given element should look like so it can derive the correct size for other variants (e.g., shrink the logo for size S or enlarge for XXL). It uses the font size from a text field that you choose as this 100% reference — typically the field for the medium size, e.g., *NumberBack_M* or *NameText_M*. Therefore, correct naming and setup of these fields is key to the proper functioning of advanced features.

- **Graphic symbols for numbers** — for this option, name the text field for **front number** ***NumberFront_size***, for back numbers ***NumberBack_size***, i.e., for size *M* it will be called ***NumberFront_M*** and ***NumberBack_M***, for size *L* the text field names should be ***NumberFront_L*** and ***NumberBack_L***, etc.
- **Font used doesn't support diacritics** — for this option, name the text field ***NameText_size***, i.e., for size *M* you need the name ***NameText_M***, for size *L* the name ***NameText_L***, etc.
- **Insert logo into numbers** — for this option, name the text field for the back number ***NumberBack_size***, i.e., for size *L* it will be ***NumberBack_L***, for size *M* name it ***NumberBack_M***, etc.
- **Warning:** For the nickname text field (*NicknameText*) and general number (*NumberText*), size-specific naming is not used (e.g., *NicknameText_M*). The script always updates **all** fields with this name found on the active page.

Key text fields overview:

Function	General name	Specific name (example for XL)	Note
Name	NameText	NameText_XL	Script looks for *NameText_XL* first. If not found, uses

			NameText. Required for diacritics.
Nickname	NicknameText	*not used*	All fields named *NicknameText* on the active page are always updated.
Back number	NumberBack	NumberBack_XL	Looks for *NumberBack_XL* first, then *NumberBack*. Key for graphic numbers and logo.
Front number	NumberFront	NumberFront_XL	Looks for *NumberFront_XL* first, then *NumberFront*. For front graphic numbers.
General number	NumberText	*not used*	Used for numbers outside the main ones (e.g., on sleeve) or in simple mode.

Important:

- **Rotated text:** The script also supports text fields rotated by 90 or 180 degrees.

8.2. Pages

For each product type and size, a separate page should exist in the CorelDraw document.

- **Naming:** For automatic recognition, we recommend names in the format *YOUR_PREFIX-SIZE* (e.g., *mikina-XL*). You can set the prefix in the "Advanced Parameter Settings" dialog (see Chapter 5.3). Default values are *dresy-* and *trenky-*.
- **Important:** Each page corresponds to one size of one product. The script automatically activates the correct page when processing a given CSV row. If you're working in group mode (see Chapter 9.1.1), pages are not used — named groups are used instead.

8.3. Symbols (for Graphic Numbers, Diacritics, Logos)

If you use features requiring symbols, they must be present **in the "Symboly" layer** in the CorelDraw document.

Important: The script looks for symbols primarily in a layer named exactly **"Symboly"**. It is not enough to have symbols only in the CorelDraw symbols docker/panel. Place them directly in the layer with this name. If the "Symboly" layer exists, the script uses it as a fast path for searching. If the layer is missing, the script performs a complete search of the entire document, which is slower.

Types of symbols:

- **Graphic numbers:** Symbols for digits 0–9 (e.g., *cislo_zadni_0*, *cislo_predni_0*).
- **Manual diacritics:** Symbols for diacritical marks. 10 types are supported: *hacek, carka, krouzek, makcen, circumflex, umlaut, ogonek, dot-above, stroke, eszett*. Activate only the ones you need — set others to “Not used” in the mapping dialog.
- **Logo in number:** Symbol for your logo.

8.4. Layers Used by the Script

The script works with several special layers in the CorelDraw document. Some are created automatically, others you must create yourself.

Layers you must create (if using the given feature):

- **Symboly** — Layer for all symbols (graphic numbers, diacritics, logo). The script reads symbols from this layer. This layer must exist if you use any symbol-based feature.
- **Volitelna_Grafika** — Layer for optional graphic elements (sponsor logos, etc.) that are conditionally displayed based on CSV data. Named groups in this layer follow the *NAME_SIZE* convention.

Layers the script creates automatically:

- **Diakritika_Skript** — The script inserts manually generated diacritics (carons, acute accents, ring accents) here. This layer is automatically cleaned before each export.
- **Graficka_Cisla** — Layer for graphic digits inserted as symbols instead of text digits. The script creates and manages it automatically.
- **Logo_Placeholders** (name may vary by language) — Layer with reference digits and placement symbols for dynamic logo placement in numbers. The script creates it automatically when the "Insert logo into numbers" feature is first activated.
- **Diacritics_Placeholders** (non-printable) — Layer with reference letters and diacritical symbols for the per-character placeholder system. The script creates it automatically when the placeholder function is activated. Symbol positions and sizes on this layer determine the exact placement of diacritics during export.
- **Laser_Kontury** — Layer for automatically generated laser contours (see Chapter 8.8). Created when the laser contours feature is run.

8.5. Dynamic Logo Placement in Numbers

The script completely changes and simplifies the preparation for dynamic logo insertion. The process is now highly automated and intuitive.

How does it work?

The entire principle is built on working with the logo visually without having to guess anything.

1. Your preparation (what you need to do in the CorelDraw file)

Before running the script, make sure you have two things properly prepared in the template:

- **Reference text field:** A text field for the back number must exist (e.g., named *NumberBack_M*). This field must have **exactly the font and size** that the final numbers on the product should have.
- **Logo symbol:** Your logo must be stored as a symbol in the document (in the "Symboly" layer).

2. Automatic script action (what the script does itself)

When you check the **"Insert logo into numbers"** option in the main dialog and select your symbol and reference field, after clicking "OK" the script performs the following:

- **Checks for placeholder existence:** It looks at the *Logo_Placeholders* layer to see if the necessary elements already exist.
- **Automatically creates them if missing:** If it doesn't find the elements, it prepares them for you:
  - Creates reference digits (*ref-0* through *ref-9*) that **automatically adopt the font and size** from your reference field (*NumberBack_M*).
  - Creates so-called **placement symbols** (*0-umisteni* through *9-umisteni*) that are **direct instances of your selected logo**.

3. Your final fine-tuning (your only and easy task)

After automatic creation, your task is extremely simple because you work directly with the visual appearance of your logo.

- On the *Logo_Placeholders* layer, find the instances of your logo (named *0-umisteni* through *9-umisteni*).
- **Move each logo to exactly the position where you want it** relative to the corresponding reference digit.

This eliminates any guesswork — **where you place the logo is exactly where it will be inserted during the final export.**

TIP: The spacing between individual reference digits can be adjusted if needed in the **"Advanced Settings"** dialog. After this one-time setup and saving the file, the script will use these perfectly set positions you configured on every subsequent run.

8.6. Intelligent Dynamic Scaling

The script uses a fully **dynamic system** that intelligently adapts to your template.

How does it work?

- **Reference field principle:** The script reads the font size from your **main reference field** (e.g., *NumberBack_M* for logos or *NameText_M* for diacritics). It considers this size as 100%.
- **Automatic adaptation:** When processing a different size (e.g., XL), the script compares the font size in the *NumberBack_XL* field with the reference size from *NumberBack_M* and calculates the exact ratio (e.g., 115%). It then automatically scales the inserted logo or diacritics by this ratio.

What does this mean for you?

- **Huge simplification:** You no longer need to worry about what is a "children's" and "adult" size. You don't need to try to maintain the same number size across different pages.
- **Complete design freedom:** Simply set the size of numbers and names in the template for each size (M, L, XL, etc.) as needed. The script will ensure that the inserted logo and diacritics always proportionally match.
- **Change in "Advanced Settings":** Values for scaling and offset for children's sizes now serve only as **additional correction** to this dynamic calculation, not as the main method.

8.7. Graphic Numbering (Instead of Fonts) — Using Custom Image Digits

The script offers a unique ability to replace standard text digits in the template with your own graphic symbols. This is ideal in situations where the client has a specific, unconventional digit design (e.g., with effects, textures, or stylization) that cannot be created using a standard font.

How does it work?

The script intelligently reads the requested number (e.g., "21") from your CSV file, breaks it down into individual digits ("2" and "1"), and then inserts the appropriate graphic symbols you have prepared in advance into the template. Thanks to dynamic scaling, the digits automatically adapt to the size of the reference text field.

Step 1: CorelDraw Template Preparation — Your Graphic Digits as Symbols

1. **Create symbols for each digit:** For each digit from *0* to *9*, create a separate graphic object. This can be vector graphics, text converted to curves, or even a raster image.
2. **Add them to the "Symboly" layer:** Drag each graphic object into the "Symboly" layer in CorelDraw.
3. **Name symbols according to the convention:**
   - For **back numbers (on the back of the jersey)**: We recommend naming symbols in the format *cislo_zadni_0*, *cislo_zadni_1*, ..., *cislo_zadni_9*.
   - For **front numbers (on the chest/shoulder of the jersey)**: We recommend naming symbols in the format *cislo_predni_0*, *cislo_predni_1*, ..., *cislo_predni_9*.
   - *(Important: You will assign these symbol names later in the script dialog. If you have the same symbols for front and back numbers, just assign them correctly.)*

Step 2: CorelDraw Template Preparation — Text Fields for Numbers

The script needs to know where to place the graphic digits and where to take the reference size for their scaling. Use standard text fields for this purpose.

1. **For back numbers:** Create a text field where the number should go on the back. We recommend naming it *NumberBack_SIZE* (e.g., *NumberBack_M*, *NumberBack_L*).
   - *Important: The font and font size in this text field determine the scale at which graphic digits will be inserted. Set it exactly as the height of your resulting graphic digits should appear.*
2. **For front numbers:** Similarly, create a text field for the front number. We recommend naming *NumberFront_SIZE* (e.g., *NumberFront_M*, *NumberFront_L*).
   - *Important: Here too, the font and font size define the scale of inserted graphic symbols.*
3. **Reference field setting:** In the script's introductory dialog, you will be prompted to select the main reference field (e.g., *NumberBack_M* or *NumberFront_M*), from which the script determines the "base" 100% font size once. All other graphic digits are then scaled relative to this reference field and the current text field size on the given page (see "Intelligent Dynamic Scaling" chapter).
4. **Script layer:** The script will automatically insert graphic digits on the *Graficka_Cisla* layer. You don't need to worry about this layer — the script creates and manages it automatically.

Step 3: Script Settings — Activation and Mapping

1. **In the main "Script Processing Preferences" dialog:**
   - Check the option **"Use graphic symbols for numbers (instead of text)"**.
   - **Important:** Make sure this option is enabled for your products in the "Advanced Parameter Settings" dialog under the "Use graphic numbers" panel.
2. **Click "1. Select reference field...":** Here in the dialog, select the text field (e.g., *NumberBack_M*) that serves as the main 100% reference for scaling. The button activates after checking the "Use graphic symbols for numbers" option.
3. **Click "Map symbols for graphic numbers...":** This option activates after selecting the reference field. The "Symbol Mapping for Graphic Numbers" dialog will open. For each digit (*0* to *9*) and for each type (*back numbers* and *front numbers*), select the corresponding symbol you created in step 1 from the dropdown list. If you don't select a symbol for a digit ("Do not use"), the graphic number will not be generated for it. Settings are saved to the *ExportPDF-Corel.settings.ini* file.

Step 4: Launching the Export

After completing settings in the "Script Processing Preferences" dialog and the subsequent "Export Settings" dialog, click "OK". The script will automatically process the CSV file, insert graphic digits, and export personalized PDFs.

Important notes and tips for this functionality:

- "Use graphic symbols for numbers" cannot be used simultaneously with the "Insert logo into numbers" option.
- Graphic digits are dynamically scaled according to the font size in the target text field and the selected reference field, just like logo and diacritics (see "Intelligent Dynamic Scaling" chapter). Make sure your graphic symbols for digits are created at a size corresponding to the reference font.

8.8. Laser Contours

The laser contours feature is entirely new and allows automatic generation of cutting/engraving contours around selected objects in the template. This is ideal for preparing files for laser cutting or engraving, where an outline with a defined offset needs to be created around graphic elements.

How does it work?

The script searches the document for all objects whose name starts with the text **"Laser"** (case-insensitive) and creates a visual "ring" around each — a pair of shapes (black outline + white overlay) that together form a contour with the configured offset.

Template preparation:

1. **Name objects:** Any object (shape, group, curve) around which you want to generate a laser contour should be named so that its name starts with the word **"Laser"**. For example: *Laser*, *Laser_logo*, *LaserOutline*. The script recognizes the name regardless of case.
2. **Set the offset:** In the *ExportPDF-Corel.settings.ini* settings file (or through advanced settings), you can set the **LaserContourOffset** parameter — contour offset in millimeters. The default value is 1.5 mm.

Automatic mode detection:

The script automatically detects whether to use page or group mode for generating contours:

- **Page mode:** If the document contains pages named in the format *product-size* (e.g., *mikina-M*), the script uses page mode. Contours are generated on each product page and placed on the **Laser_Kontury** layer, which is automatically created at the bottom position (below other content).
- **Group mode:** If the document does not contain product pages (products are organized as groups), the script uses group mode. Contours are generated inside template groups. Objects named "Laser*" that are not inside any product group are processed separately and their contours are placed on the* Laser_Kontury layer.

Re-run support:

The script automatically removes old contours from previous runs (identified by the name "Laser-řez" or as black curves with uniform fill) before generating new ones, so you can run the function repeatedly without duplicates.

Output:

For each found "Laser" object, the script creates a "Laser-řez" group containing two shapes: a black shape with an outline (width = 2× offset) and a white shape without outline, which together visually represent the contour. A summary message with the number of successfully processed contours is displayed upon completion.

Note: The laser contours feature is launched as a standalone action and does not depend on PDF export. It is a preparatory step that you perform before the actual export.

9. Using the Script — Export Process

9.1. Export Settings (Processing Product 1 and/or Product 2)

- **Panel: Page Mapping:** Assign pages to type (e.g., Hoodies/Caps) and size.
- **Panel: Export Settings:** Main export folder, PDF preset.
- **Panel: Folder and File Name Configuration:** Detailed name template settings using placeholders ({*cislo*}, {*jmeno*}, {*prezdivka*}, {*velikost*}, {*kod*}, {*docName*}). Separately for each product. Includes preview and help.

Explanation of automatic file and folder name generation:

- {*cislo*} — the number from the CSV file will be inserted into the folder or file name.
- {*jmeno*} — the name from the CSV file will be inserted into the folder or file name.
- {*prezdivka*} — the nickname from the CSV file will be inserted into the folder or file name.
- {*velikost*} — the size from the CSV file will be inserted into the folder or file name.
- {*kod*} — the code from the CSV file will be inserted into the folder or file name.
- {*docName*} — the name of the CorelDraw file from which PDFs are generated.

9.1.1. Page vs. Group Mode (Layout Mode)

The script supports two basic working modes with the document, which determine how products and their sizes are organized in the CorelDraw template.

Page mode (page) — default

In page mode, each size of each product corresponds to a separate page in the CorelDraw document. For example, for hoodies in sizes M, L, XL, you will have three pages named *mikina-M*, *mikina-L*, *mikina-XL*.

This mode is ideal for:

- Products with significantly different layouts for different sizes.
- The traditional way of working in CorelDraw, where each size has its own page.
- Simple navigation in the document — each page = one product in one size.

In this mode, the script displays the full export dialog with page mapping, export folder settings, and PDF preset.

Group mode (group)

In group mode, products are organized as **named groups** on one (or more) pages. The group name follows the format *product-size* (e.g., *mikina-M*, *mikina-L*), just as pages would be named in page mode.

This mode is ideal for:

- Situations where you want to have all sizes together on a single page.
- Batch processing where the script automatically duplicates template groups and fills them with CSV data.
- Production where the output is not printed as PDF but processed directly in CorelDraw (e.g., for cutting plotters or printers with their own workflow).

In group mode, the script skips the standard export dialog (does not perform PDF export) and instead directly processes CSV data and applies it to groups in the document. The script automatically creates copies of template groups, fills text fields, and aligns results on the page.

Mode setting:

The mode is set in the **"Advanced Parameters"** dialog (see Chapter 5.3) on the **"Products and Sizes"** tab in the **Working Mode** section. Two options are available: **Page mode** (default) or **Group mode**. The selection is automatically saved to the *ExportPDF-Corel.settings.ini* file and the script recognizes it at each startup.

9.2. CSV Row Processing Flow

For each row and product:

1. **Page activation** (in page mode) or finding the template group (in group mode).
2. **Cleaning dynamic layers** (for graphic numbers, logos, diacritics).
3. **Updating text fields** (name, number, size, code) according to CSV.
   - If **number** data exists in CSV, but no **visible field** for its display is found on the active page, the script reports an error for this specific product and **does not export the product**.
   - If **number** data is not in CSV, existing number fields on the page are cleared (set to empty content "").
4. **Special processing** (diacritics, graphic numbers, logo, optional graphics).
5. **PDF export** according to name templates (page mode only).
6. **Skipping duplicate exports.**
7. **Automatic renaming of duplicate files:** The script includes logic that, in case a file with an already existing name would be generated (within a single run), automatically adds a numeric suffix to the file name, e.g., (1), (2), etc., to prevent overwriting.

9.3. Error States and Logging

- **Log file:** *ExportPDF_Log.txt* in the main export folder. This file contains a detailed record of the export process, including information about processed rows, skipped items, warnings, and errors.

- **Summary report:** After the export is complete, a clear summary dialog is displayed, including the number of successfully exported files, skipped items, and any errors.
- **Missing glyph warnings:** If the script detects during processing that the font used is missing some necessary characters, the warning is displayed in the summary report and details are written to the log.

10. Troubleshooting (FAQ)

Here you will find solutions to the most common problems you may encounter while working with the script.

Problem: The script doesn't work at all or throws an error immediately after launch.

- **Solution:**
  - Make sure you have a document (.cdr file) open in CorelDraw.
  - Check your license status in the "About Script" dialog. Your demo version may have expired.
  - Check the *ExportPDF_Log.txt* file in your export folder — it may contain a more detailed error description.

Problem: The script says it cannot continue and mentions "editing inside a group" or "Focus Mode".

- **Solution:** The script automatically detects whether CorelDraw is in group editing mode (so-called Focus Mode, which occurs e.g., after double-clicking on a group). In this mode, the script cannot correctly work with document pages and layers.
  - **How to exit Focus Mode:** Press the **Escape** key or click outside the group until you return to the main document level. The CorelDraw window title should not display any group name.
  - After exiting Focus Mode, run the script again.

Problem: The script doesn't update text fields for name, number, or code.

- **Solution:** This is the most common problem — please check the following:
  - **Exact field name:** The text field name in CorelDraw must exactly match the expected name (e.g., *NameText*, *NumberBack_M*). Names are case-sensitive.
  - **Visibility and locking:** Neither the field nor the layer it's on must be locked or hidden.
  - **Active page:** The script primarily works with fields on the currently processed page. Make sure the field is on the correct page.

Problem: The script reports error "No visible text field found on page ... for displaying number ...".

- **Solution:** This message means your CSV file **contains a number** for the given product, but the script didn't find a corresponding visible text field in the template. The script will not perform PDF export in this case to prevent manufacturing a product without a number.

How to correctly generate a product without a number? If you intentionally want to generate a product without a number, the procedure is:

- Your CSV file must still contain a column for the number (e.g., *cislo*, *cislo-mikiny*, etc.) because the script expects it.
- For the product that should not have a number, simply leave the cell in this column **empty**.

The script will then generate the product normally, just without a number, and no error will be reported.

Problem: Diacritics are not displayed correctly, or spaces appear instead of text.

- **Solution:** This problem almost always means the font used in your CorelDraw template doesn't contain all necessary characters (glyphs), such as Czech characters like "ř", "ě", or "ů".
  - **Automatic glyph check:** The script includes an advanced automatic check that can detect this problem. The method works by creating test text, converting it to curves, counting nodes, and comparing the result with a reference ".notdef" glyph (placeholder symbol for missing

characters). If the script finds a match, it alerts you in the **final summary report** and writes detailed information to the *ExportPDF_Log.txt* file.

- **Skipping the check:** If you know your font is fine but the check still reports false positives (e.g., with special fonts), you can skip the check by enabling the **SkipGlyphCheck** option in advanced settings ("Graphic Functions" tab).
- **Special fonts:** The script includes special rules for some problematic fonts (e.g., "FaceOff") where standard detection doesn't work reliably.
- **How to fix it:**
  1. **If the font should support diacritics:** Make sure the **"Font used for names supports Czech diacritics"** option is checked in the introductory dialog. If the error persists, the font truly doesn't have all characters, and you need to change the font in the CorelDraw template to one that supports them.
  2. **If the font doesn’t support diacritics (manual method):**
     Uncheck “The font used for names supports diacritics”. The script activates manual diacritics with per-character placeholders. Create placeholders, visually set symbol positions, and run the export. Requires symbols in the “Symbols” layer (see 8.3) and correct mapping in the dialog (see 5.2). If placeholders don’t exist, the script offers to create them.

Problem: Graphic numbers or logo in number are not being inserted.

- **Solution:**
  - **Feature activation:** Make sure you have the corresponding option checked in the introductory dialog ("Use graphic symbols..." or "Insert logo...").
  - **Mutual exclusivity:** These two features cannot be used simultaneously. If one is active, the other must be disabled.
  - **Symbol mapping:** Check that you have correctly mapped symbols in the corresponding dialog.
  - **Reference field existence:** Both features require reference text fields (e.g., *NumberBack_M*, *NameText_M*) for proper functioning and scaling. Without them, they won't work correctly.
  - **"Symboly" layer:** Make sure all symbols are placed directly in a layer named "Symboly" in the CorelDraw document, not just in the symbols docker.

Problem: I changed settings (e.g., CSV column names) but the script still uses the old ones.

- **Solution:** Make sure you confirmed the dialog where you made changes (e.g., "Advanced Parameter Settings") with the "OK" button. Simply closing the window with the X button will not save changes. Settings are permanently written to the *ExportPDF-Corel.settings.ini* file only after this confirmation.

Problem: The script reports duplicate pages in the document.

- **Solution:** The script checks at startup whether the document contains two or more pages with the same name. Duplicate page names can cause unpredictable behavior during export. Rename duplicate pages so each has a unique name.

Problem: A watermark appears on exported PDFs.

- **Solution:** A watermark may appear when the script detects a license integrity issue. Check the license status in the "About Script" dialog and if necessary, perform activation with a valid license key.

Problem: Laser contours are not being generated.

- **Solution:**
  - Check that the objects around which you want contours have a name starting with **"Laser"** (e.g., *Laser*, *Laser_logo*). The name is sensitive only to the first 5 characters "Laser" (case-insensitive).
  - In page mode, "Laser*" objects must be placed on product pages (named in the* product-size format).
  - In group mode, "Laser" objects must be inside product groups or freely on the page.
  - Check that a non-zero offset is set in settings (`LaserContourOffset`).

Problem: The script reports "Your subscription has expired".

- **Solution:** Contact the seller for a new license key. In the "About the Script" dialog, you will find the Machine ID — send it along with your order. Enter the new key via the "Extend license..." button.

Problem: After upgrading to a new version, the script requires a new key.

- **Solution:** Re-enter your existing license key via the "About the Script" dialog → "Activate full version...". A reactivation is sometimes necessary when the version changes.

11. License Information and Activation

License types:

Type	Description	Price
Demo	30 launches, max 20 CSV rows	Free
Monthly subscription	Full access for 30 days	According to the current price list at www.exportPDF.cz
Yearly subscription	Full access for 365 days	According to the current price list at www.exportPDF.cz
Permanent license	Full access without limitations	According to the current price list at www.exportPDF.cz

Activation: Via the "About the Script" dialog → "Activate full version..." → paste your key.

Subscription renewal: Via the "About the Script" dialog → "Extend license..." → paste your new key. Remaining days are automatically added to the new period.

Version upgrade: When updating the script to a new version, you may need to re-enter the license key. The script will prompt you automatically.

Machine ID: A unique identifier of your PC displayed in the "About the Script" dialog. Required for ordering a license — send it to the seller.

12. Contact

For problems, questions, or suggestions for script improvement, contact the author:

- **Author:** Aleš Ulrych
- **E-mail:** info@exportpdf.cz
- **Web:** www.exportPDF.cz
- **Phone:** 776 149 945

Thank you for using the script!