

isCOBOL Evolve: IDE

Integrated Development Environment

Key Topics:

- Configuration
- The isCOBOL IDE Perspective
- Working with Projects
- Working with Screen Programs and File Layouts
- Importing programs from AcuBench



Copyrights

Copyright (c) 2019 Veryant
6390 Greenwich Drive, #225, San Diego, CA 92122, USA

All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution and recompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Veryant and its licensors, if any.

Overview

isCOBOL offers developers the choice of continuing to use their current editor, command line utilities, and scripts for source code control and builds or leveraging the isCOBOL Integrated Development Environment (IDE). Enabling rapid COBOL application development, the isCOBOL IDE includes a COBOL source editor, complete with COBOL syntax highlighting, compiling, and project handling, providing a single, flexible environment for all COBOL development tasks (design, coding, testing and debugging) in the open and extensible Eclipse framework.

The current version of isCOBOL IDE is based on Eclipse 2018-09 and requires Java 8 or greater.

Getting Started

The setup of isCOBOL IDE requires the following steps:

1. [Download and install the Java Development Kit \(JDK\)](#)
2. [Download and install isCOBOL IDE](#)
3. [Activate the License](#)

In order to activate your isCOBOL Evolve products, you will need the e-mail you received from Veryant containing your license key. Contact your Veryant representative for details.

Download and install the Java Development Kit (JDK)

A JDK must be installed on your machine in order to use isCOBOL IDE. For best results and performance, install the latest JDK version available for your platform. isCOBOL Database Bridge is certified to work correctly with both Oracle JDK and OpenJDK from version 8 to version 11.

Self-extracting setups are provided for the Windows platform.

On Unix/Linux platforms Java may be already installed. If it's not the case, you can install it using the appropriate system commands (e.g. yum, or apt-get).

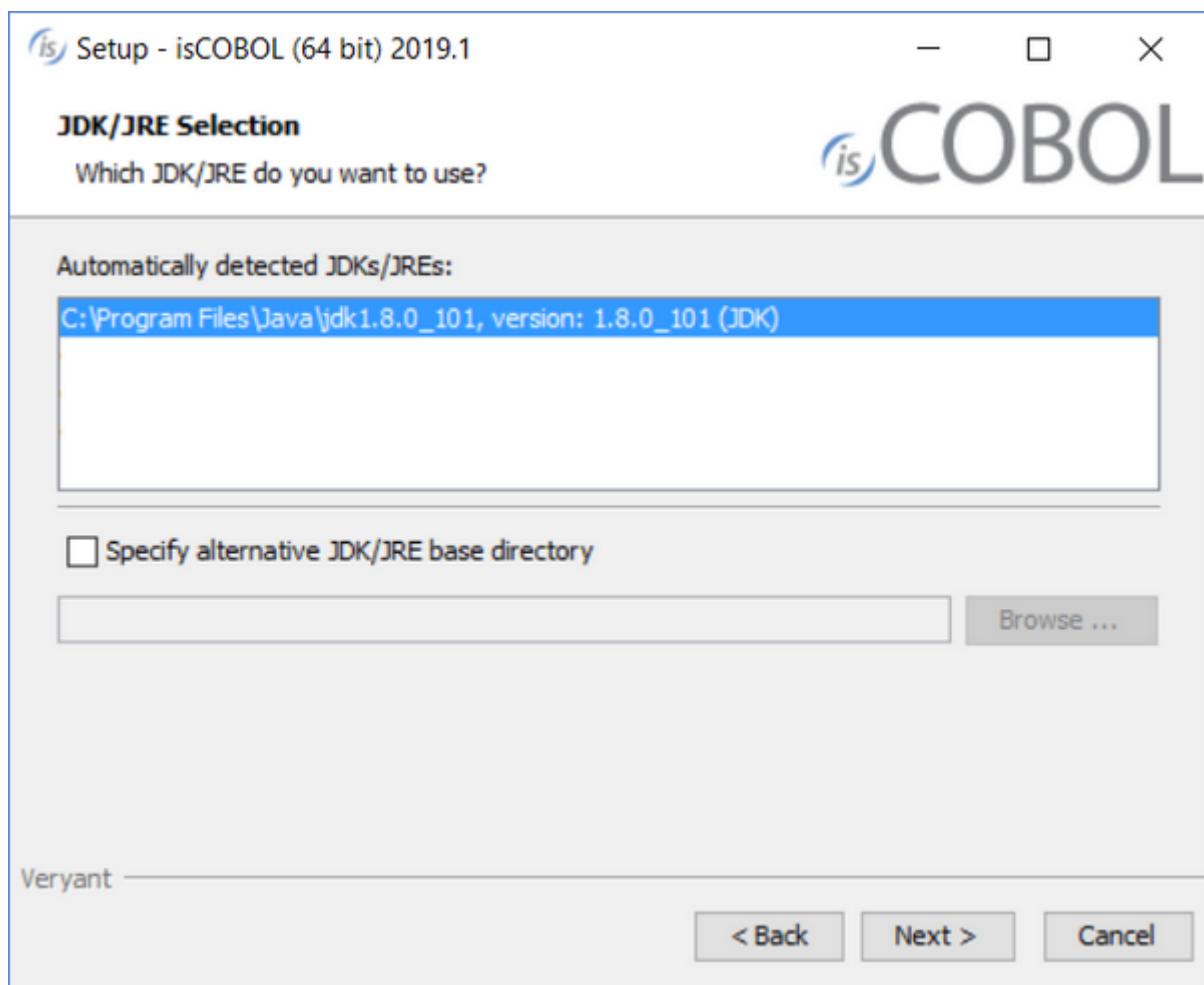
Download and install isCOBOL IDE

Windows

1. If you haven't already done so, [Download and install the Java Development Kit \(JDK\)](#).
2. Go to "<http://www.veryant.com/support>".
3. Sign in with your User ID and Password.
4. Click on the "Download Software" link.
5. Scroll down to the list of files for Windows x64 64-bit or Windows x86 32-bit. Select isCOBOL_IDEyyyy_r_n_Windows_arc.exe, where yyyy is the year, r is the release number, n is the build number and arc is the system architecture.
6. Run the downloaded installer to install the files.

Note - If your Windows has the option "Run as Administrator", you should run the setup with that option.

7. Select your JDK when prompted



8. Follow the wizard procedure to the end. In the process you will be asked to provide the installation path ("C:\Program Files\Veryant" by default) and license keys. You can skip license activation and perform it later, as explained in [Activate the License](#).

Linux and Mac OSX

1. If you haven't already done so, [Download and install the Java Development Kit \(JDK\)](#).
2. Go to "<http://www.veryant.com/support>".
3. Sign in with your User ID and Password.
4. Click on the "Download Software" link.
5. Scroll down, and select the appropriate .tar.gz file for the product and platform you require. Note that isCOBOL IDE is available only for Linux 64 bit and Mac OSX 64 bit.
6. Extract all contents of the archive. For example,
on Linux:

```
gunzip isCOBOL_IDE2019.1_*__Linux_64.tar.gz  
tar -xvf isCOBOL_IDE2019.1_*__Linux_64.tar
```

on Mac OSX:

```
gunzip isCOBOL_IDE2019.1_*__macosx_64.tar.gz  
tar -xvf isCOBOL_IDE2019.1_*__macosx_64.tar
```

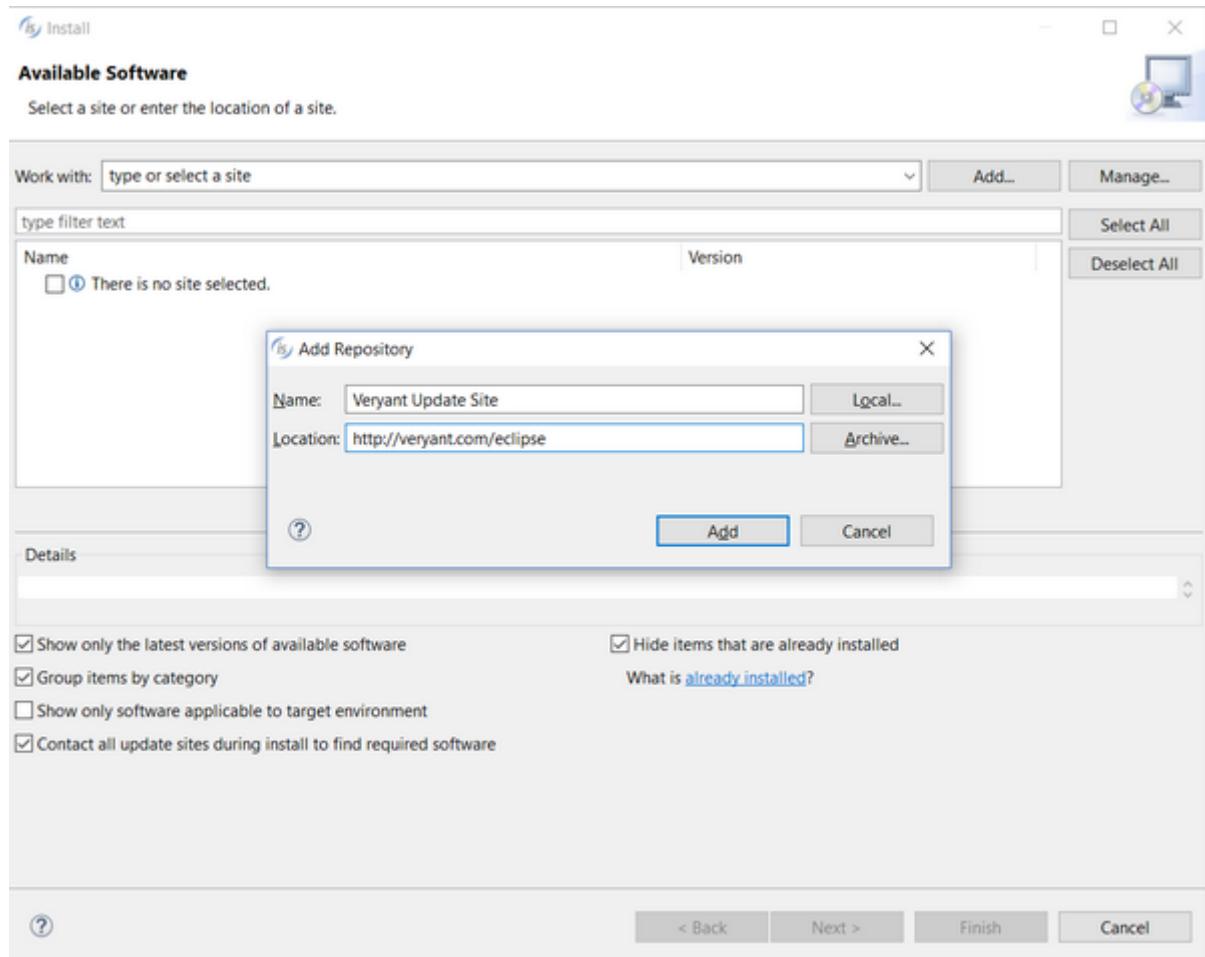
Other

For other platforms or for existing Eclipse environments, it is possible to install the isCOBOL IDE plugins using the Eclipse “Install New Software” feature.

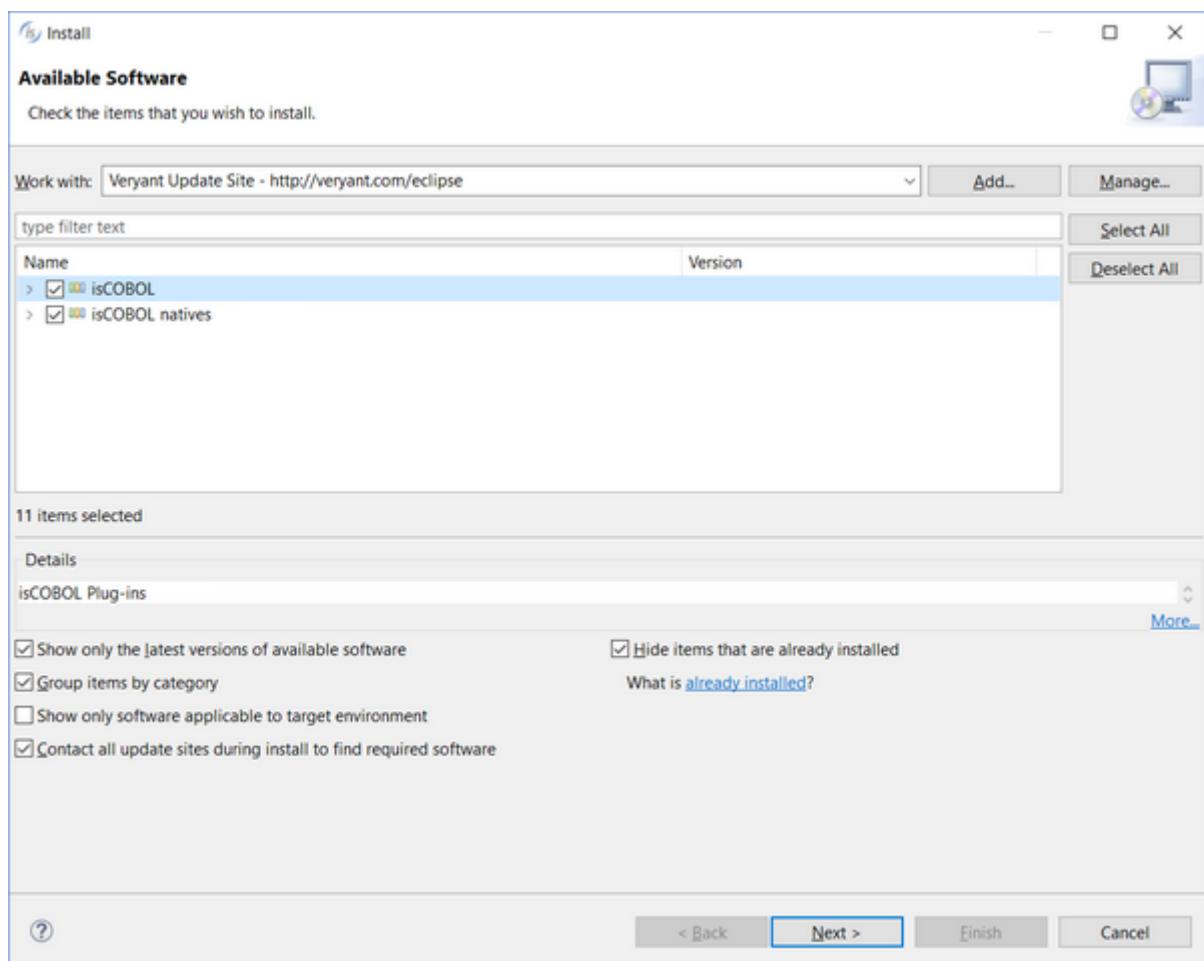
The minimum Eclipse version required to install the isCOBOL IDE plugins is 3.4.2.

To install the isCOBOL IDE plugins in an existing Eclipse environment:

1. Click on *Help* menu and choose *Install New Software...*
2. Click on the *Add...* button
3. Fill in the fields as follows:



4. proceed in the wizard procedure and choose isCOBOL from the list of available products:



How to install a previous release

By filling the *Location* field with "http://veryant.com/eclipse" you obtain the latest IDE release. In order to download previous releases, use this kind of url instead: "http://veryant.com/eclipse/older/v<version>". For example, in order to download isCOBOL IDE 2016 R1, use "http://veryant.com/eclipse/older/v2016R1".

Activate the License

If you provided license keys during the installation, on Windows, you should skip reading this chapter.

isCOBOL Compiler and Runtime included in the IDE look for the following configuration properties for license keys:

```
iscobol.compiler.license.2019=<license_key>
iscobol.license.2019=<license_key>
```

The keys should be stored in one of the following files (if they exist):

Windows

1. \etc\iscobol.properties in the drive where the working directory is
2. C:\Users\<username>\iscobol.properties (the setup wizard saves licenses here, if you don't skip

- activation)
- 3. iscobel.properties found in the Java Classpath
- 4. %ISCOBOL%\iscobel.properties
- 5. a custom configuration file passed on the command line

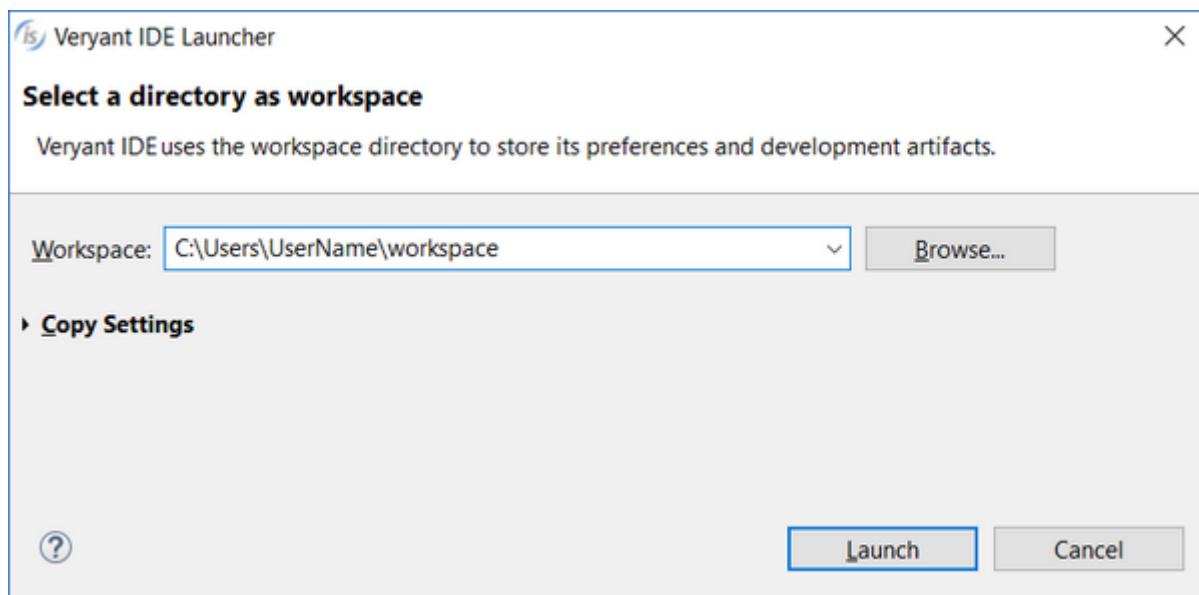
Unix/Linux

- 1. /etc/iscobel.properties
- 2. \$HOME/iscobel.properties
- 3. iscobel.properties found in the Java Classpath
- 4. \$ISCOBOL/iscobel.properties
- 5. a custom configuration file passed on the command line

NOTE - Files are listed in the order they're processed. If the license key appears in more than one of the above files, then the last occurrence is considered.

Starting isCOBOL IDE and selecting the workspace

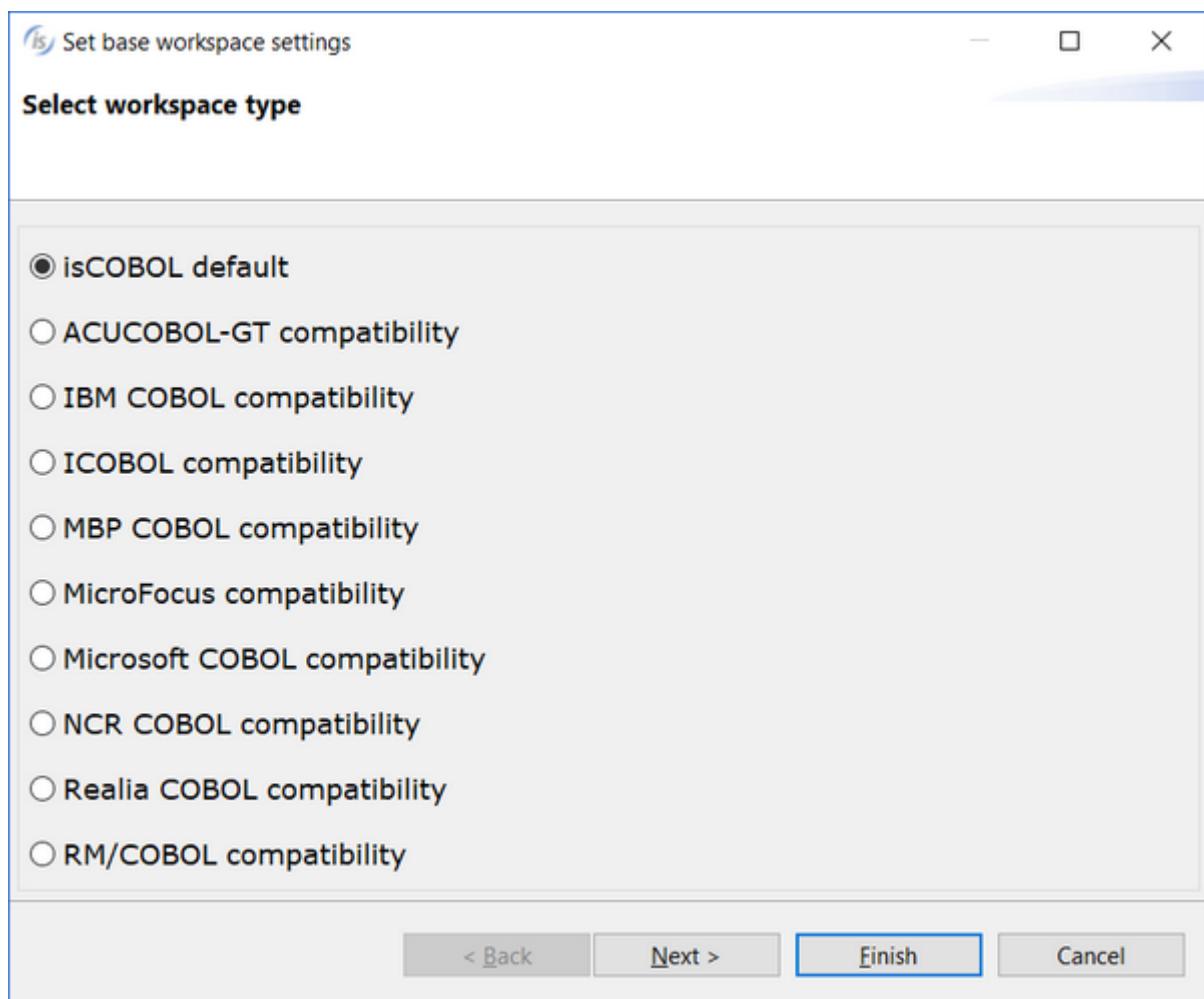
As soon as you start the isCOBOL IDE you're prompted to select the workspace.



The workspace is a physical folder in which the projects created with the IDE are stored.

If you select an existing workspace, the IDE loads the projects of that workspace.

If you select an empty folder, instead, the IDE creates a brand new workspace in there. In this case, you're asked if you wish to have a preset configuration for your projects.



If you're going to maintain a COBOL application that was written with another COBOL, it's good practice to choose the proper preset configuration. If you're going to develop a brand new COBOL application, instead, choose "isCOBOL default".

In the next pages of the wizard you can configure the default options for the projects. This task can also be done later by clicking on *Window* in the menu bar of the IDE and choosing *Preferences*.

See [Customization](#) for details about the available configuration options.

Configuration

This chapter explains how to configure the isCOBOL IDE to match your needs.

Startup options

The isCOBOL IDE runs inside a Java Virtual Machine (JVM). The command-line options used by this JVM at startup are stored in a configuration file named *isIDE.ini*, placed in the product installation directory.

By changing this file it's possible to add new options as well as altering the existing ones.

For example, if you wish to give isCOBOL IDE more memory, alter the value of the -Xmx option.

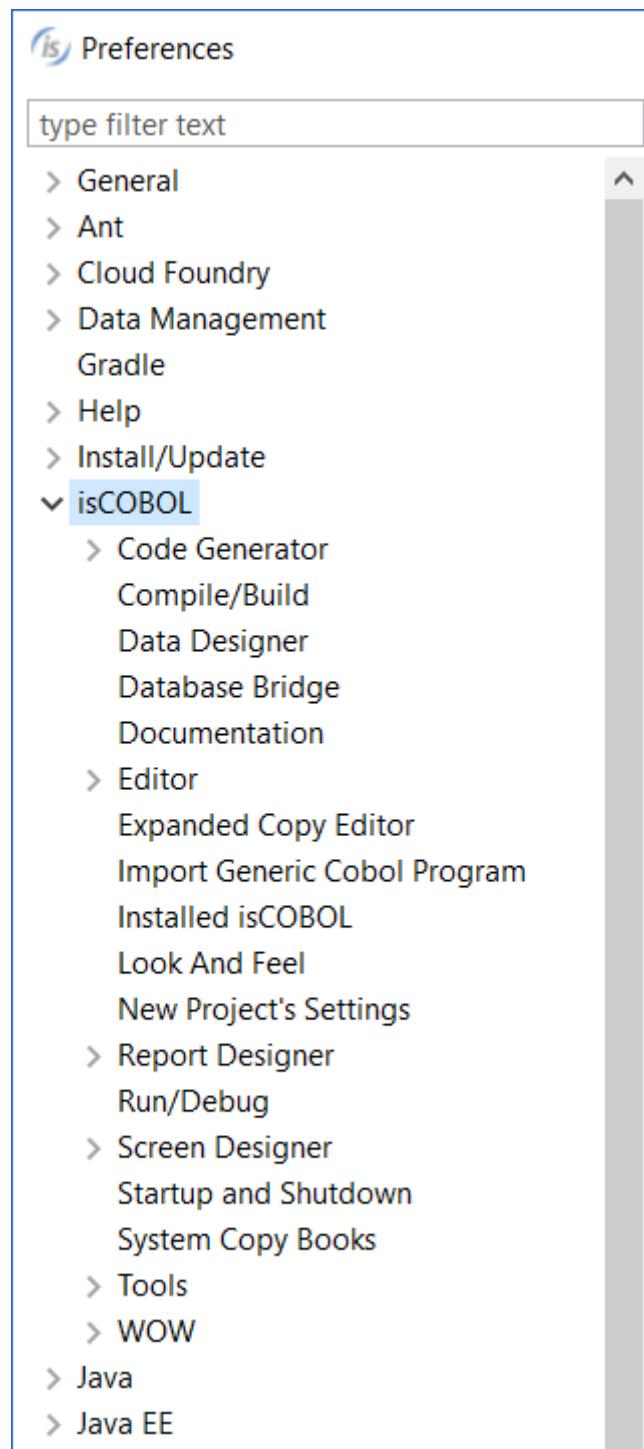
This is also the place to configure iscobel.font properties as the IDE doesn't look for them in iscobel.properties files. If you wish to customize the quick loaded fonts (DEFAULT, MEDIUM, LARGE, SMALL, FIXED and TRADITIONAL) so that the IDE Screen Designer and Report Designer render them differently, you should set the corresponding isCOBOL property here, using the -D java option. For example, by adding the following line

```
-Discobol.font.default=Arial-12
```

the IDE will use the font Arial with size 12 to represent controls whose font is set to 'Default Font' in the Screen Designer and Report Designer.

Customization

Custom settings for isCOBOL IDE can be configured by clicking on the *Window* menu, choosing *Preferences* from the list and selecting *isCOBOL* from the tree.



Verifying Java availability

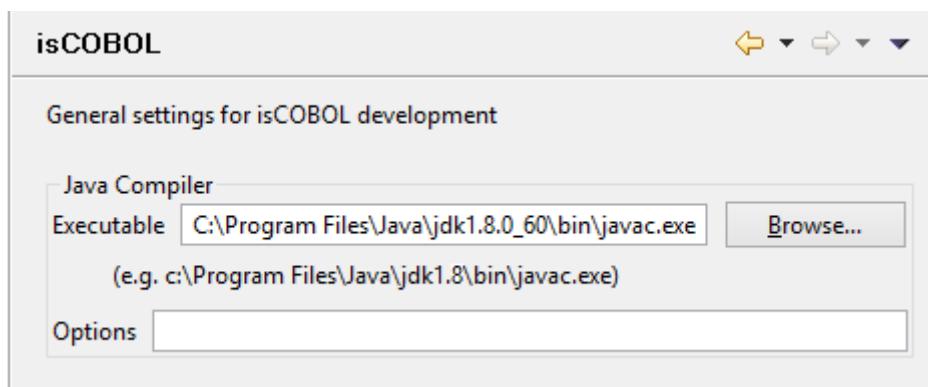
Preferences: isCOBOL

Before starting to use the IDE for the development and maintenance of isCOBOL programs, it is

recommended that you verify the association of an existing JDK to the IDE. This kind of setting is available as soon as *isCOBOL* is selected from the tree.

The **Executable** field should point to the JDK chosen during the installation process. If it's empty or if you want to use a different JDK, provide a new path.

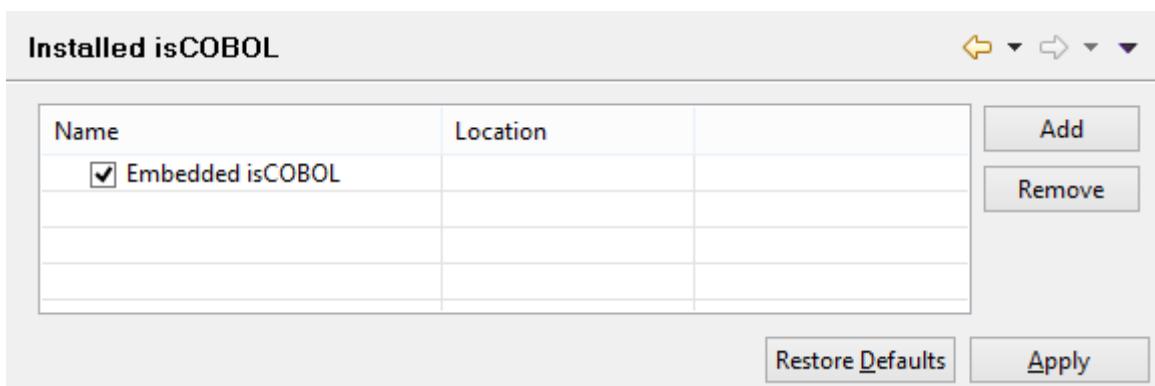
At this point you can also specify javac options that should be used at compile time (for example `-g:none` to discard debug infos), an alternate isCOBOL runtime library and the use of message boxes to inform the user about compiling result.



Binding additional isCOBOL SDK

Preferences: isCOBOL -> Installed isCOBOL

isCOBOL IDE embeds isCOBOL compiler and runtime libraries of the same version and bitness. It's possible to link external isCOBOL SDKs in order to compile and run your programs with a different version of isCOBOL.



Click on the **Add** button and browse for the desired isCOBOL SDK main folder. The selected folder must include "bin" and "lib" subfolders where isCOBOL compiler and runtime libraries are found.

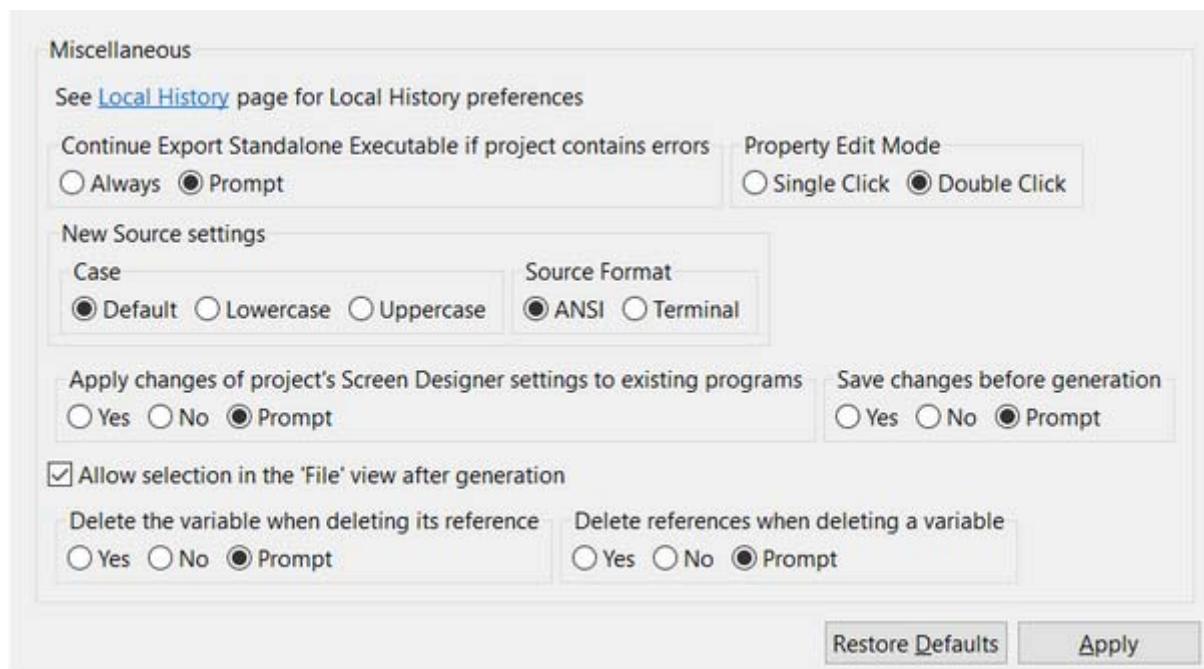
If the selected folder is recognized as a valid isCOBOL SDK, a new line appears in the table and you can now choose which isCOBOL SDK must be used. The choice made here is applied to every new project created in the workspace.

Note - The isCOBOL SDK version must be 2017 R1 or greater. It's not possible to link older isCOBOL SDKs.

Miscellaneous options

Preferences: isCOBOL

Below the Java association, you can configure miscellaneous options.



By default you have to double click on a property value in the Screen Designer in order to edit it. If you prefer to use a single click, this is the place to configure it.

By default the IDE prompts you to save changes made on the screen before generating the source code. From this panel you can also choose if duplicated control IDs are allowed by the Screen Designer; by default, if you set the ID property of a control to a value that has already been used by another control ID, the IDE shows an error.

By default, the IDE automatically selects the generated source code after the code generation is complete. This will allow you to quickly compile your code without having to select it from the isCOBOL Explorer tree. If you wish to disable this feature, uncheck the option "Allow selection in the isCOBOL Explorer after generation".

"Delete the variable when deleting its reference" and "Delete references when deleting a variable" allow you to avoid leaving useless items in the program when something is deleted. For example, if you delete a graphical control from the Screen Designer, you probably don't need the variables that were associated with it, so it's a good idea to delete them. By default the IDE always asks you what to do, but you can configure it to always delete referenced items or never delete referenced items without asking the user.

Code Generation settings

Preferences: isCOBOL -> Code Generator

The *Code Generator* panel allows you to configure the code generated by the IDE for Screen Programs.

In particular you can set:

- which parts of the program should be generated. By default the whole program is generated, but you could ask the IDE to generate just the Screen Section copybook or other single parts.
- the extension of the generated files.
- how many files should be generated. Here you can choose to have multiple copy files or one single source file.
- the output format. The IDE can generate code in "ANSI" and "Terminal" format.
- the text case. The IDE can be forced to generate all the code in lower-case or upper-case. This setting affects all the generated code except for the text between quotes, the method names, the code in the Event Paragraph section, the code outside the Tagged Areas and the comments.
- the DECIMAL-POINT Special Name. Flag the option to generate "DECIMAL-POINT COMMA" in the program Special-Names.
- where to generate column settings (e.g. DISPLAY-COLUMNS and DATA-COLUMNS) for Grid and ListBox. You can choose between having them in the SCREEN SECTION as control properties or in the PROCEDURE DIVISION as MODIFY statements.
- which statement will be generated for the program exit. Choose between "GOBACK" and "STOP RUN". You also choose a variable to be returned as program exit status (Goback variable).
- the value to be moved to the Accept-Control special registry at the end of the Accept.
- if quotes should be generated for alphanumeric properties.
- if the ACCEPT cycle must exit when the EXIT-PUSHED condition name is true or when the KEY-STATUS data item is equal to 27.

Note: if you change the *Source Format* the IDE will ask you to regenerate all the programs in the workspace.

Code generation rules:

- When *Program file* is selected, and *Regenerate tagged area only* is not selected, isCOBOL IDE generates code for the program source file, replacing the existing cbl file.
- When both *Program file* and *Regenerate tagged area only* are selected, isCOBOL IDE generates code in the tagged portions of the program source file only. This preserves any code that you may have added outside of the tags. See [Code Editing](#) for more details.
- When *Working storage* is selected, isCOBOL IDE generates the Working-Storage copy file.
- When both *Working storage* and *Exclude variables in program file* are selected, isCOBOL IDE does not generate Working-Storage items in the .wrk file for variables that are already declared in the program source file.
- When *Event paragraph* is selected, isCOBOL IDE generates the event paragraph copy file.
- When both *Event paragraph* and *Exclude paragraphs in program file* are selected, isCOBOL IDE does not generate paragraphs that are already included in the program source file, it assumes that a paragraph in the program file with the same name as a paragraph that could be generated in the .evt copy file suppresses the paragraph that could be generated.

Code Generator



Generated documents

Merge to one program file

Program file

Regenerate tagged area only

Working storage

Exclude variables in program file

Event paragraph

Exclude paragraphs in program file

Screen section

Report section

Procedure division

Menu paragraph

Linkage section

File procedure

FD

Sort

Select

ANSI Fixed Format Terminal Fixed Format

Program file as read only Copy book as read only

Default case Lower case Upper case

Miscellaneous

Decimal point is comma

Accept Control

Generate column settings code (List-Box, Grid)

in PROCEDURE DIVISION in SCREEN SECTION

Generate Referenced Data Layout(s) files if they don't exist

Generate 'goback' instead of 'stop run' Stop Run/Goback variable:

Generate double quotes for alphanumeric properties

Exit Condition

Default Exit-Pushed Key-Status = 27

Compile/Build settings

Preferences: isCOBOL -> Compile/Build

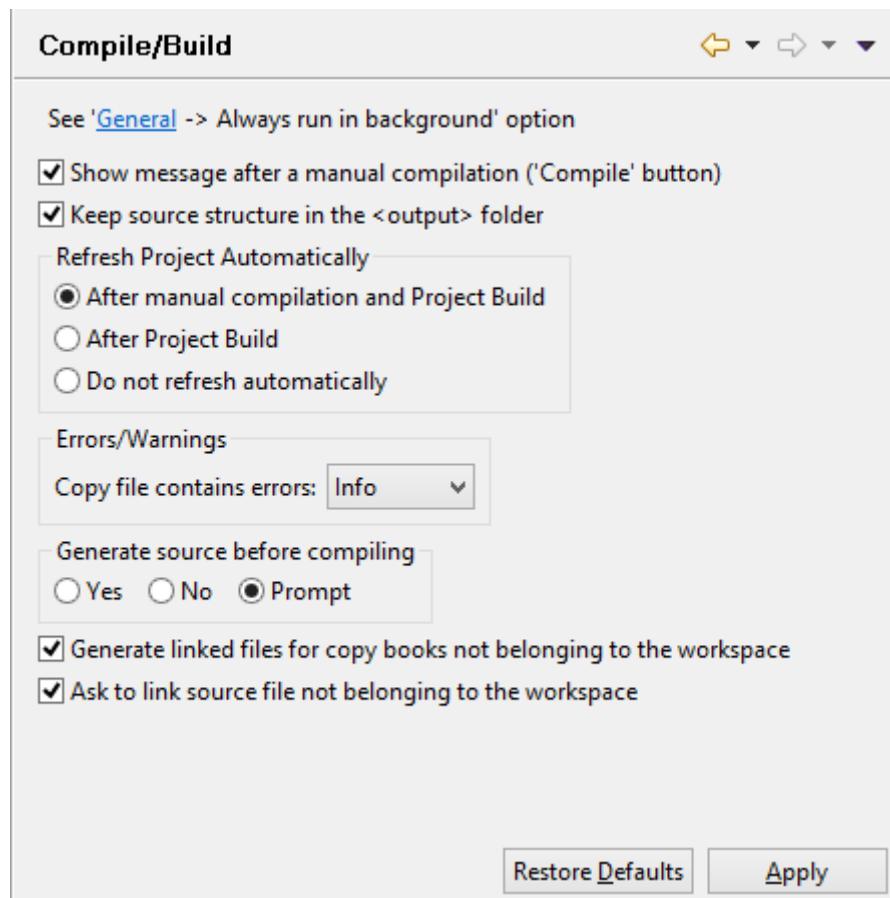
When the IDE builds a project or just compiles a COBOL program it shows a message box that informs the user about the outcome. In this panel you can disable this message. Here it's also possible to instruct the IDE to replicate source sub folders structure (if any) in the output folder.

You can also configure the frequency of project content refresh. By default, the IDE refreshes the project content after each compile and build. You can choose to refresh only after build or to never refresh automatically. Reducing the content refresh may increase performance.

The 'Copy contains errors' option specifies the severity of the corresponding entries in the 'Problems' view. When set to 'Error' or 'Warning' the icons of the related source programs in the 'File' view are marked with the Error/Warning decoration.

By default the IDE generates a linked copybook in the cpy folder for each copybook found via -sp compiler option whose path doesn't belong to the workspace. By unchecking the option 'Generate linked files for copy books not belonging to the workspace', you disable this automatic link.

When you compile the source code in the current editor, if the source doesn't belong to any project, the IDE asks if you wish to add the file to a project. By unchecking the option 'Ask to link source file not belonging to the workspace', when you compile the source code in the current editor, if the source doesn't belong to any project, the IDE shows an error.

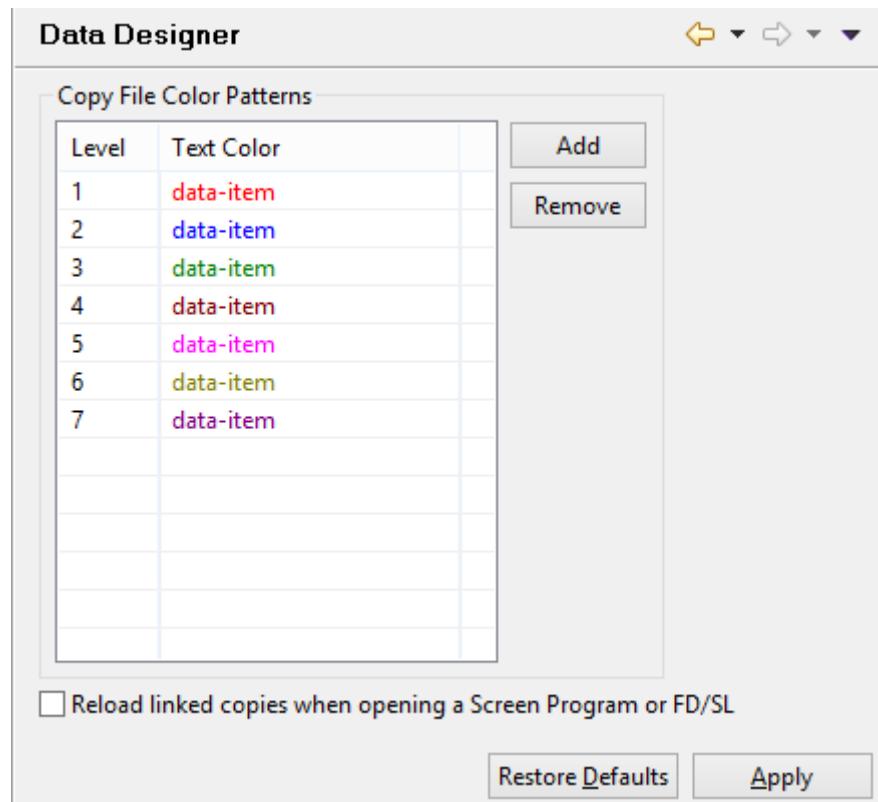


Setting Data Designer options

Preferences: isCOBOL -> Data Designer

When you import or link a copy file in the graphical Linkage Section, Working-Storage or File Designer, the content of the copy file is colored to easily distinguish it from the rest of the data items. Nested copy files are colored with different colors. This panel allows to configure these colors.

After a copy file has been linked, if the user changes it with external editors, the IDE is not aware of the changes. You have to right click on the copy file and choose "Reload" in order to make the IDE aware of the changes. An alternate solution is to check the option "Reload linked copies when opening a Screen Program or FD/SL". In this way the IDE automatically reloads all the linked copy file each time you open the Screen Program or the FD in *Structural* or *Data* view respectively.



Setting Editor preferences

Preferences: General -> Appearance -> Colors and Fonts

The general Colors and Fonts panel allows to configure fonts and colors for the variables highlight and tool-

tips shown in the Code Editor

Colors and Fonts

Colors and Fonts (? = any character, * = any string):

type filter text

- ▷ Basic
- ▷ CVS
- ▷ Debug
- ▲ isCOBOL
 - ▲ Editor
 - Member Highlight
 - Tooltip Background Color
 - Aa* Tooltip Editor Font (set to default)
 - Tooltip Foreground Color
- ▷ Java
- ▷ JavaScript
- ▷ Remote System Explorer
- ▷ Structured Text Editors
- ▷ SVN
- ▷ Tasks

[Edit...](#)

[Use System Font](#)

[Reset](#)

[Edit Default...](#)

[Go to Default](#)

Description:

Color used to highlight a variable name or a paragraph name.

Preview:

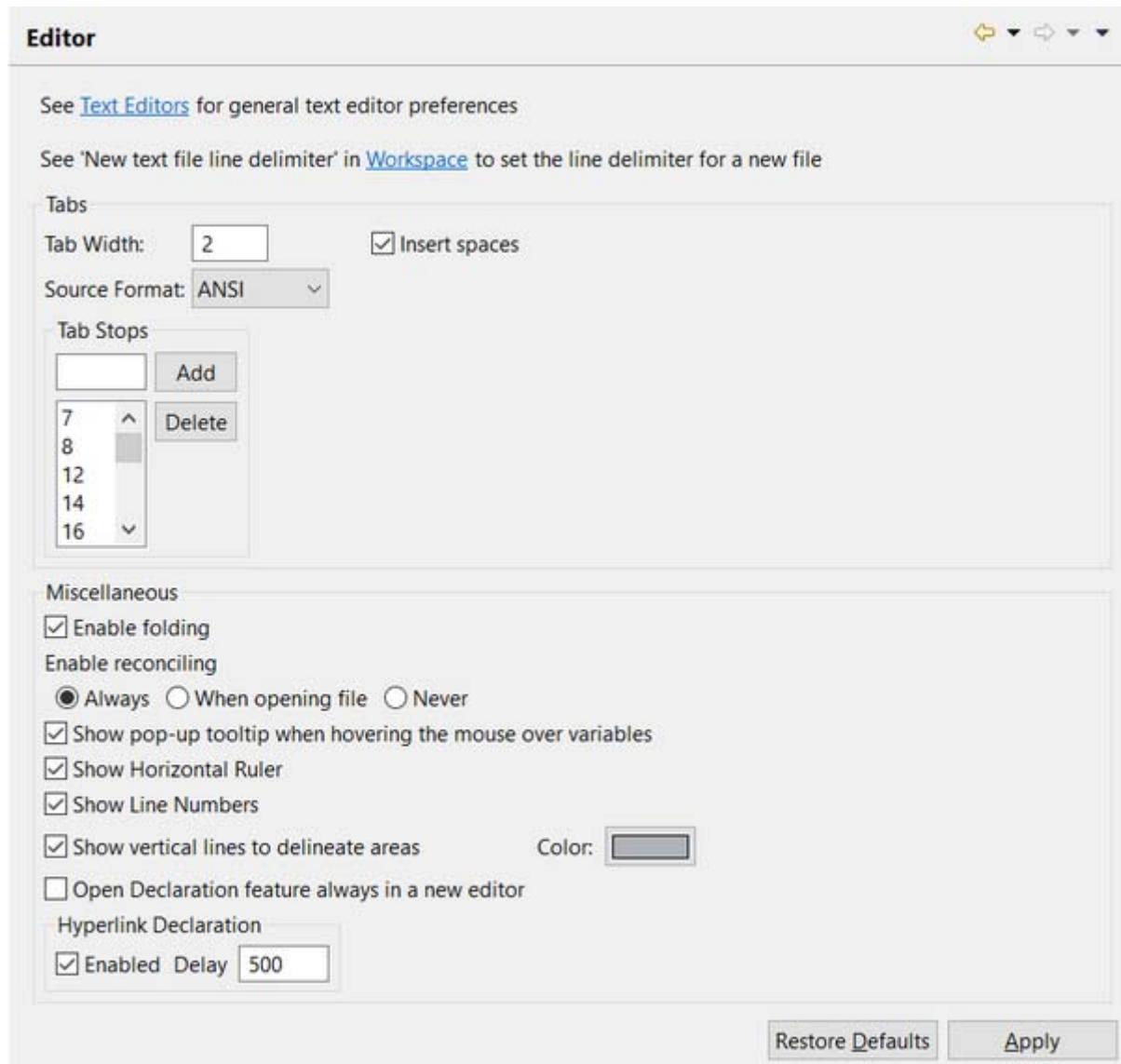
Sample text	Sample text	Sample text
RGB(212, 212, 212)	RGB(212, 212, 212)	RGB(212, 212, 212)

[Restore Defaults](#)

[Apply](#)

Preferences: isCOBOL -> Editor

The Editor panel allows you to configure tab spacing, syntax coloring and templates for the code completion feature. It also allows you to activate the [Horizontal Ruler](#).



Note - folding (ability to expand and collapse blocks of code) and reconciling (real time syntax checking) may cause long response times when editing huge source files (e.g. a source file with over 15.000 lines of code). If you experience bad response times while editing huge source files, you may think about disabling these features.

Syntax Coloring



Source Format

Enable source format color settings

- ▷ ANSI
- ▷ Terminal
- ▷ Free
- ▷ Variable
- ▷ Long Line

Background:

Foreground:

Syntax highlighting

Enable syntax highlighting settings

Comments

Keywords

String constants

Numeric constants

Statements

Level numbers

Figurative constants

Debugger current line

Background:

Foreground:

[Restore Defaults](#)

[Apply](#)

Templates



Create, edit or remove templates:

Name	Context	Description	Auto Ins
<input checked="" type="checkbox"/> A...	Procedure Division Con...	ADD - GIV...	on
<input checked="" type="checkbox"/> A...	Procedure Division Con...	ADD state...	on
<input checked="" type="checkbox"/> C...	Procedure Division Con...	CALL state...	on
<input checked="" type="checkbox"/> C...	Procedure Division Con...	CONTINU...	on
<input checked="" type="checkbox"/> I...	Procedure Division Con...	INQUIRE s...	on
<input checked="" type="checkbox"/> M..	Procedure Division Con...	MODIFY st...	on
<input checked="" type="checkbox"/> M..	Procedure Division Con...	MOVE CO...	on
<input checked="" type="checkbox"/> M..	Procedure Division Con...	MOVE stat...	on
<input checked="" type="checkbox"/> M..	Procedure Division Con...	MULTIPLY ...	on
<input checked="" type="checkbox"/> M..	Procedure Division Con...	MULTIPLY ...	on
<input checked="" type="checkbox"/> n...	Procedure Division Con...	not on exc...	on
<input checked="" type="checkbox"/> o...	Procedure Division Con...	on excepti...	on
<input checked="" type="checkbox"/> P...	Procedure Division Con...	PERFORM ...	on
<input type="checkbox"/> S...	Procedure Division Con...	SUBTRACT...	on

- [New...](#)
- [Edit...](#)
- [Remove](#)
- [Restore Removed](#)
- [Revert to Default](#)
- [Import...](#)
- [Export...](#)

Preview:

[Restore Defaults](#) [Apply](#)

Preferences: isCOBOL -> Expanded Copy Editor

In this panel you can configure the background color used to distinguish copy files from the rest of the source code in the [Copy View Editor](#).

Expanded Copy Editor

isCOBOL Expanded Copy editor preferences:
See [isCOBOL Editor](#) for general isCOBOL editor preferences

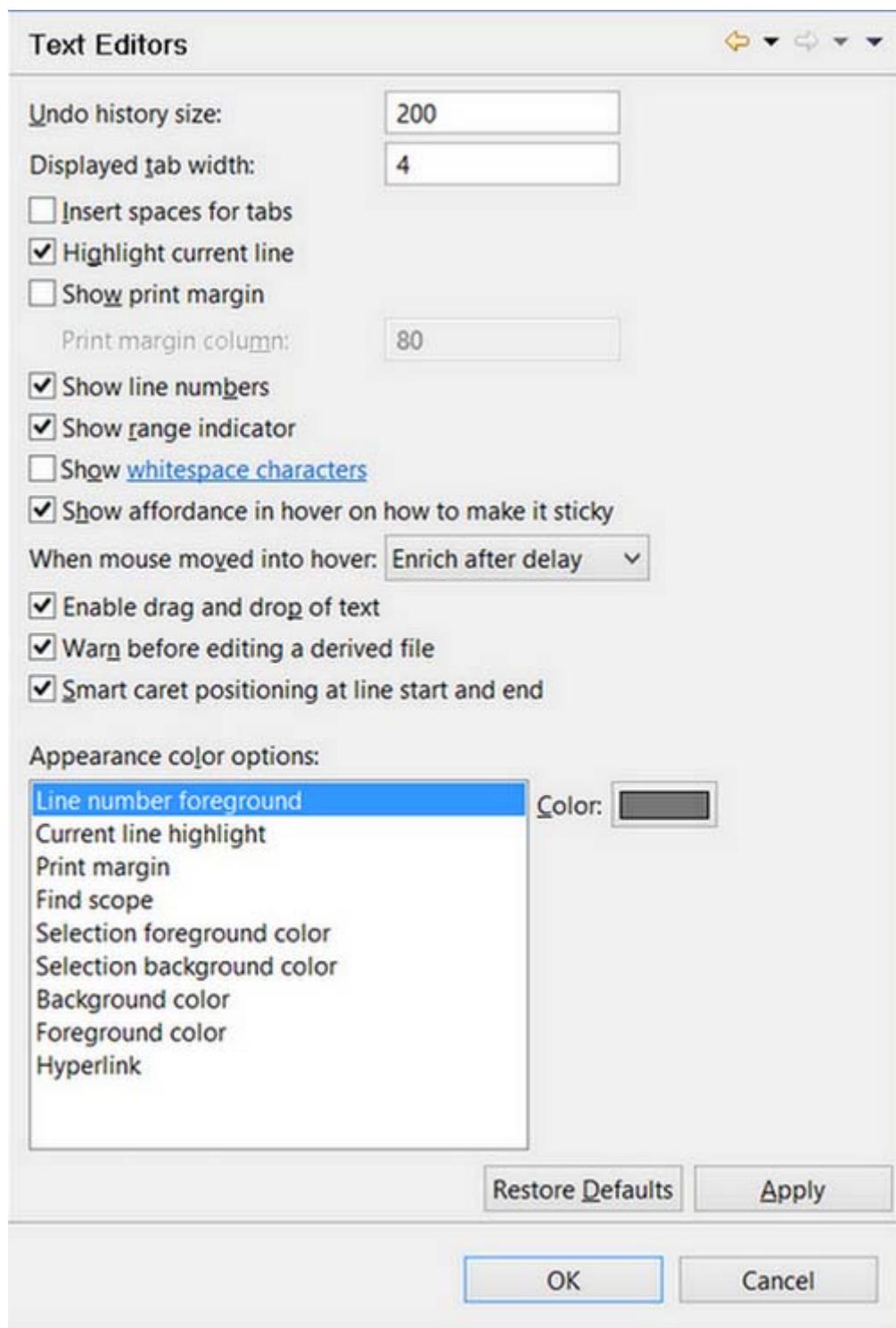
Copy Source Background Color Patterns

Level	Background Color
1	Text
2	Text
3	Text

Add Remove

Restore Defaults Apply

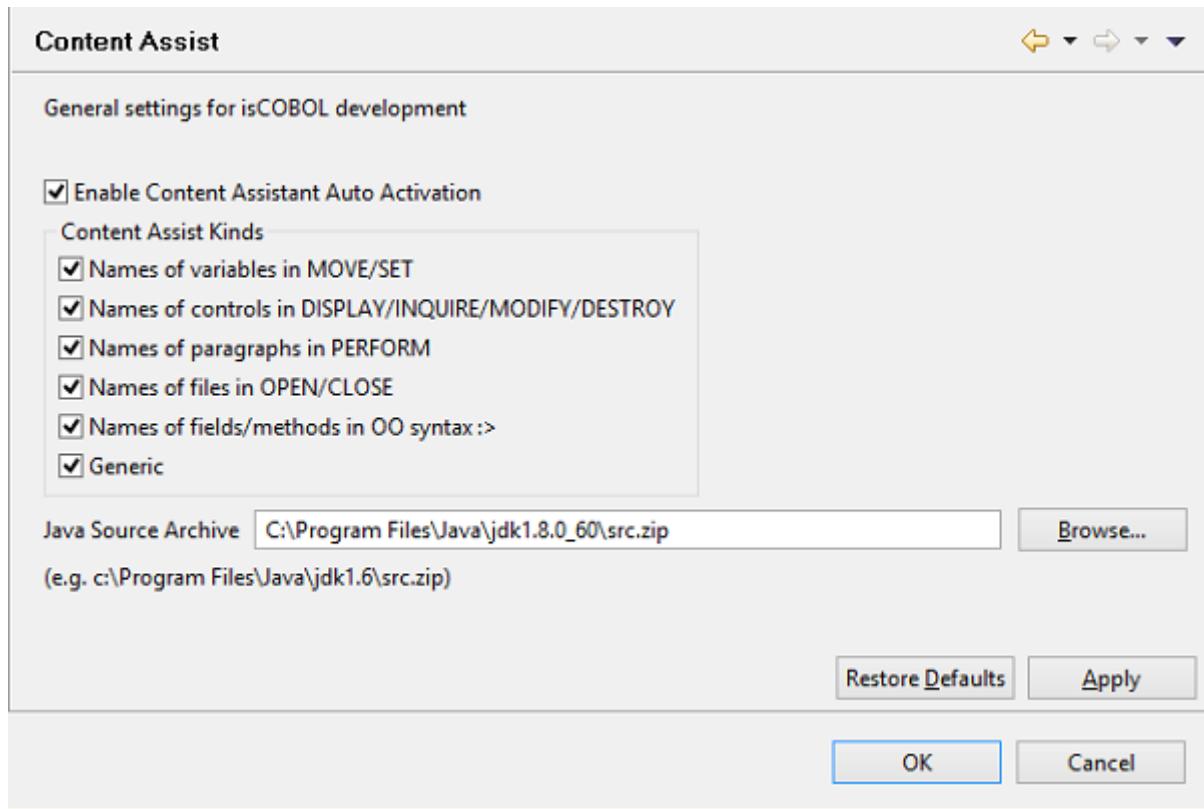
Preferences: General -> Editors -> Text Editors



In this panel you can configure general editor settings that will be applied to every text editor installed in Eclipse, including the isCOBOL Editor.

Configuring the Content Assistant

Preferences: isCOBOL -> Editor -> Content Assist



If *Enable Content Assistant Auto Activation* is checked (default) the Content Assistant is automatically shown while you're editing the code. If the option is unchecked, instead, you have to call the Content Assistant by pressing CTRL+SPACEBAR.

In this panel it's possible to choose also the areas in which the Content Assistant should be activated.

Associating file types with the proper editor

Preferences: General -> Editors -> File Associations

isCOBOL IDE includes different editors in addition to the COBOL editor. Different file types are associated with different editors. You can check and update these associations from the *File Associations* panel.

File Associations



See '[Content Types](#)' for content-type based file associations.

File types:

*.[a](#)

*.[bin](#)

*.[cab](#)

*.[cbl](#)

*.[cl2](#)

*.[class](#)

*.[cob](#)

*.[com](#)

*.[cpy](#)

*.[dat](#)

*.[data](#)

*.[dll](#)

*.[def](#)

*.[dll](#)

*.[drv](#)

[Add...](#)

[Remove](#)

Associated editors:

IscobolEditor (default)

[Add...](#)

[Remove](#)

[Default](#)

The editor is invoked each time you open a file from the isCOBOL Explorer.

There can be one or more editors associated with each file type. Click on the *Add* and *Remove* buttons to create new associations or to remove an existing one.

Files that don't have an associated editor are open through external programs provided by the operating system, outside of the IDE window.

Setting keyboard shortcuts

Preferences: General -> Keys

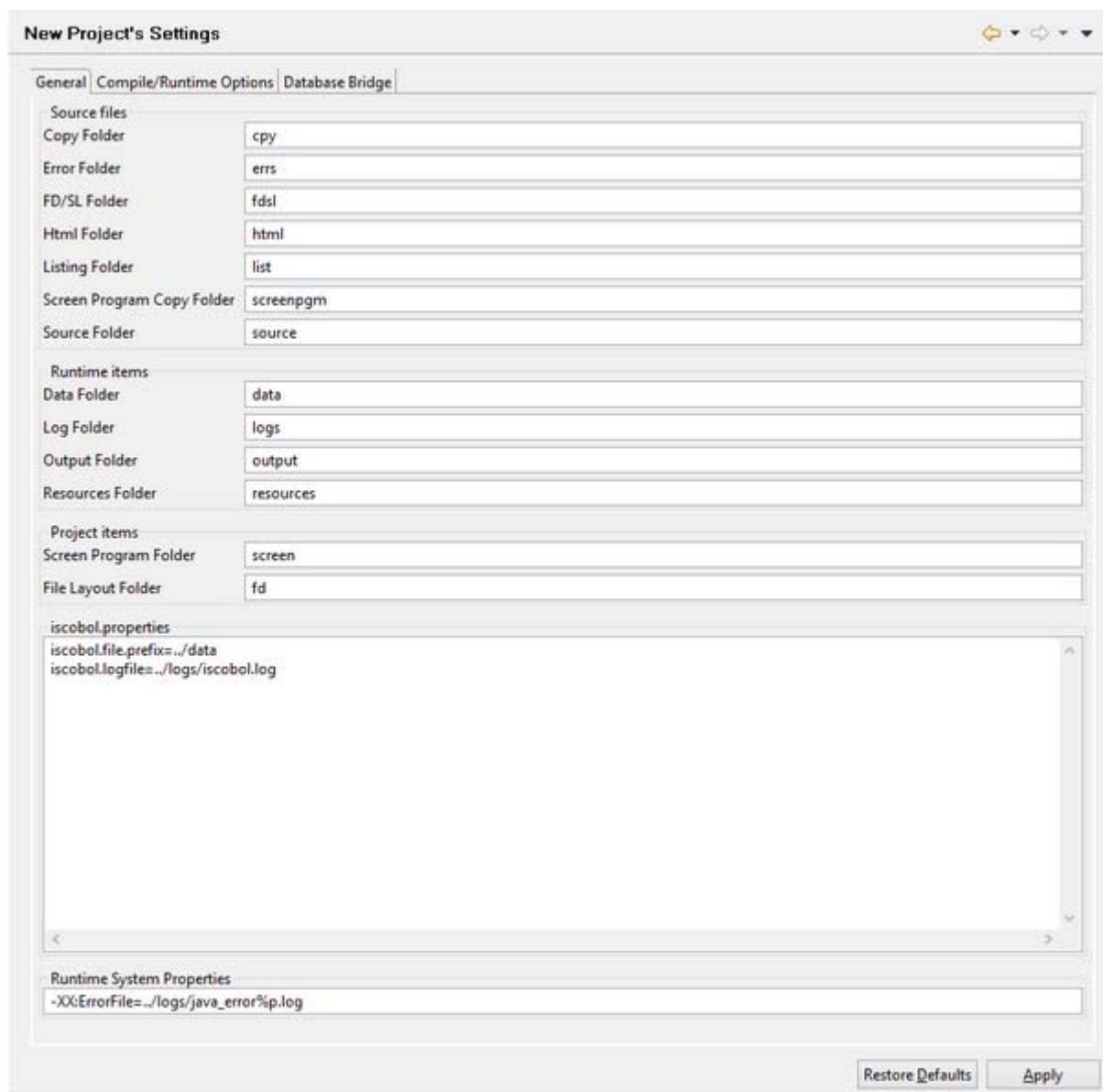
The “Keys” panel allows you to see and change keyboard shortcuts. You can choose between three different presets and also customize them if you need to.

Command	Binding	When	Cat
&Edit Namespace Prefixes			XPa
&Rename XSD element			Edit
About			Hel
Activate Editor	F12	In Windows	Wir
Activate Selected Task			Nav
Activate Task	Ctrl+F9	In Windows	Nav
Add Attribute to XML			JPA

Setting the Project initial settings

Preferences: isCOBOL -> New Project's settings

The “New Project’s settings” panel allows you to customize the name of the Project folders, a list of configuration properties that should always be used for Project programs, and a list of Java options that must be used at program launch.



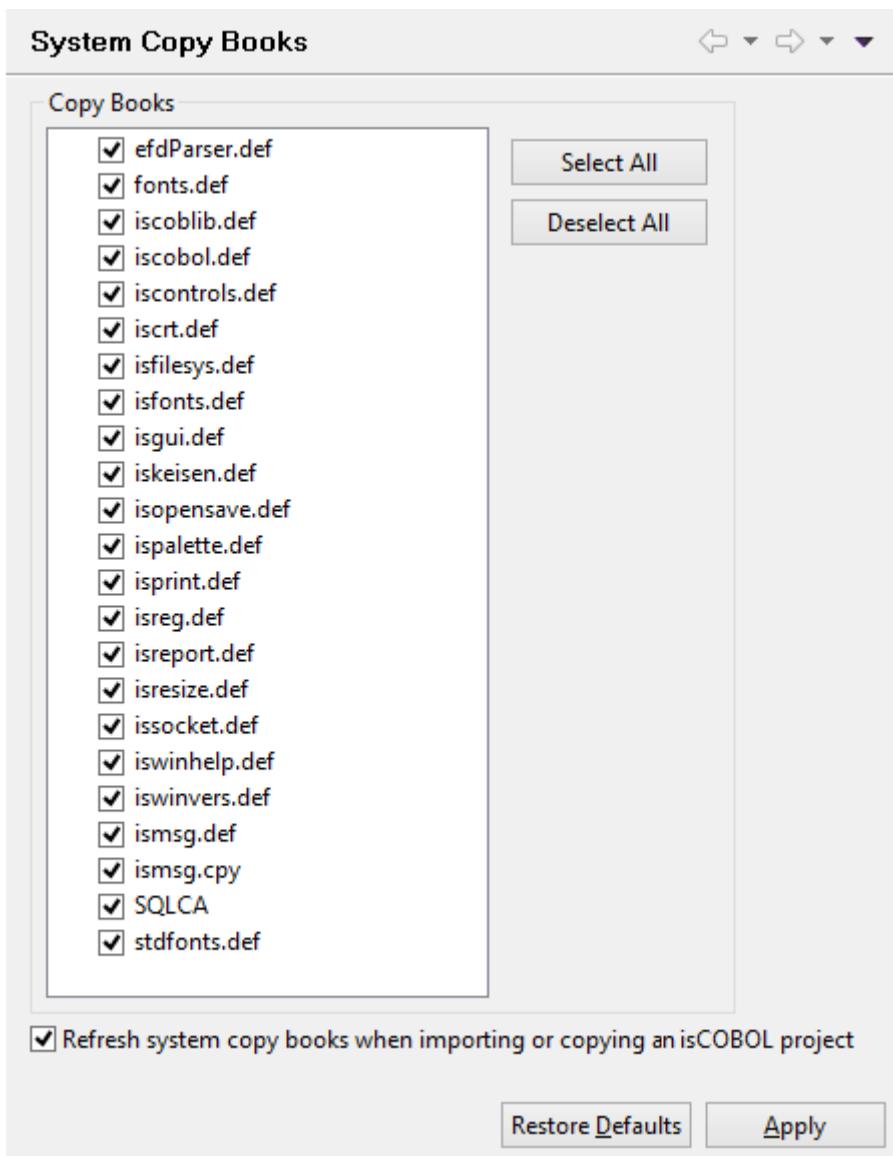
In addition, compile and runtime options can be set. See [Compiling](#) for a complete description of these options.

If you change the name of any of the directories in this panel, the corresponding compiler option is updated.

In this panel you can also configure Database Bridge subroutines generation. See [Generating Database Bridge subroutines](#) for details about this feature.

These settings will be applied to the next project you create; you will be then allowed to edit them in each single project by clicking on *Project* menu and selecting *Properties*.

Preferences: isCOBOL -> System Copy Books



This panel allows to choose which system copy books must be included by default in the project cpy folder.

You can keep your copybooks updated by checking the "Refresh system copy books when importing or copying an isCOBOL project" box. Having this checked means that when you import or copy an isCOBOL project, existing copybooks are replaced with current versions. Unchecking this box will cause only the missing copybooks to be loaded.

Binding Database Bridge to the IDE

Preferences: isCOBOL -> Database Bridge

The Database Bridge panel allows to tell where the EDBIIS executable and its templates are located. License information is shown below the two fields.

By default the IDE uses an internal copy of EDBIIS. Change these fields if you want it to use a different EDBIIS.

Setting Screen Designer preferences

Preferences: isCOBOL -> Screen Designer

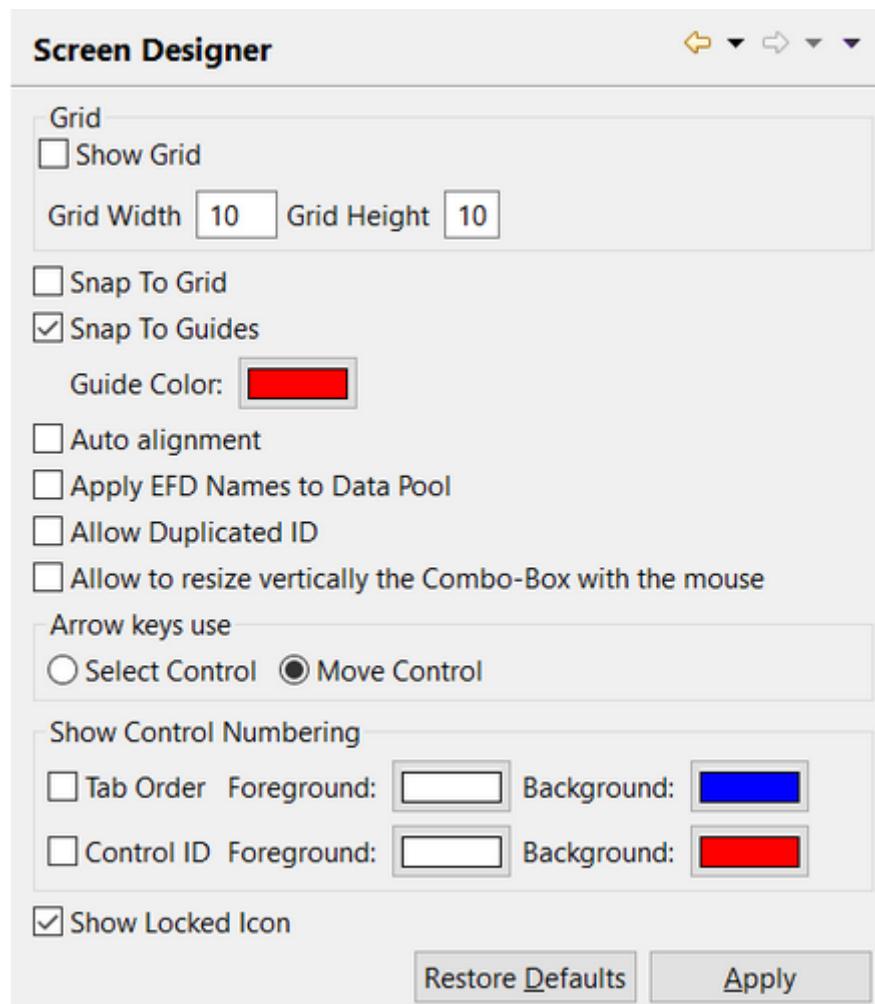
The Screen Designer panel allows you to enable and configure the grid on the background of each screen you draw. When you create a screen using the IDE, a grid made of dotted lines can be shown on the background to help you in placing and aligning the graphical controls. From this panel you can configure the size of the grid cells in pixels and you can activate the 'Snap To Grid' feature to make the IDE automatically align controls to cell boundaries.

Another feature that allows you to easily align controls is the 'Snap To Guides'. With this feature enabled, when you drag a control over the screen, guide lines will be shown on the X and Y axes allowing you to check if the current control position is on the same line or column of other controls.

The 'Allow to resize vertically the Combo-Box with the mouse' enables the ability to alter a Combo-Box height by clicking on its bottom border and dragging the mouse down or up. By default, only the Combo-Box width can be changed by dragging the mouse. This option is evaluated each time a new Screen Designer is opened, so changing it doesn't affect those Screen Designers that are already open.

The 'Show Locked Icon' option causes a lock icon to be shown on the bottom left of locked controls.

In this panel you can also configure the behavior of arrow keys on selected controls. By default, if you select a control on the screen designer and then press arrow keys, the control is moved. You can change this behavior in order to use the arrow keys to switch the selection to another control.



Setting Screen defaults

Preferences: isCOBOL -> Screen Designer -> Default

The Screen Designer / Default panel allows the user to configure default values for properties and styles that will be applied to graphical controls as soon as they're drawn in the Screen Designer. For example, if you want all text fields to be unboxed, you can set the NO-BOX style as default for the ENTRY-FIELD border instead of changing this setting in the Properties view each time you draw a new ENTRY-FIELD.

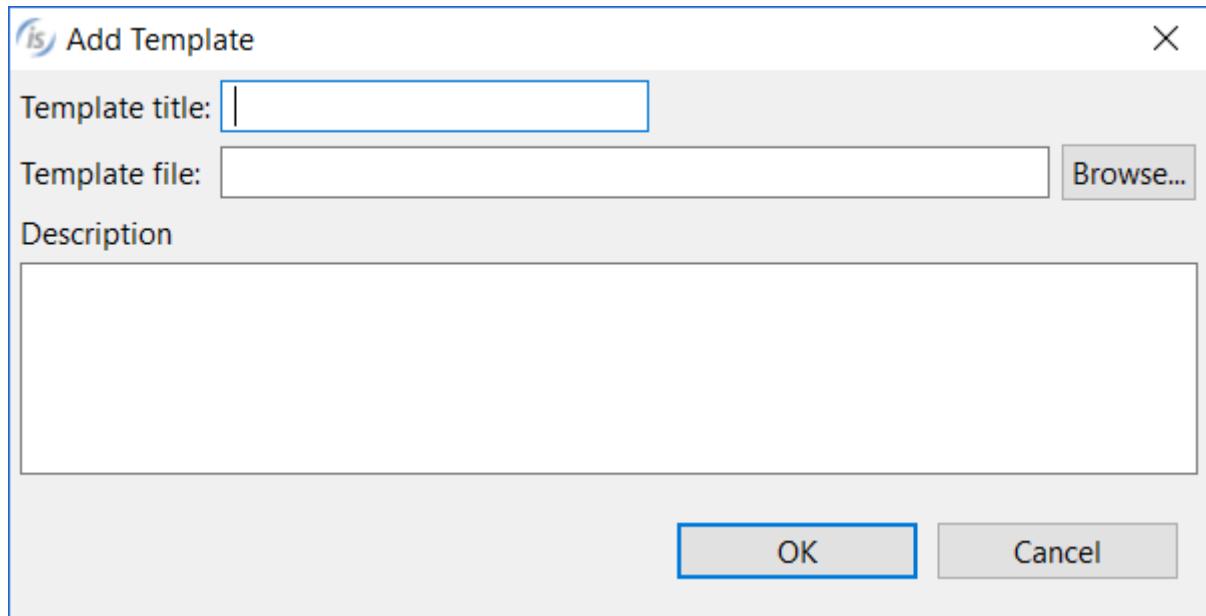
Default

Property	Value
action	None
additional properties	
alignment	NONE
auto	false
autodecimal	0
autodecimal variable	
background-color	
background-color variable	
border	NO-BOX
case	3-D
event list	BOXED
exclude event list	NO-BOX
color	None
color variable	
column variable	
css-style-name	
css-style-name variable	
cursor	0
cursor variable	
cursor-col	0
cursor-col variable	

Loading Screen Templates

Preferences: isCOBOL -> Screen Designer -> Templates

The Templates panel allow you to load previously exported Screen Sections (*.isl) in order to use them as template for the new screen that you will create in the project. Click on the "Add" button to make the following dialog appear:



Provide a name and an optional description for your template, then browse for the isl file.

See [Import / Export of Screens](#) for information about how to produce a isl file.

Setting property Visibility

Preferences: isCOBOL -> Screen Designer -> Visibility

The "Visibility" panel allows you to choose which entries will appear in the Property list of each single control. This feature allows you to filter entries excluding the ones that you will never manage and making it easier to reach the ones you're interested in.

Visibility

The screenshot shows the 'Visibility' panel of the isCOBOL IDE. On the left, a vertical list of UI elements is shown, with 'Screen' currently selected and highlighted in blue. To the right is a table-like grid where each row represents a property name and its corresponding checked status. The first column contains the property names, and the second column contains checkboxes indicating their current state. At the bottom of the panel are two buttons: 'Restore Defaults' and 'Apply'.

Property Name	Value
Properties	<input checked="" type="checkbox"/>
action	<input checked="" type="checkbox"/>
additional properties	<input checked="" type="checkbox"/>
allowing messages	<input checked="" type="checkbox"/>
auto fit	<input checked="" type="checkbox"/>
auto minimize	<input checked="" type="checkbox"/>
auto-resize	<input checked="" type="checkbox"/>
background image	<input checked="" type="checkbox"/>
background-color	<input checked="" type="checkbox"/>
before time	<input checked="" type="checkbox"/>
bind to thread	<input checked="" type="checkbox"/>
boxed	<input checked="" type="checkbox"/>
cell	<input checked="" type="checkbox"/>
cell measure	<input checked="" type="checkbox"/>
cell measuring control	<input checked="" type="checkbox"/>
cell measuring font	<input checked="" type="checkbox"/>
cell measuring style	<input checked="" type="checkbox"/>
cell-height	<input checked="" type="checkbox"/>
cell-width	<input checked="" type="checkbox"/>
color	<input checked="" type="checkbox"/>

Buttons at the bottom:

- Restore Defaults
- Apply

Setting the Look And Feel

Preferences: isCOBOL -> Look And Feel

The "Look And Feel" panel allows the user to choose the Look and Feel that will be applied to the windows of the programs and the isCOBOL utilities run from the IDE.

The "Look And Feel" panel lists the LAFs that are available in the current virtual machine. Clicking on the 'Add' button pops up a dialog that allows you to choose a custom jar library with custom LAFs that will be added to the list making it available to the IDE.

Look And Feel

Available Look and Feels:

Name	Class	Location	
<input type="checkbox"/> CDE/Motif	com.sun.java.swing.plaf.motif.MotifLoo...	None	
<input checked="" type="checkbox"/> Metal	javax.swing.plaf.metal.MetalLookAndFe...	None	
<input type="checkbox"/> Nimbus	javax.swing.plaf.nimbus.NimbusLookAn...	None	
<input type="checkbox"/> Windows	com.sun.java.swing.plaf.windows.Wind...	None	
<input type="checkbox"/> Windows Classi	com.sun.java.swing.plaf.windows.Wind...	None	

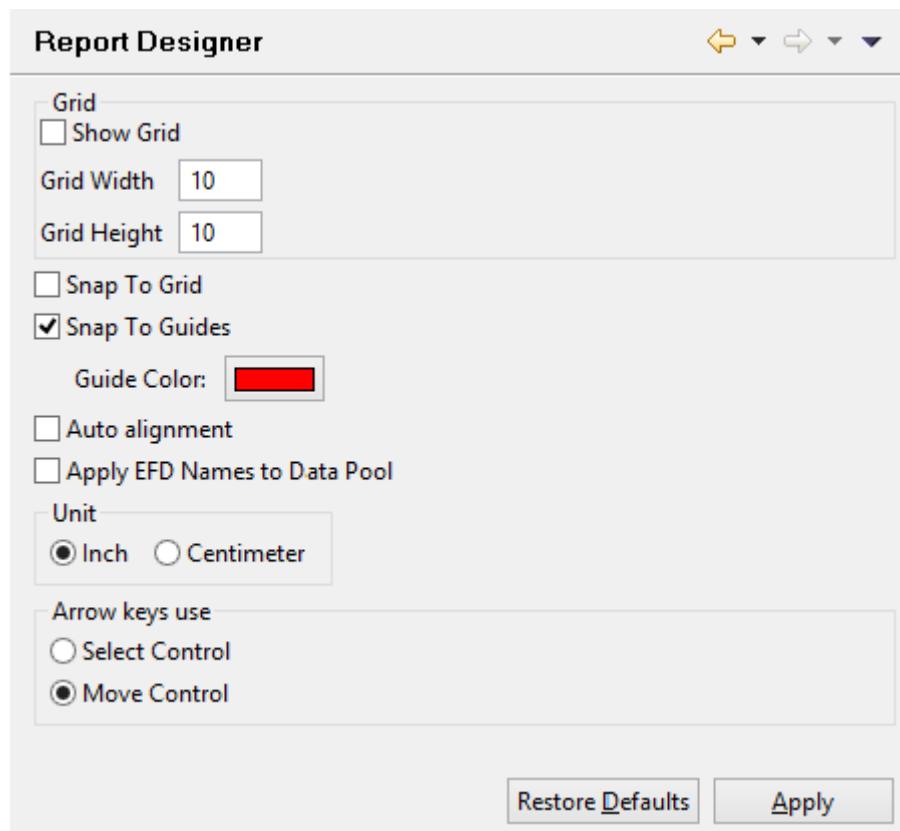
Setting Report Designer preferences

Preferences: isCOBOL -> Report Designer

The Report Designer panel allows you to enable and configure the grid on the background of each report you draw. When you create a report using the IDE, a grid made of dotted lines can be shown on the background to help you in placing and aligning the graphical controls. From this panel you can configure the size of the grid cells in pixels and you can activate the 'Snap To Grid' feature to make the IDE automatically align controls to cell boundaries.

Another feature that allows you to easily align controls is the 'Snap To Guides'. With this feature enabled, when you drag a control over the report, guide lines will be shown on the X and Y axes allowing you to check if the current control position is on the same line or column of other controls.

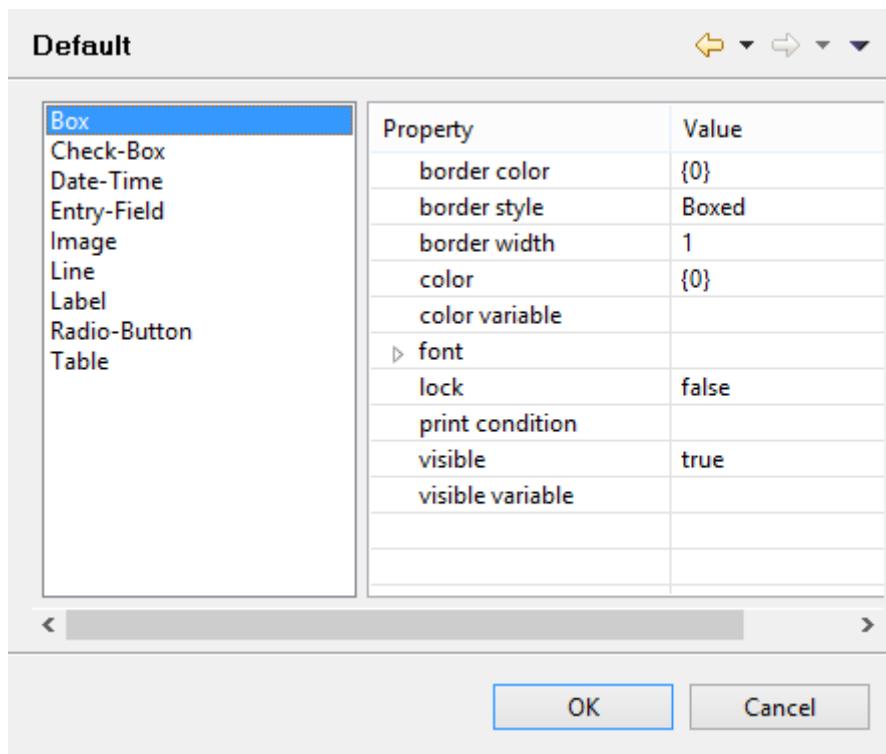
From this panel you can also configure the behavior of arrow keys on selected controls and the measurement unit.



Setting Report defaults

Preferences: isCOBOL -> Report Designer -> Default

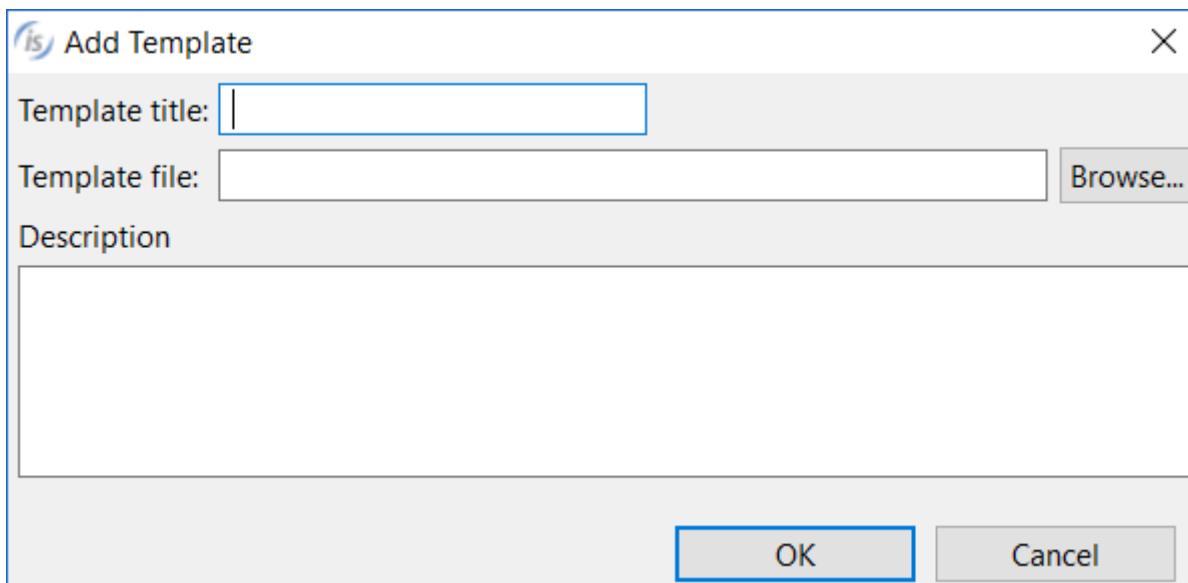
The Report Designer / Default panel allows the user to configure default values for properties and styles that will be applied to graphical controls as soon as they're drawn in the Report Designer.



Loading Report Templates

Preferences: isCOBOL -> Report Designer -> Templates

The Templates panel allow you to load previously exported Reports (*.irl) in order to use them as template for the new reports that you will create in the project. Click on the "Add" button to make the following dialog appear:



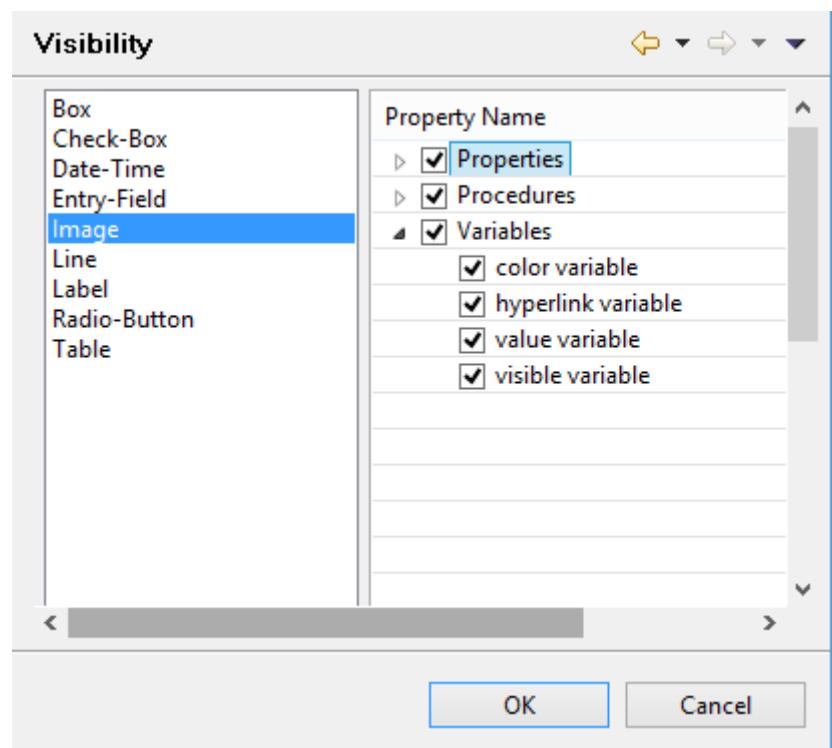
Provide a name and an optional description for your template, then browse for the irl file.

See [Import / Export of Reports](#) for information about how to produce a irl file.

Setting Property visibility

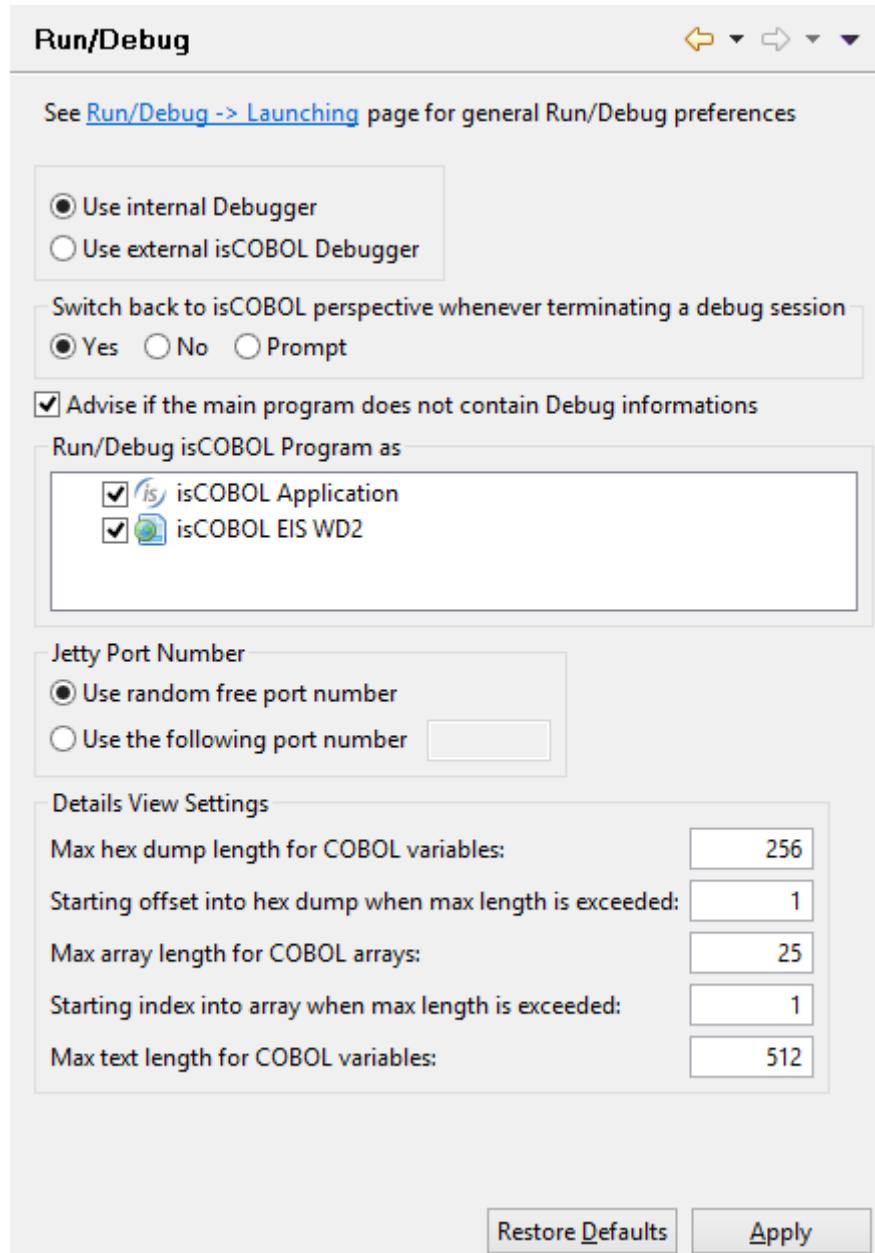
Preferences: isCOBOL -> Report Designer -> Visibility

The “Visibility” panel allows you to choose which entries will appear in the Property list of each single control. This feature allows you to filter entries excluding the ones that you will never manage and making it easier to reach the ones you’re interested in.



Setting Run/Debug options

Preferences: isCOBOL -> Run/Debug



The "Run/Debug" panel allows to configure options for the *Run As* and *Debug As* functions.

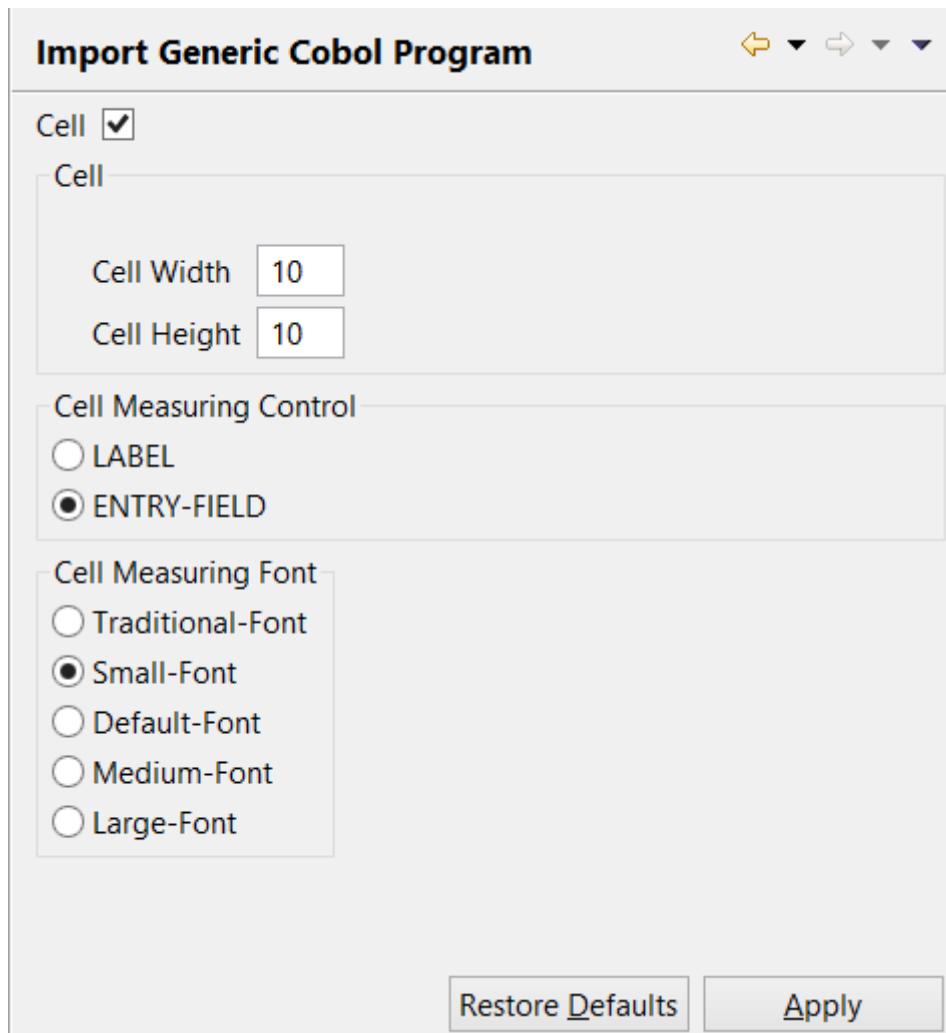
In this panel is also possible to configure the port used by Jetty, an internal servlet container that the IDE uses to run programs as 'isCOBOL EIS WD2' or 'isCOBOL EIS Servlet'. By default, a random port is used at each launch. If you want the IDE to use always the same port, here you can configure it.

Unchecking either *isCOBOL Application* or *isCOBOL EIS WD2* causes the other option to be used directly when the user run or debug a program.

Configuring the Import Generic COBOL Program Feature

Preferences: isCOBOL -> Import Generic Cobol Program

In this dialog it's possible to specify the cell measurement approach to be used when a generic COBOL program with Screen Section is imported in the IDE. See [Importing a Program with Screen Section](#) for further details about importing programs with Screen Section in the IDE.



Configuring isCOBOL Tools

isCOBOL Utilities Settings

Preferences: isCOBOL -> Tools -> Index and Relative File Editor

These panels allow you to configure the Class Path and the Environment for the GIFE utility. See [Setting the Class Path](#) and [Setting the Environment](#) for details about the Class Path and Environment panels. This kind of setting is particularly useful in order to add JDBC driver libraries for the “easydb” file handler.

Preferences: isCOBOL -> Tools -> Index File Migration

These panels allow you to configure the Class Path and the Environment for the ISMIGRATE utility. See [Setting the Class Path](#) and [Setting the Environment](#) for details about the Class Path and Environment panels. This kind of setting is particularly useful in order to add JDBC driver libraries for the “easydb” file handler.

Preferences: isCOBOL -> Tools -> isCOBOL Launcher

These panels allow you to configure the Class Path and the Environment for the ISL utility. See [Setting the Class Path](#) and [Setting the Environment](#) for details about the Class Path and Environment panels. This kind of setting is particularly useful in order to add JDBC driver libraries for the “easydb” file handler.

Preferences: isCOBOL -> Tools -> Java-Bean Copy Generator -> Class Path

This panel allows to configure the Class Path for the cpgen command. See [Setting the Class Path](#) for details about this panel.

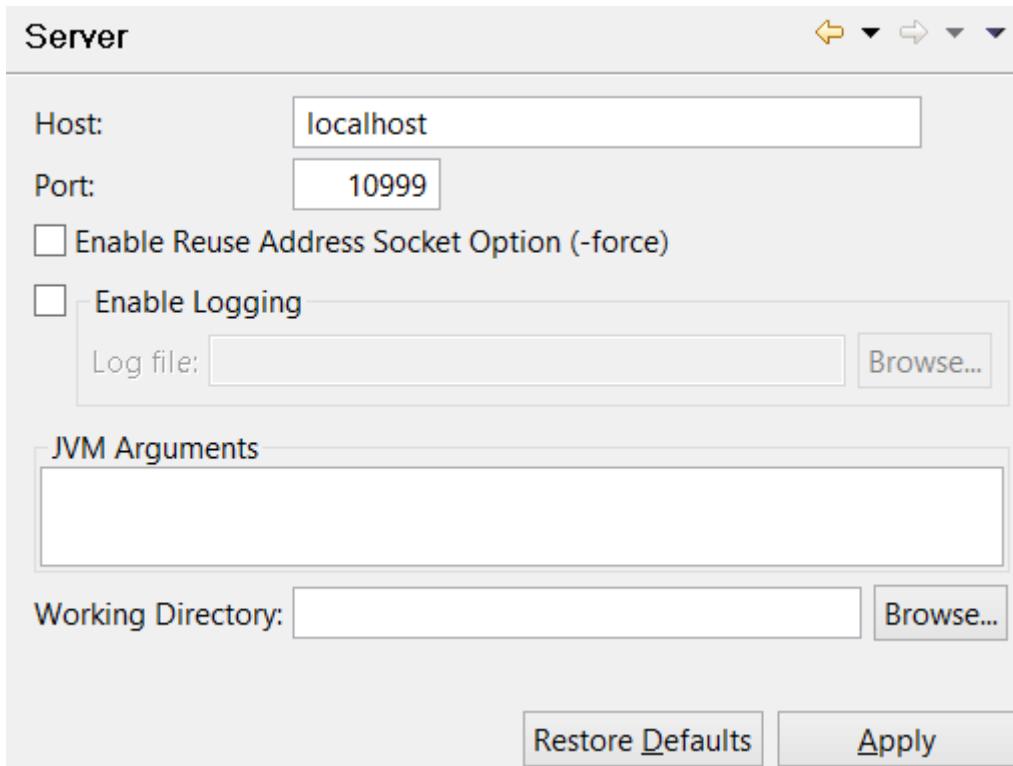
Preferences: isCOBOL -> Tools -> Jdbc2FD -> Class Path

This panel allows to configure the Class Path for the JDBC2FD utility. It’s particularly useful in order to add JDBC driver libraries. See [Setting the Class Path](#) for details about this panel.

isCOBOL Server Settings

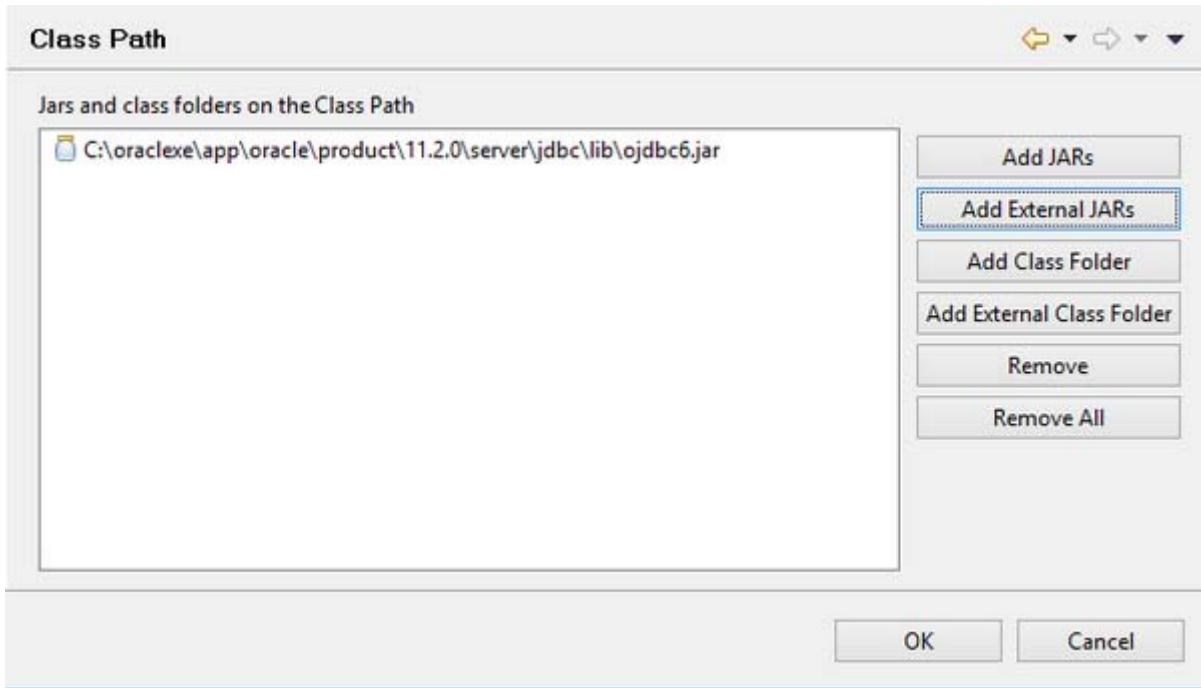
Preferences: isCOBOL -> Tools -> Server

The “isCOBOL Server” panel allows you to set options and parameters for the isCOBOL Server managed by the isCOBOL IDE.



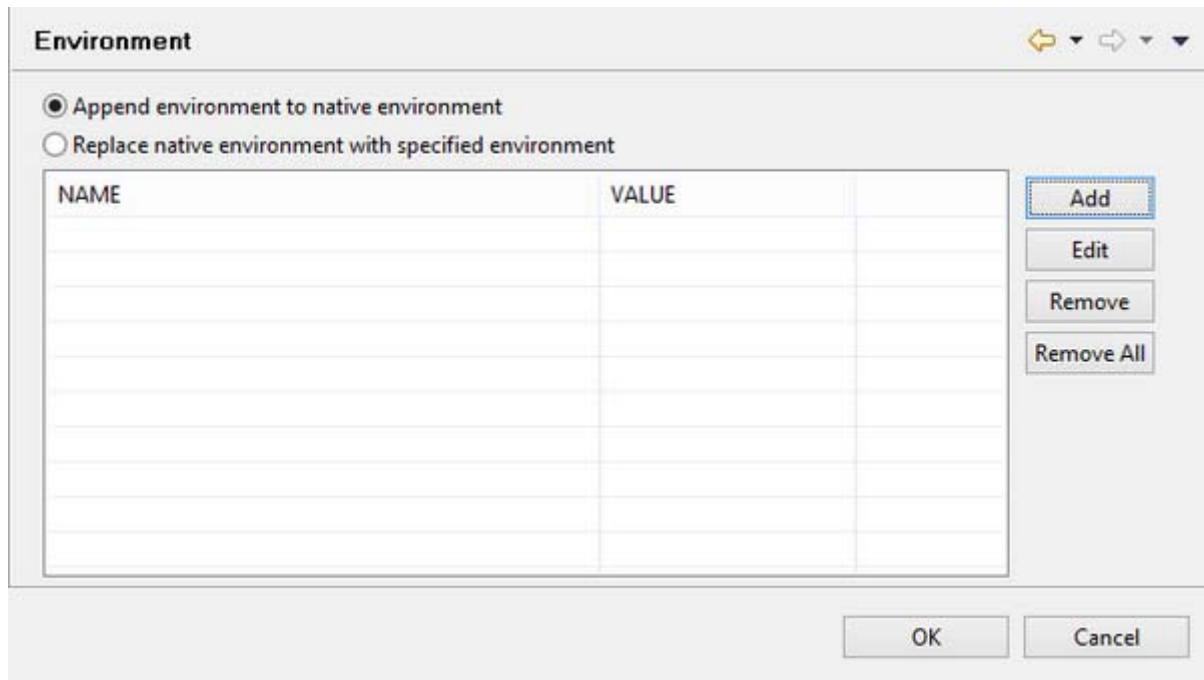
The “Class Path” panel allows to add jars and folders to the isCOBOL Server Class Path. The “Environment” panel allows to set environment variables for the isCOBOL Server. See [Setting the Class Path](#) for details about this panel.

Setting the Class Path



- Use the "Add JARs" button to browse for a jar among project folders and add it to the Classpath.
- Use the "Add External JARs" button to browse for a jar among system folders and add it to the Classpath.
- Use the "Add Class Folder" button to browse for a folder in the project and add it to the Classpath.
- Use the "Add External Class Folder" button to browse for a folder in the system and add it to the Classpath.
- Use the "Remove" button to remove an item from the list.
- Use the "Remove All" button to clean the list.

Setting the Environment



Use the "Add" button to define a new environment variable.

Use the "Edit" button to change the value of the selected variable.

Use the "Remove" button to remove a variable from the list.

Use the "Remove All" button to clean the list.

When the "Append environment to native environment" option is selected, the variables specified by the user plus the variables already defined in the system are included in the process environment. If a variable specified by the user is already defined in the system, its value is overridden.

When the "Replace native environment with specified environment" option is selected, only the variables specified by the user are included in the process environment.

The library path variable (PATH on Windows, LD_LIBRARY_PATH on Linux/Unix) is managed differently, as follows: if it is already defined in the process environment, either by the user or by the system, the isCOBOL native libraries location is appended to its value; if it is not defined, it is set to the isCOBOL native libraries location.

Load Balancer Settings

Preferences: isCOBOL -> Tools -> Load Balancer

The "Load Balancer" panel allows you to set options and parameters for the isCOBOL Load Balancer managed by the isCOBOL IDE.

Load Balancer



Load Balancer Property File

C:\Users\Luciano\isbalancer.properties

[Browse...](#)

Host: localhost

Port: 10999

Update Interval: 60 seconds

Update Timeout: 60 seconds

Enable Reuse Address Socket Option (-force)

Enable Logging

Log file:

[Browse...](#)

JVM Arguments

Servers

[New](#)

[Edit](#)

[Remove](#)

[Remove All](#)

License Key

[Insert new License](#)

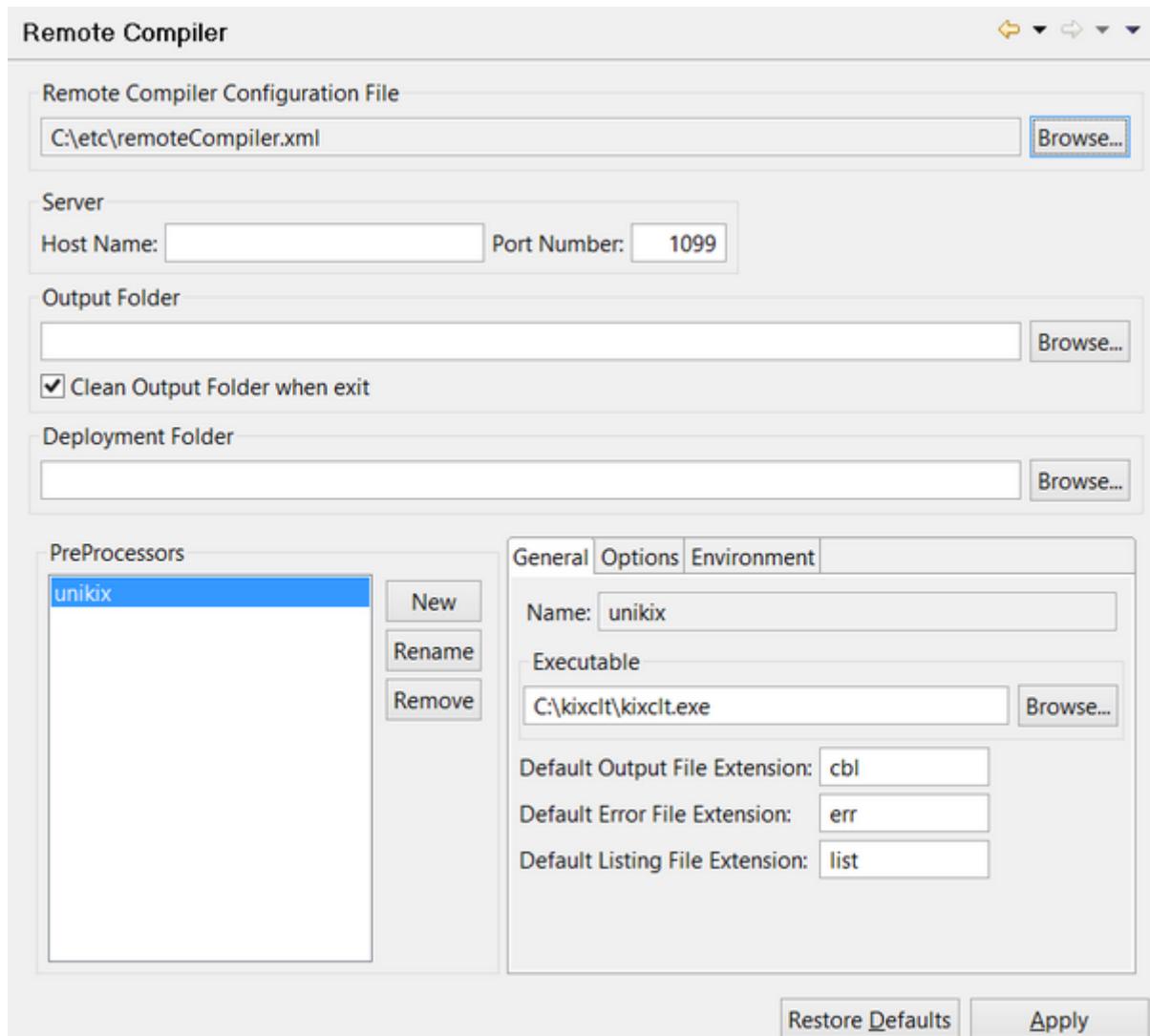
[Restore Defaults](#)

[Apply](#)

Remote Compiler Settings

Preferences: isCOBOL -> Tools -> Remote Compiler

The “Remote Compiler” panel allows you to set options and parameters for the Remote Compiler managed by the isCOBOL IDE.



Configuring Cobol WOW

Preferences: isCOBOL -> WOW

These settings should be reviewed only if you're going to import and maintain Cobol WOW programs from RM/COBOL.

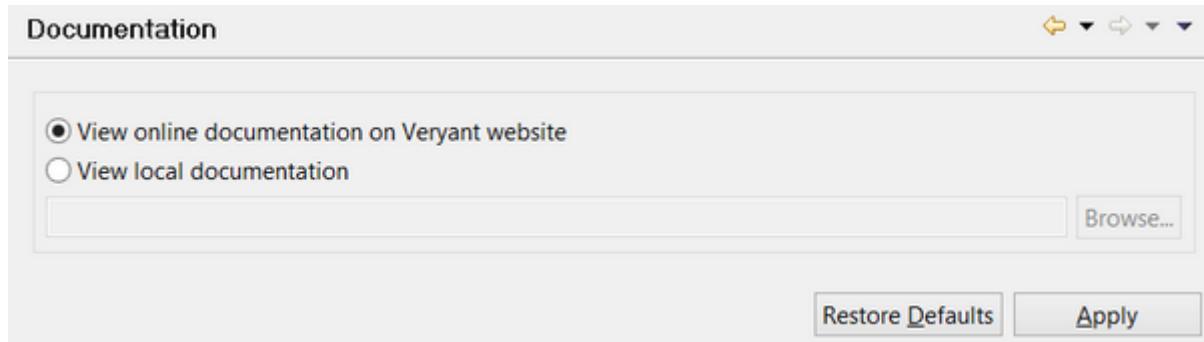
This argument is discussed in the Transitioning from RM/COBOL guide.

See [Before you start – Initial IDE configuration and creation of a new project](#) for more information.

Linking the isCOBOL documentation

Preferences: isCOBOL -> Documentation

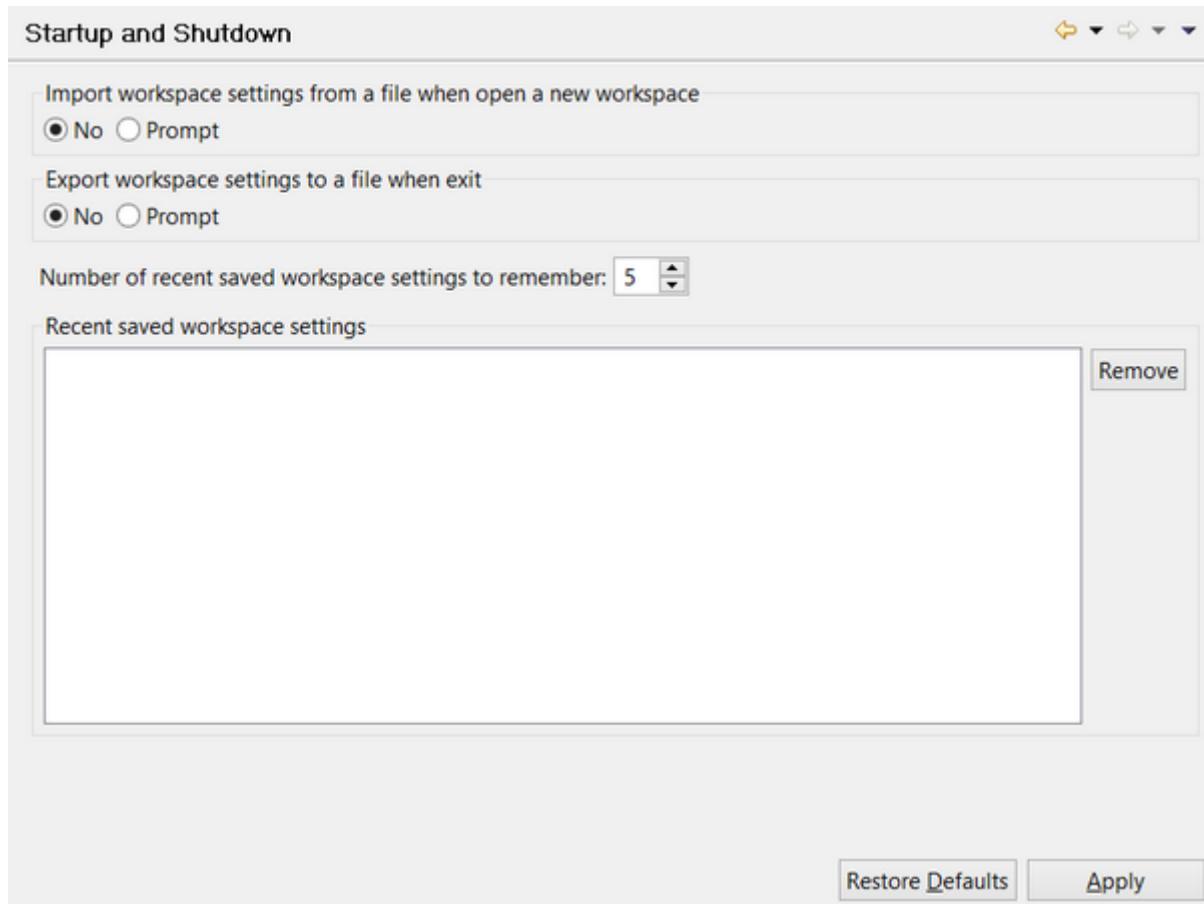
The “Documentation” panel allows you to choose which isCOBOL Documentation will be open from inside the IDE when you select *Open isCOBOL Documentation* in the *Help* menu. By default the IDE opens the online documentation available on Veryant’s website, which is the most up to date, however you can make it link to a local copy of the isCOBOL documentation if you prefer.



Configuring Startup and Shutdown preferences

Preferences: isCOBOL -> Startup and Shutdown

The “Startup and Shutdown” panel allows you to activate and deactivate the option to save workspace settings to a file when the workspace is closed and to load them when a new workspace is open. By default the IDE asks the user for the name of the file in which preferences are stored while opening or closing a workspace. To disable the feature, change the setting from “Prompt” to “No”.



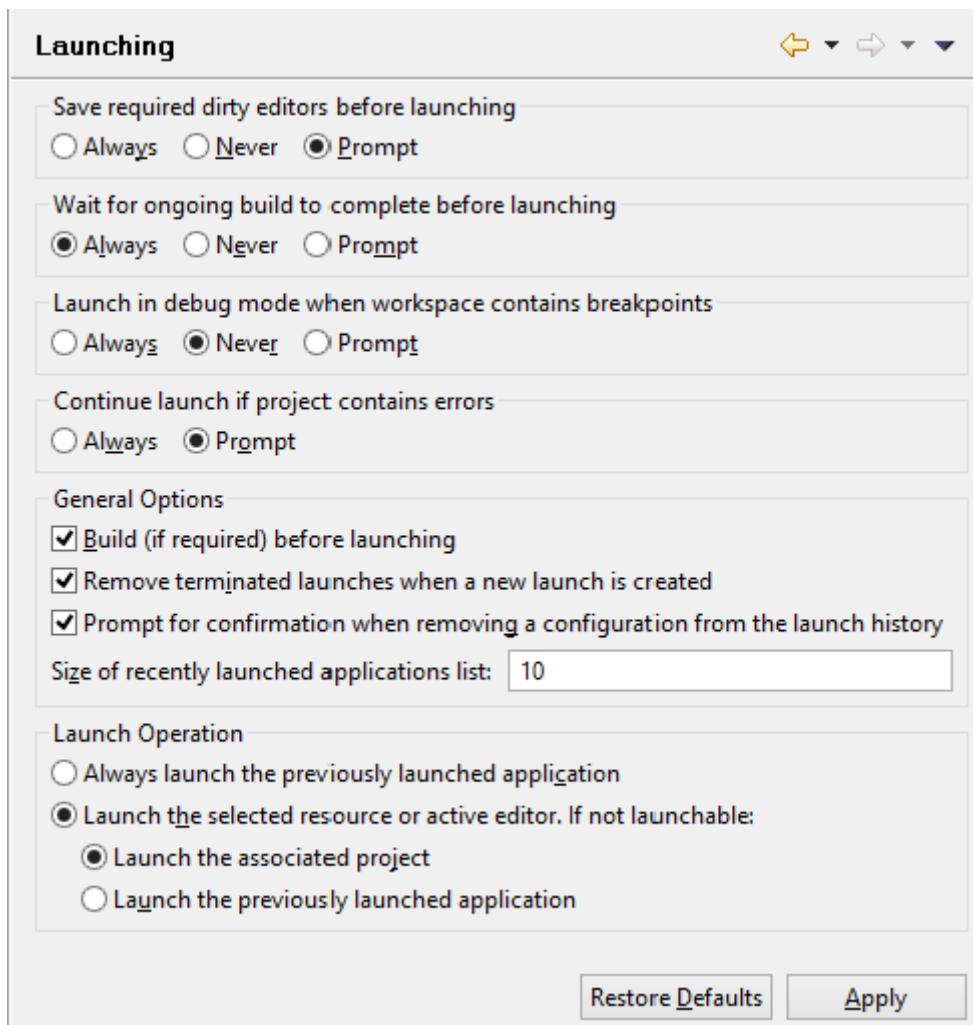
Configuring launching preferences

Preferences: Run/Debug -> Launching

The "Launching" panel allows to configure the actions taken when a program is launched.

When you run or debug a program, if there are some unsaved editors, a message box appears asking if you want to save the editors before launch. Also, if the option *Build (if required) before launching* is checked, a build of the project is executed, so the modified source will be recompiled.

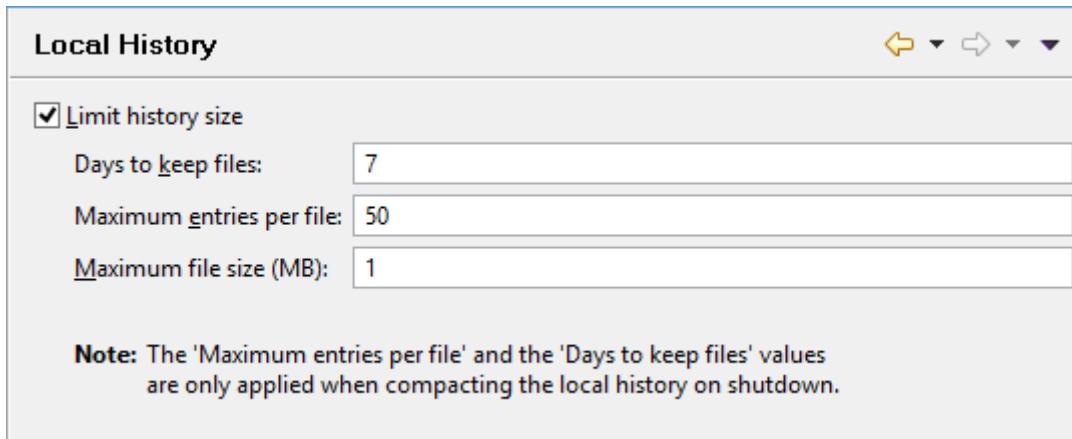
This behavior is identical with or without 'Build Automatically' set for the project, although, when the 'Build Automatically' is active, whenever you save a source, the build starts automatically.



Configuring the Local History

Preferences: General -> Workspace -> Local History

The Local History keeps track of modifications to the project items allowing to restore a previous version of the item. In this panel you can limit the size of the history. If "Limit history size" is disabled, the other settings are ignored and the Local History entries are saved indefinitely.



Limit history size: enables the limits to the History size (default enabled)

Days to keep files: when close and save the workspace, removes the entries older than <n> days (default 7 days).

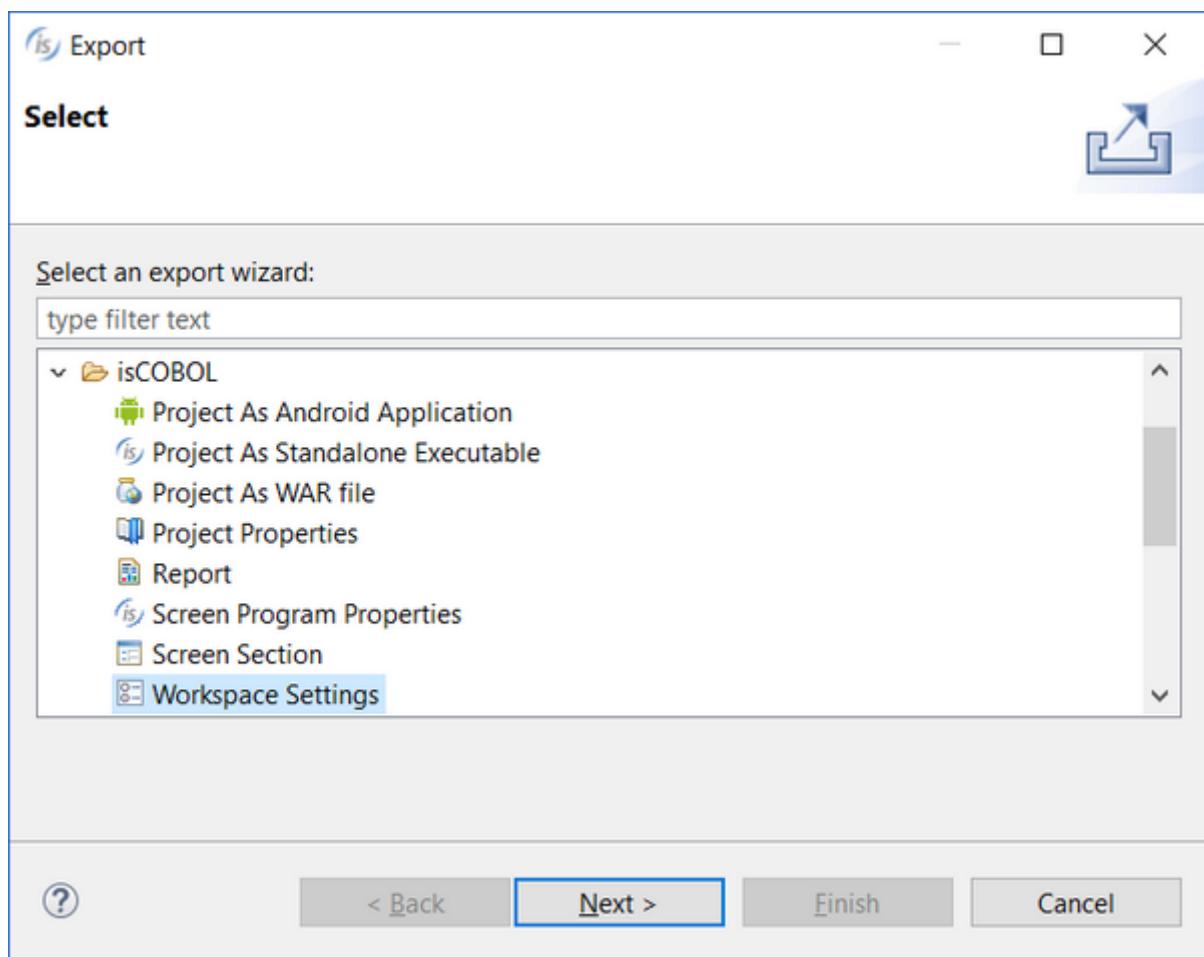
Maximum entries per file: when close and save the workspace, maintains only the most recent <n> entries (default 50 entries).

Maximum file size (MB): set the maximum file size for which the 'Local History' feature is enabled (default 1 MB).

Import / Export of Workspace settings

To export workspace preferences:

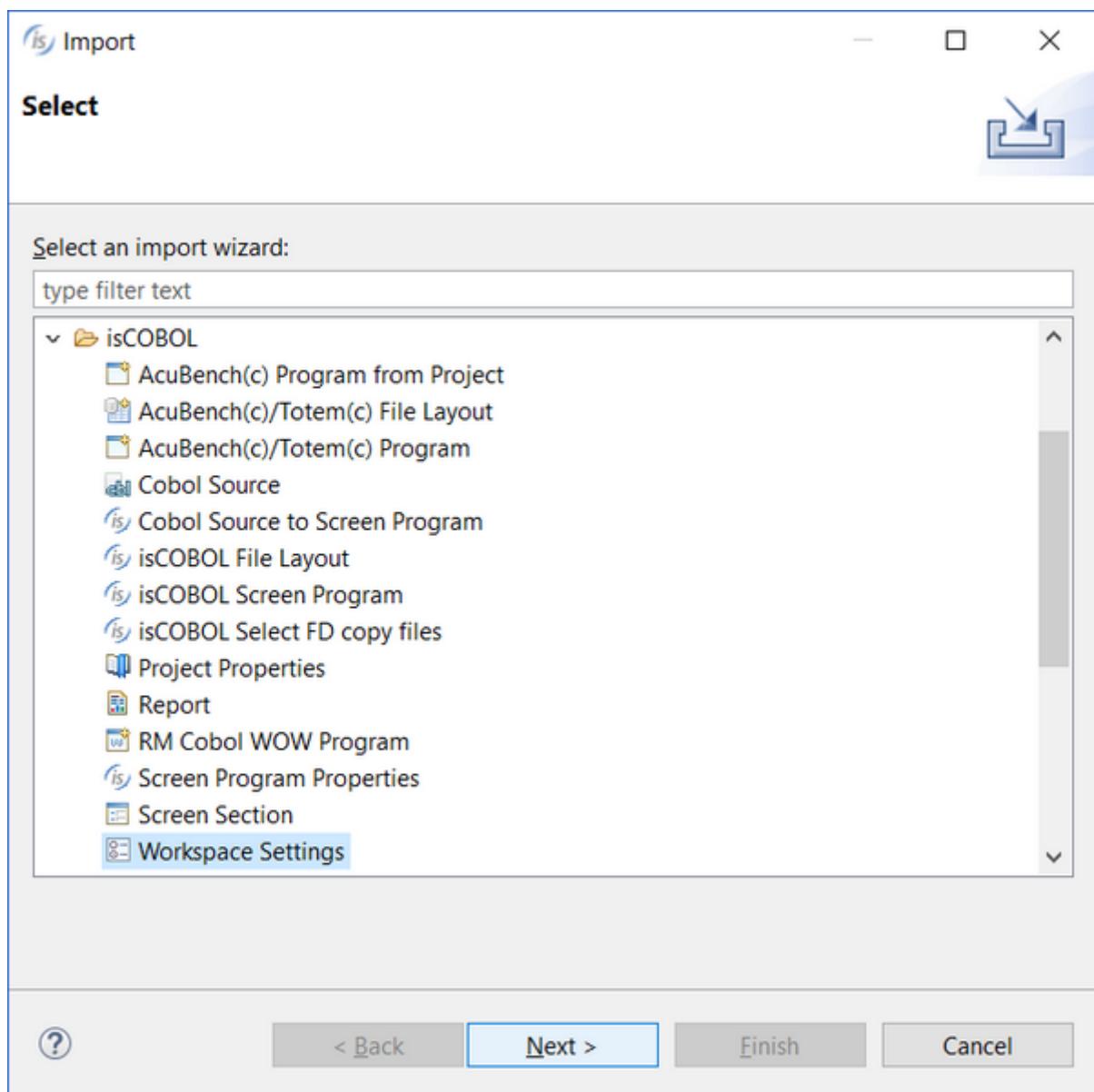
1. Click on the *File* menu
2. Choose *Export*
3. Expand *isCOBOL*
4. Choose *Workspace settings*



5. Click *Next*
6. Choose the destination file (it must have .prefs extension)
7. Click *Finish*

To import workspace preferences:

1. Click on the *File* menu
2. Choose *Import*
3. Expand *isCOBOL*
4. Choose *Workspace settings*



5. Click *Next*
6. Choose a saved settings file from disk (it must have .prefs extension)
7. Click *Finish*

The isCOBOL IDE Perspective

The isCOBOL Perspective consists of the following:

- isCOBOL Explorer
- Editors
- Outline

- Properties
- Problems
- Console
- Search
- Bookmarks
- Tasks
- History
- Error Log
- Customization

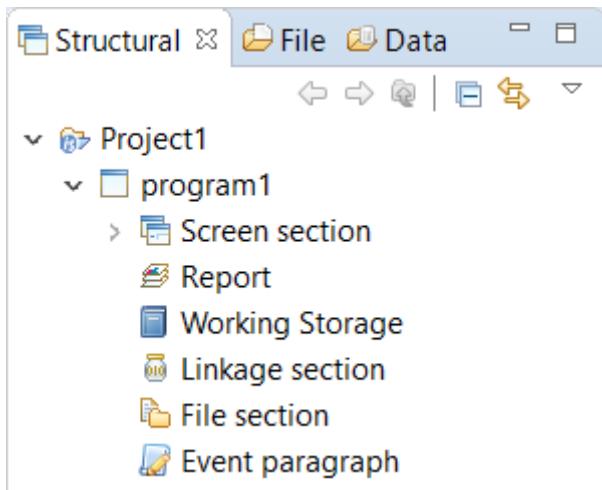
isCOBOL Explorer

The isCOBOL Explorer view lists all the projects of the current workspace and allows you to manage their resources. By default this view is placed on the left side of the IDE window.

The isCOBOL Explorer consists of three pages:

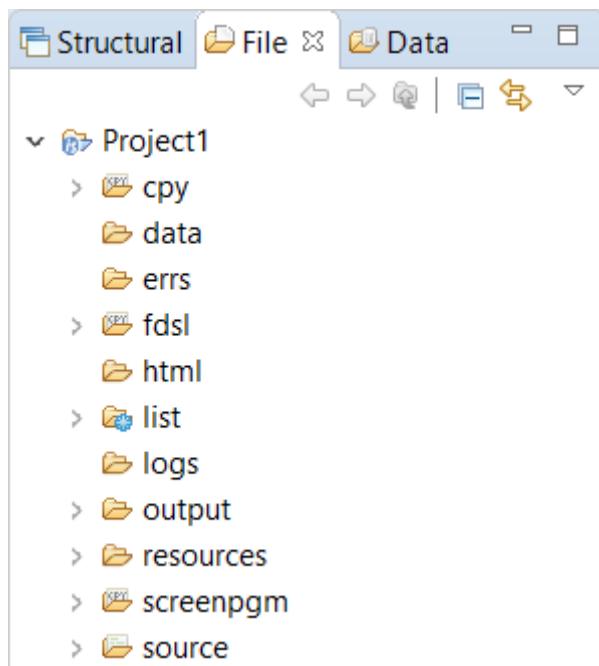
Structural

The *Structural* page lists the Screen Programs available in the project.



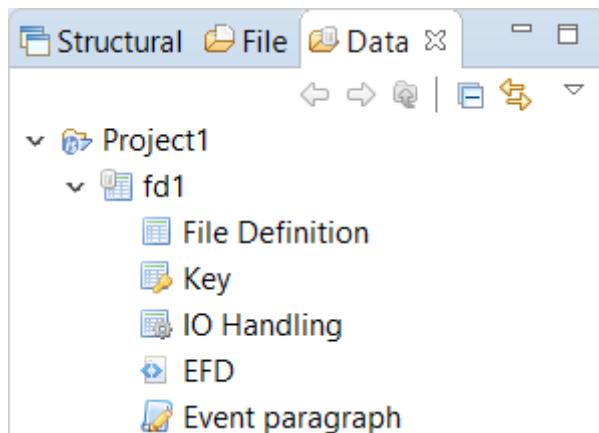
File

The *File* page lists the physical disc files of the project.



Data

The *Data* page lists the Data Layouts available in the project.



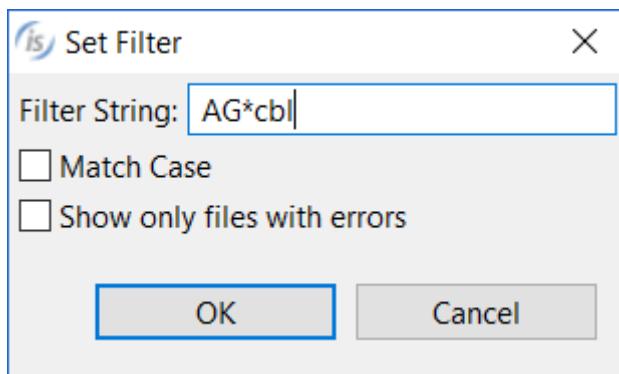
Filtering content

It's possible to set a filter for the programs list in the *Structural* view, the *fd* list in the *Data* view and for the folders in the *File* view.

In order to set or edit a filter

1. select the resource in the view
2. right click and choose *Set Filter* from the pop-up menu
3. fill the dialog with the desired pattern. For example, in order to filter all source files whose name begins

with "AG", set



When the container is filtered, a decoration is added to its icon in the view.

To remove a filter

1. select the resource in the view
2. right click and choose *Remove Filter* from the pop-up menu

To remove all filters in the workspace

1. select the resource in the view
2. right click and choose *Remove All Filters* from the pop-up menu

Editors

Editors create and modify resources. By default, editors are placed in the center of the IDE window. The following editors are available in the isCOBOL IDE:

- [Code Editor](#)
- [Copy View Editor](#)
- [Screen Designer](#)
- [File Designer](#)
- [File Section Designer](#)
- [Working Storage Designer](#)
- [Linkage Section Designer](#)
- [isCOBOL Service Editor](#)
- [JOE Editor](#)

Code Editor

The Code Editor modifies COBOL code as well as other text. COBOL keywords are shown with a different color so they can be easily distinguished from strings, numbers and standard text. The Code Editor also offers a code completion feature and is used as an integrated Debugger.

On the left side of the Code Editor, Bookmarks, Tasks, compiler errors and Debugger breakpoints are shown.

Vertical lines are shown to help you align the code according to the active source format. You can change the active format at any time by right clicking in the editor area and choosing *COBOL Source* from the pop-up menu.

The screenshot shows a COBOL source code editor window titled '*menu.cbl'. The code is as follows:

```

44
25      77  hwin handle of window.
26
27  PROCEDURE DIVISION.
28
29  MAIN.
30  opn input f.
31
32      display standard graphical window
33          background-low
34          title "MAIN MENU"
35
36      call "w$menu" using wmenu-new giving menu-handle
37
38      call "w$menu" using wmenu-new giving sub-handle-1

```

The cursor is at line 30, position 10, under the word 'opn'. A code completion dropdown is open, listing suggestions like 'open', 'openr', 'openu', etc.

Code Completion

In order to take advantage of the code completion feature, you can press **Ctrl+Space** or right click and select "Code Completion" from the pop-up menu. The Editor will suggest some possible items depending on the context.

For object oriented programming, the code completion is automatic. For example, if you write the name of an object reference, followed by ":"> the Editor will automatically list the object methods along with their javadoc.

The screenshot shows a COBOL source code editor window. The code is as follows:

```

CONFIGURATION SECTION.
REPOSITORY.
  CLASS Jstring AS "java.lang.String".
WORKING-STORAGE SECTION.
77 str object reference Jstring.

PROCEDURE DIVISION.

MAIN.
  set str:>
  goback.

```

The cursor is at the end of the line 'set str:>' in the MAIN section. A code completion dropdown is open, listing Java String methods such as 'CASE_INSENSITIVE_ORDER', 'charAt', 'codePointAt', etc. A tooltip provides detailed information about the 'CASE_INSENSITIVE_ORDER' method.

Method Detail: CASE_INSENSITIVE_ORDER

A Comparator that orders `String` objects as by `compareToIgnoreCase`. This comparator is serializable.

Note that this Comparator does *not* take locale into account, and will result in an unsatisfactory ordering for certain locales. The `java.text` package provides `Collators` to allow locale-sensitive ordering.

See:
`java.text.Collator#compare(String, String)`

Since:
1.2

Data Items highlight

If you left click on a data-item name, all occurrences of that data-item in the current source file are highlighted.

```

INPUT-OUTPUT SECTION.
FILE-CONTROL.
  SELECT F ASSIGN TO RANDOM
    STATUS F-STATUS.
FILE SECTION.
FD F.
01 F-REC PIC X(11).

WORKING-STORAGE SECTION.
77 F-STATUS PIC XX.

PROCEDURE DIVISION.
DECLARATIVES.
F SECTION.
  USE AFTER STANDARD ERROR ON F.
  DISPLAY F-STATUS.

```

When a data-item has been highlighted, it's possible to add it to the list of monitored variables in the Debugger perspective. Right click on the item name and select "Add variable" from the pop-up menu.

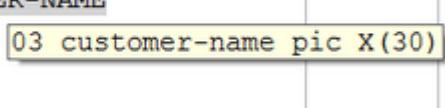
Tooltip items definition

When you leave the mouse pointer on the name of a data item, a tooltip appears and shows the item definition.

```

move low-value to CUSTOMER-NAME.
start customer-file key is not less CUSTOMER-NAME
  invalid
    display "File empty"
end-start.

```



Hyperlink to item definition

By leaving the mouse pointer over a paragraph name or a data item name, in addition to the data definition tooltip mentioned above, you get the following feature: the mouse pointer changes to a hand shape and the item name is underlined. By left clicking in that moment, the Editor jumps to the item definition.

```

53      03 radio-button
54          line           2
55          col            34
56          title          "Run on Server"
57          group          1
58          group-value    3
59          value          rb-value
60          exception-value 103
61          enabled         e-remote
62
63      03 label
64          title          "File"

```

Text Selection upon mouse clicks

By multiple clicking within a text string, you can obtain different text selections.

One click places the cursor within the string:

DISPLAY "Hello World!".

Two clicks among the letters of a word select the whole word:

DISPLAY "Hello World!".

Two clicks between the quotes and the first digit after the quotes select all the quoted text:

DISPLAY "Hello World!".

Three clicks select the whole line:

DISPLAY "Hello World!".

Block Selection

This feature allows the user to select a block of code by drawing a rectangle in the source area. This is also known as Rectangle Selection or Vertical Selection. To activate this feature press Alt+Shift+A. The cursor shape will change to a cross and it will be possible to select a block of text like shown in the picture below. To restore the standard cursor and stop taking advantage of the feature, press Alt+Shift+A again.

```

WORKING-STORAGE SECTION.
77 tabe pic 9 occurs 10.

PROCEDURE DIVISION.

MAIN.

move 0 to tabe(1) .
move 1 to tabe(2) .
move 2 to tabe(3) .
move 3 to tabe(4) .
move 4 to tabe(5) .
move 5 to tabe(6) .
move 6 to tabe(7) .
move 7 to tabe(8) .
move 8 to tabe(9) .
move 9 to tabe(10) .

goback.

```

Text case altering

When a block of text is selected, you can change the case of its words with one of the following shortcuts.

CTRL+G	capitalize text
CTRL+U	make text lower case
CTRL+SHIFT+U	make text upper case

The same functions are available in the *Source* menu.

Correct Indentation

When a block of text is selected, you can ask the Editor to correct the indentation of the rows with the following shortcut.

CTRL+I

The same function is available in the *Source* menu.

Open Declaration

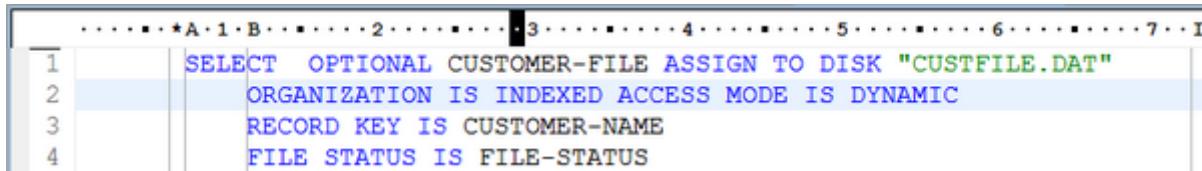
This feature allows the user to jump to a paragraph or variable declaration. To take advantage of it, select the variable or paragraph name, then press F3 or right click and select "Open Declaration" from the pop-up menu.

The function moves the cursor in the Editor to highlight the line where the item is defined. If you don't like this behavior, check the option "Open Declaration feature always in a new editor" in Editor's preferences (see

[Setting Editor preferences](#)). In this case, the function opens a new Editor with a copy of the source. In the new Editor the cursor is on the item declaration, while in the original editor the cursor is still on the line where the function was invoked.

Horizontal Ruler

This feature causes a ruler bar to be shown at the top of the editor. It helps to monitor the cursor position while typing text.



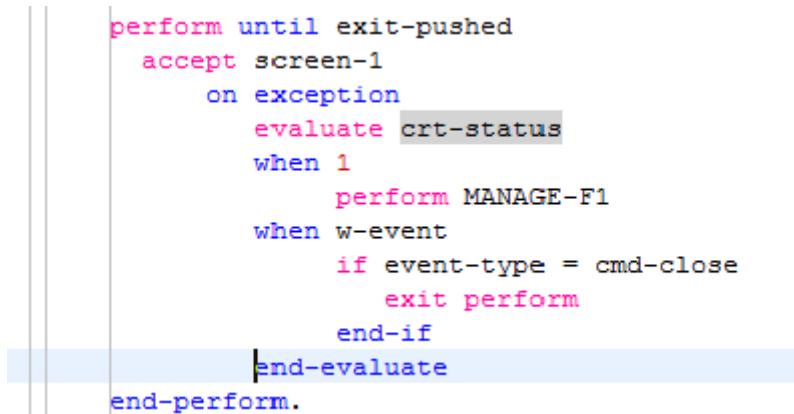
The screenshot shows a code editor window with a horizontal ruler bar at the top. The ruler bar has numerical markings from 1 to 7 and letter markers A, B, I. Below the ruler, there is a line of COBOL code:

```
1   SELECT OPTIONAL CUSTOMER-FILE ASSIGN TO DISK "CUSTFILE.DAT"
2   ORGANIZATION IS INDEXED ACCESS MODE IS DYNAMIC
3   RECORD KEY IS CUSTOMER-NAME
4   FILE STATUS IS FILE-STATUS
```

The ruler is not shown by default, it must be activated in the Preferences. See [Setting Editor preferences](#) for details.

Find Scope

This feature allows to highlight a block. This is useful to double check if all blocks are closed correctly in a source file where there are a lot of nested statements. To take advantage of this feature, place the cursor on a statement or on a END-statement.



The screenshot shows a code editor with the 'Find Scope' feature active. A specific block of code is highlighted with a light blue background:

```
perform until exit-pushed
  accept screen-1
    on exception
      evaluate crt-status
        when 1
          perform MANAGE-F1
        when w-event
          if event-type = cmd-close
            exit perform
          end-if
        end-evaluate
    end-perform.
```

After it, press CTRL+SHIFT+S to highlight the whole block. Alternatively it's possible to right click and choose *Source -> Find Scope* from the pop-up menu.

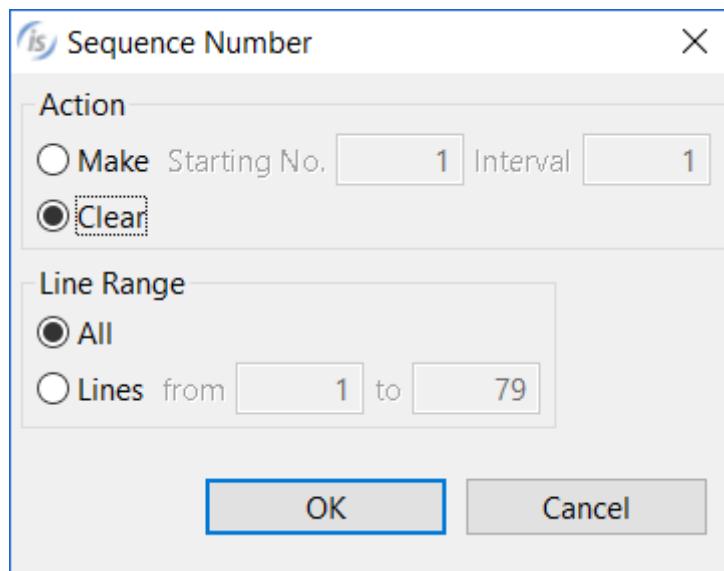
```

perform until exit-pushed
  accept screen-1
    on exception
      evaluate crt-status
      when 1
        perform MANAGE-F1
      when w-event
        if event-type = cmd-close
          exit perform
        end-if
      end-evaluate
    end-perform.

```

Sequence Number

This feature allows to generate line numbers in the sequence number area (columns 1 to 6) of ANSI source files. To activate it, click on the *COBOL Source* menu in the menu bar or right click in the code editor and choose *COBOL Source* from the pop-up menu, then choose *Sequence Number*. The following dialog is shown allowing to provide line numbering rules.



Action allows to choose between clearing existing line numbers or generating new ones starting from a given number (default: 1) and using a given interval (default: 1).

Line Range allows to specify which lines in the source file will be updated. By default, all lines are updated.

```

1 000001/*> Copyright (c) 2005 - 2017 Veryant. Users of isCOBOL
2 000002/*> may freely modify and redistribute this program.
3 000003
4 000004 IDENTIFICATION DIVISION.
5 000005 PROGRAM-ID. ISCHECK.
6 000006
7 000007 ENVIRONMENT DIVISION.
8 000008 CONFIGURATION SECTION.
9 000009 REPOSITORY.
10 000010      CLASS out-of-memory AS "java.lang.OutOfMemoryError".
11 000011
12 000012 INPUT-OUTPUT SECTION.
13 000013 FILE-CONTROL.
14 000014
15 000015 DATA DIVISION.
16 000016 FILE SECTION.
17 000017
18 000018 WORKING-STORAGE SECTION.
19 000019 copy "isfilesys.def".
20 000020 01 cmd      pic x(10).
21 000021 01 max-threads pic 9(18).
22 000022 01 max-files   pic 9(18).
23 000023 01 max-locks   pic 9(18).
24 000024 01 max-memory  pic 9(18).
25 000025 77 pic 9 value 0.
26 000026     88 max-threads-error value 1.
27 000027     88 max-files-error  value 2.
28 000028     88 max-memory-error value 3.
29 000029
30 000030 77 file-io   pic x(256).
31 000031 01 sio-lparms.
32 000032 03 max-rec-sz pic 9.
-- ~~~~~~    no fillin    pic "         " "

```

List of referenced items

Every time the source file in the Code Editor is automatically or manually compiled, the *Properties* view is populated by three lists:

- the list of referenced copybooks
- the list of called programs
- the list of invoked objects

By double clicking on the name of a copybook, the file is opened in a new editor panel.

Property	Value
List of copy files	
iscrt.def	
isgui.def	
List of CALLED programs	
CSFILEINFO	Library Routine
PRINTPROG	Program
List of INVOKED classes	

Copy View Editor

The Copy View Editor allows to see copybooks content in the same Editor as the source file, without opening them in separate tabs. It is useful to have a COBOL source with the inclusion of all copy files to make a full text search that includes also all used copy files.

To open a cobol file with the expanded editor you have to select it in the source folder, right-click and select *Open With -> isCOBOL Copy View Editor*.

The expanded editor is read-only, but it will reflect the changes of the original sources if they have been done with another eclipse text editor, e.g. with the standard Code Editor.

The lines of the copybooks have a different background and the line number is relative to the original file.

```

-----A'1'B-----2-----3-----4-----5-----6-----7-----I-----8-----
31      *end (iscobol)file-section
32      working-storage section.
33      *begin (iscobol)is-def
34      copy "isgui.def".
1 *>***** Copyright (c) 2005 - 2015 Veryant. Users of isCOBOL ****<*
2 *>***** may freely modify and redistribute this program. ****<*
3 *>***** ****<*
4 *>***** ****<*
5
6      >>SOURCE FORMAT FREE
7
8 78  mb-ok                      value 1.
9 78  mb-yes-no                  value 2.
10 78  mb-ok-cancel              value 3.
11 78  mb-yes-no-cancel          value 4.
12 78  mb-retry-cancel          value 5.
13 78  mb-abort-retry-ignore   value 6.
14 78  mb-cancel-retry-continue value 7.
15

```

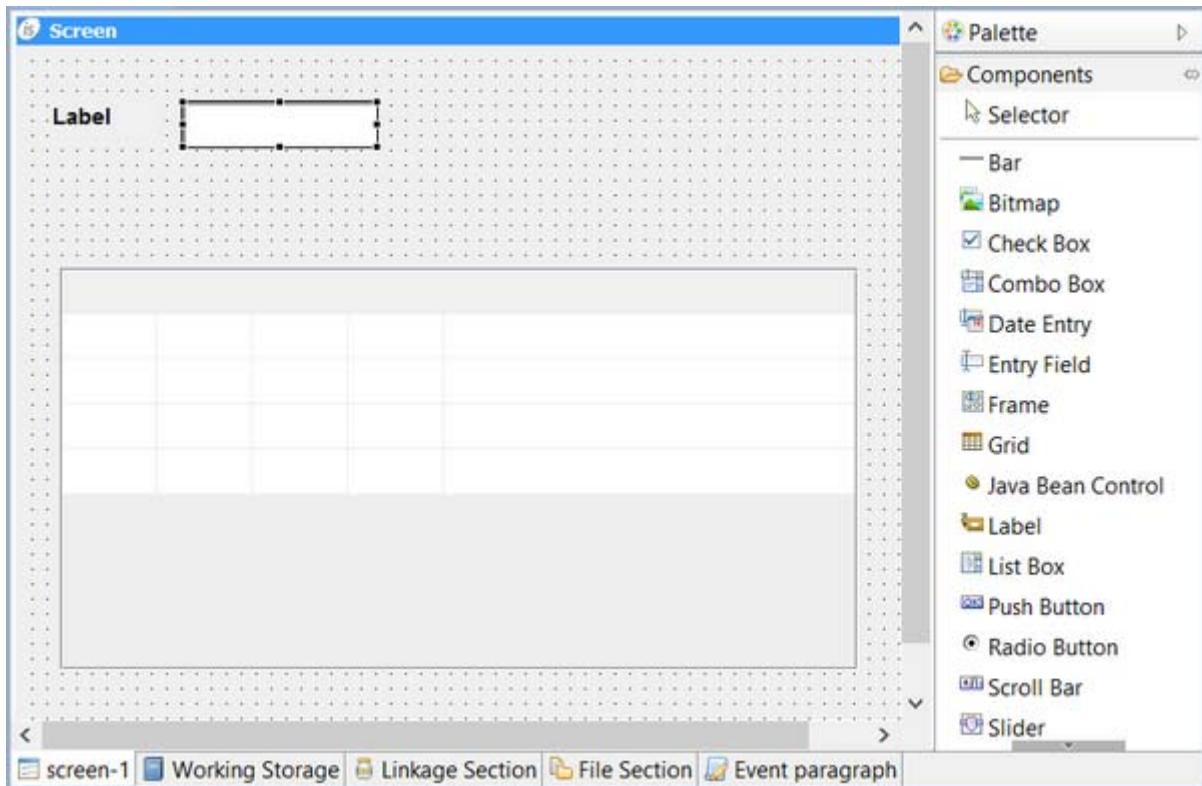
By typing F4, or selecting the *Open isCOBOL Editor* item in the editor's context menu, you can open the standard Code Editor for the file that corresponds to the current line of the Copy View Editor (i.e. the line of

the current cursor position).

The copybooks background color can be configured. See [Setting Editor preferences](#) for details.

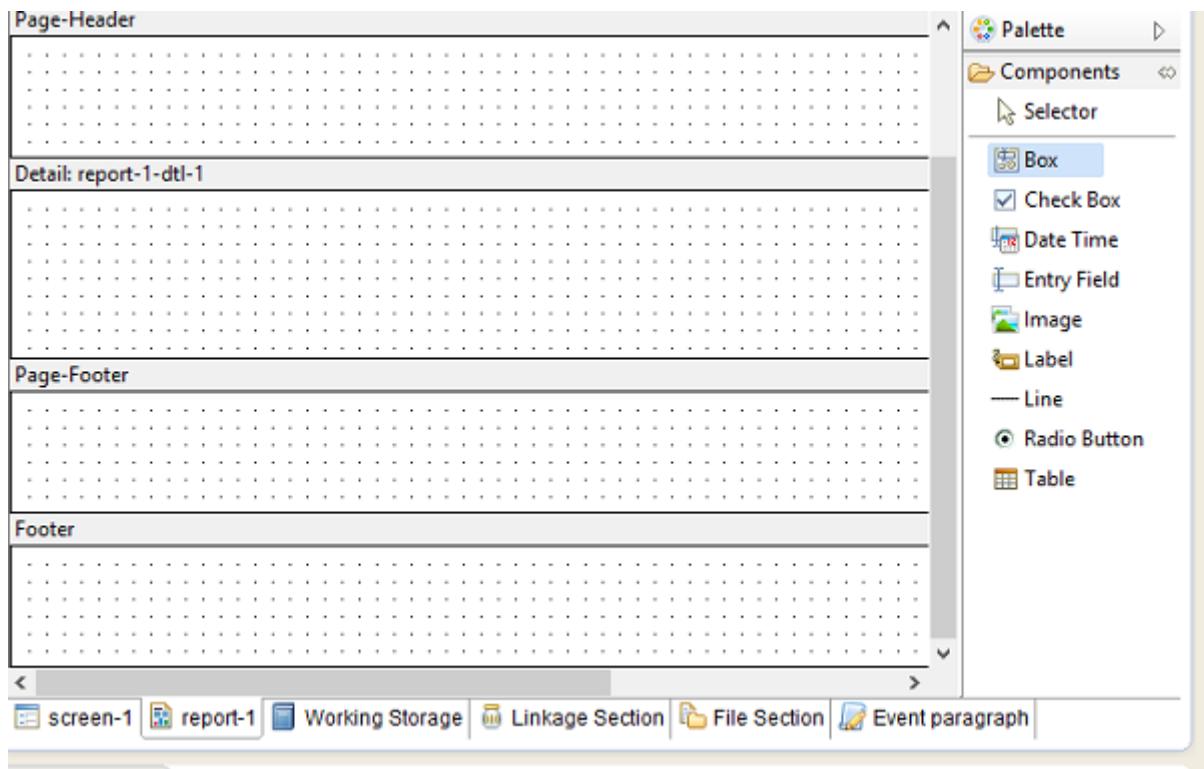
Screen Designer

The Screen Designer builds graphical screens by drawing controls with the mouse. The isCOBOL IDE generates the COBOL code for the screen.



Report Designer

The Report Designer builds graphical reports by drawing controls with the mouse. The isCOBOL IDE generates the COBOL code for printing these reports to HTML file or printer.



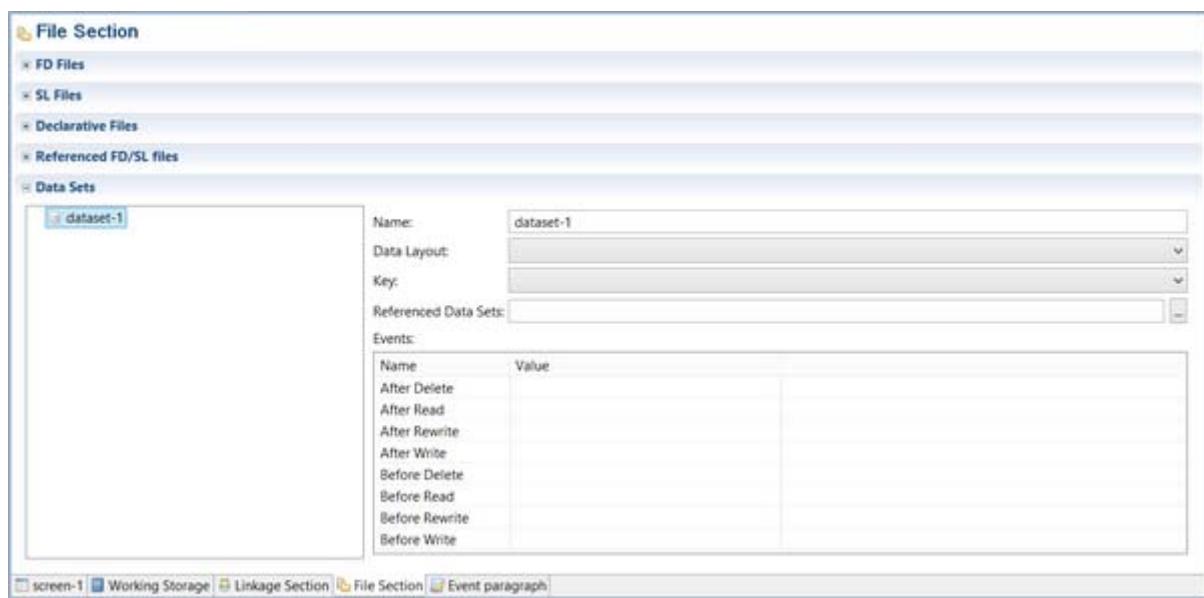
File Designer

The File Designer creates new file definitions through a graphical interface. The isCOBOL IDE generates FD/SL copybooks for each one of these files.



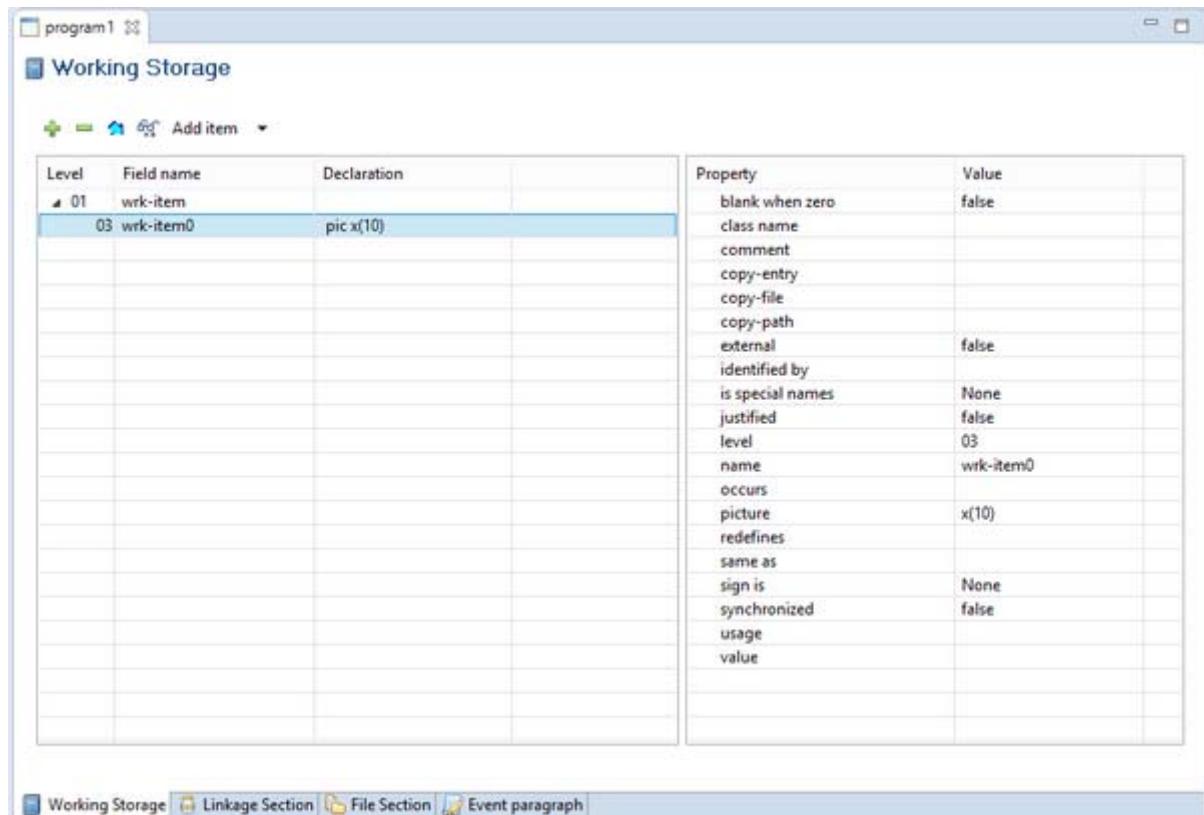
File Section Designer

The File Section Designer associates some files with the current program.



Working Storage Designer

The Working Storage Designer defines data items for the program's Working-Storage Section as well as including existing copybooks. The isCOBOL IDE will generate the COBOL code for the program's Working-Storage Section.



Linkage Section Designer

The Linkage Storage Designer defines data items for the program's Linkage Section as well as including existing copybooks. The isCOBOL IDE will generate the COBOL code for the program's Linkage Section.

Linkage Section

The screenshot shows the Linkage Section Designer interface. At the top, there are icons for adding a new item, deleting, and saving, followed by the text "Add item". Below this is a table divided into two sections: "Declaration" and "Properties".

Level	Field name	Declaration	Property	Value
01	Iks-item	pic x(10)	blank when zero	false
			class name	
			comment	
			copy-entry	
			copy-file	
			copy-path	
			external	false
			identified by	
			is special names	None
			justified	false
			level	01
			name	Iks-item
			occurs	
			picture	x(10)
			redefines	
			same as	
			sign is	None
			synchronized	false
			usage	
			value	

isCOBOL Service Editor

The isCOBOL Service Editor activates and configures the Service Bridge facility. When you open a source file with this editor, you're allowed to map the Linkage Section parameters to Web Service parameters as well as choosing the Web Service type and characteristics. As soon as you save modifications, the original source code is updated with the proper compiler directives.

The screenshot shows the isCOBOL IDE interface with the following details:

- Service Settings Tab:**
 - Enable Service:** Checked.
 - Type:** SOAP
 - Style:** RPC
 - Prefix:** soap
 - URL:** http://localhost:8080
 - Namespace:** http://tempuri.org
 - Decorations:** Default
 - Operations:** procedure reguser
- Data Map Tab:**
 - Linkage Section Fields:**

Data Item	Value
01 input-parms	03 user-fname pic x(64) 03 user-sname pic x(64) 03 user-email pic x(64) 03 user-addr pic x(96) 03 user-city pic x(32) 03 user-state pic x(32)
01 op-result	03 op-status pic 9 03 reason pic x(128)
 - Service Fields:**

Data Item	Name	Direction	Type
input-parms	user-fname	input	string
	user-sname	input	string
	user-email	input	string
	user-addr	input	string
	user-city	input	string
	user-state	input	string
op-result	op-status	output	string
	reason	output	integer

JOE Editor

The JOE Editor modifies JOE scripts code. Reserved keywords are shown with a different color so they can be easily distinguished from strings, numbers and standard text.

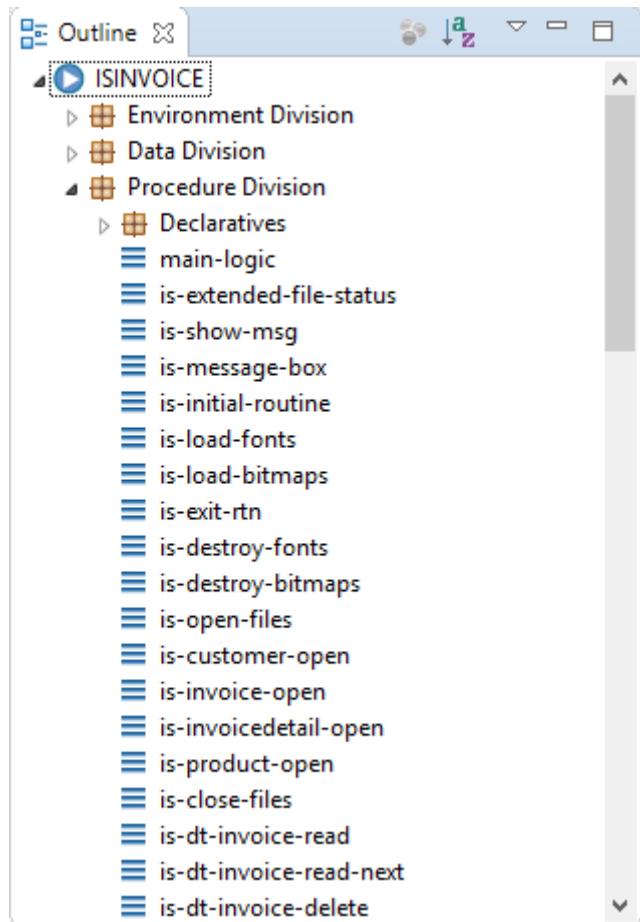
The screenshot shows a window titled "validator.joe" containing COBOL source code. The code defines two procedures: checkSurname and checkCompany. The checkSurname procedure checks if a surname is "DOE" or "SMITH". If either is true, it prints "Yes, he's a member of Veryant staff". Otherwise, it prints "Veryant staff only allowed here!". The checkCompany procedure checks if a company is "VERYANT". If it is, it prints "Yes". Otherwise, it prints "Only Veryant is allowed!". Both procedures return a value.

```
1  ** isCOBOL exec JOE */
2
3  checkSurname := {:a_value.
4    a_surname := a_value toString ; toUpperCase ; trim.
5    !if ((a_surname = "DOE") or (a_surname = "SMITH")),(
6      "Yes, he's a member of Veryant staff".
7    ),(
8      "Veryant staff only allowed here!".
9    )
10   .
11
12  checkCompany := {:a_value.
13    a_company := a_value toString ; toUpperCase ; trim.
14    !if (a_company = "VERYANT"),(
15      "Yes".
16    ),(
17      "Only Veryant is allowed!".
18    )
19  ).
```

Outline

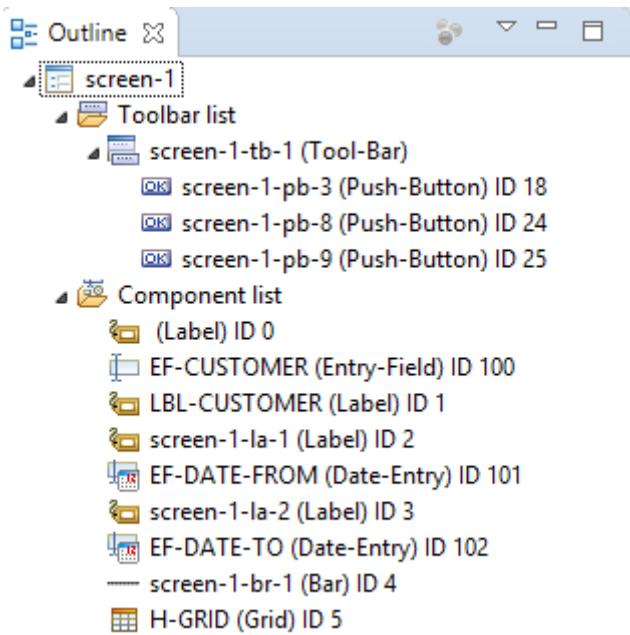
The Outline view displays an outline of a structured file that is currently open in the editor area, and lists structural elements.

Code Editor's Outline



When the focus is in the Code Editor, the Outline view provides the list of sections and paragraphs of the current program. By clicking on the section or paragraph name, the corresponding line in the editor is highlighted.

Screen / Report Designer's Outline

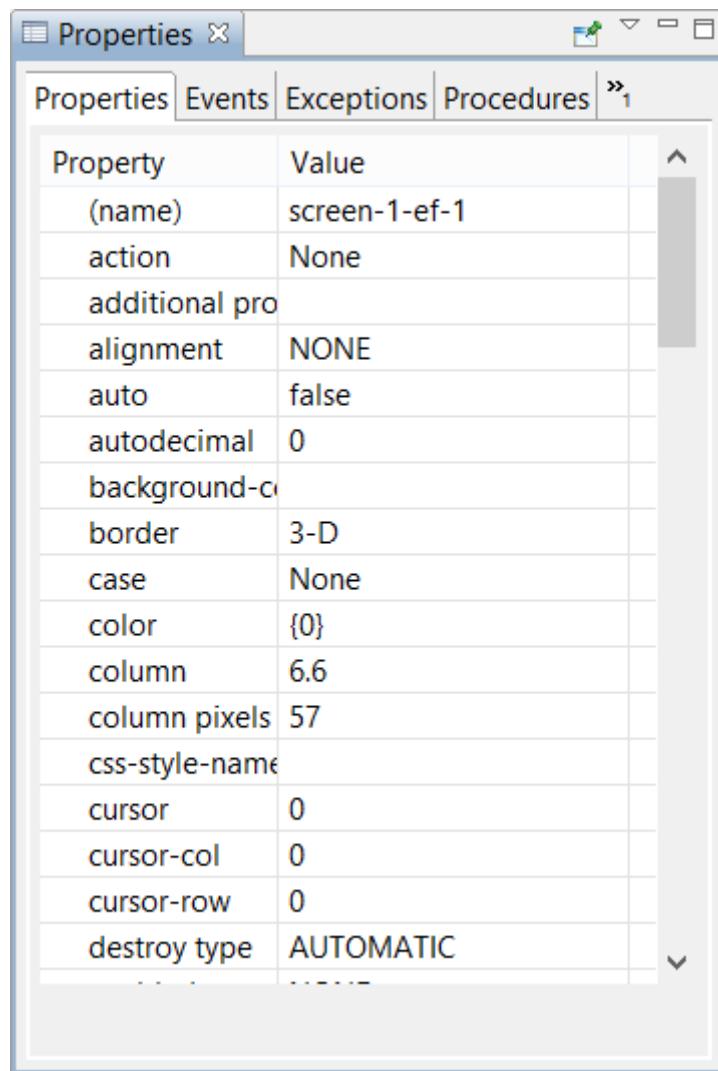


While editing a Screen or a Report, the Outline view provides the list of controls. Name, type and ID are shown for each control. Clicking on a control selects it in the Designer.

Properties

The Properties view shows the list of properties and styles that can be set for the current control selected on the Screen Designer.

The Properties view also allows you to define Embedded and Event procedures for the control.



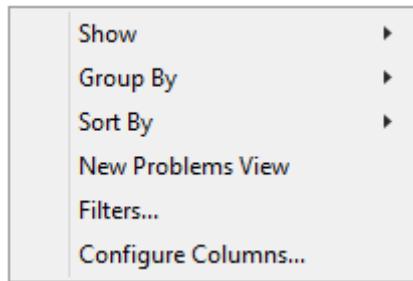
Problems

The Problems view lists compiler errors and warnings.

The Problems view displays a table of compiler issues. The table has four columns: Description, Resource, Path, and Location. There are two error entries: one for an undefined data item VAR and another for a file containing errors. Both entries are categorized as 'Problem'.

Description	Resource	Path	Location	Type
Errors (1 item) --S: #35 Undefined data item VAR	program1.evt	/COBOL/screenpgm	program1.cbl [id=00...]	Problem
Infos (1 item) Copy file contains errors	program1.cbl	/COBOL/source	program1.cbl [id=00...]	Problem

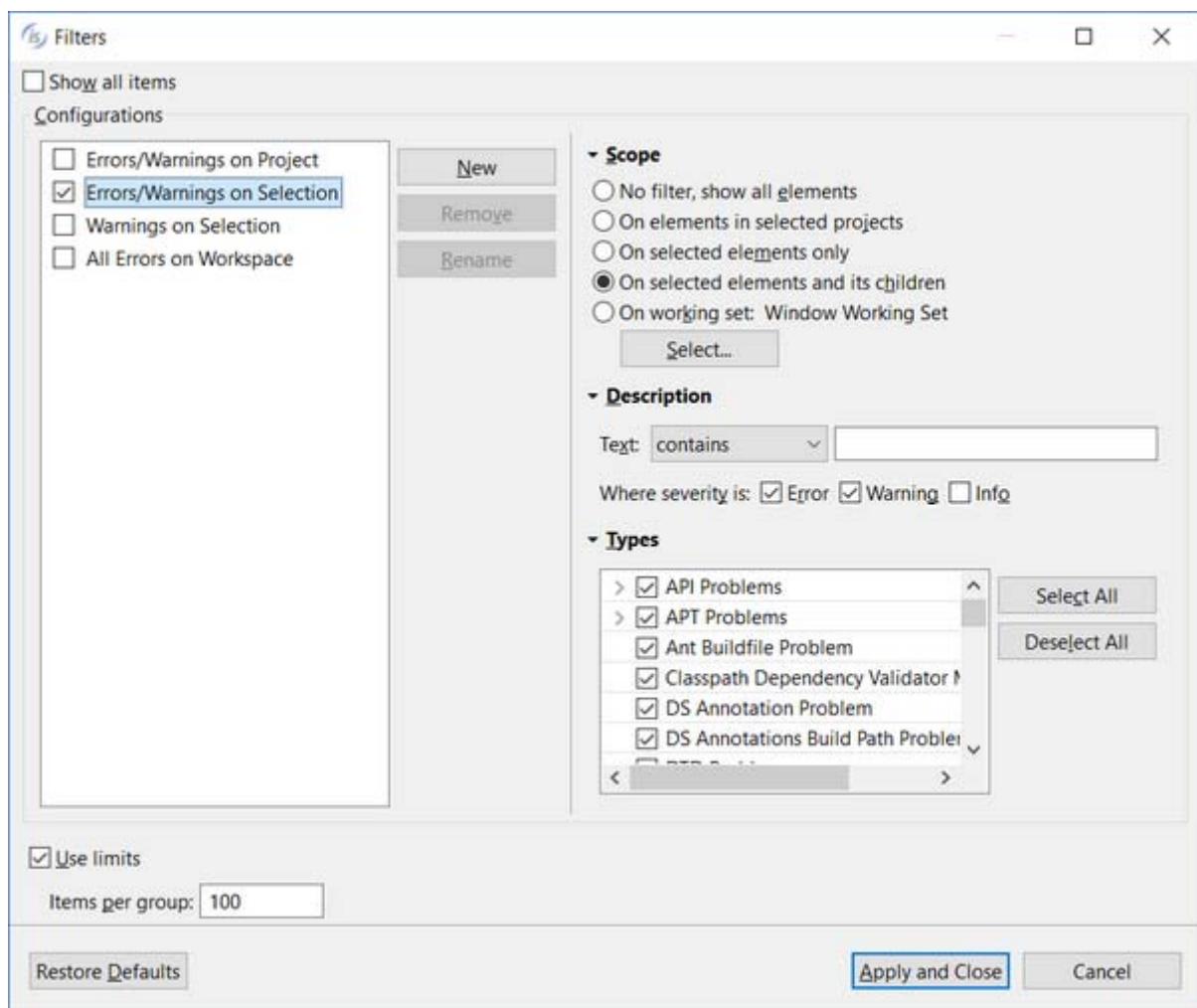
The content of the Problems view and its layout can be configured via the pull-down menu:



- *Show* allows you to filter the types of errors that must be shown. By default all types of errors are shown.
- *Filters...* pops up a dialog in which you can set advanced filters.
- *Configure Columns...* allows you to configure the order and the width of the columns.

Suggested Configuration

For a clear and precise output in the *Problems* view, Veryant recommends the following configuration, that you can set in the *Filters* dialog:



- "Show all items" should be unchecked
- only "Errors/Warnings on Selection" should be checked in the list
- Scope should be set to "On selected elements only"

Also, it's best practice to sort problems by Location rather than by Description. You can set this preference by selecting *Sort By* in the pull-down menu.

Console

The Console view displays the content of the system output and the system error.

```
<terminated> PROG1 [isCOBOL Application] PROG1
Converting file 1...
Conversion done.
```

Search

The Search view shows the result of research features available in the *Search* menu.

'move' - 113 matches in workspace

- COBOL
 - cpy
 - isfilesys.def
 - 26: 88 remove-function value 13.
 - isgui.def (2 matches)
 - ismsg.cpy (47 matches)
 - isresize.def (12 matches)

Bookmarks

The Bookmarks view displays all Bookmarks set in the current Workspace.

Description	Resource	Path	Location
CONVERSION-ROUTINE.	PROG1.cbl	/COBOL/source	line 30

To create a new Bookmark, right click on the vertical gray bar at the left of the Code Editor and choose *Add Bookmark...* from the popup menu.

To reach a Bookmark, double click on its row in the view.

Tasks

The Tasks view displays all Tasks set in the current Workspace.

Tasks				
	Description	Resource	Path	Location
<input type="checkbox"/>	*TODO: retrive desktop heigh/witdh and adapt window	menu.cbl	/Project1/source	line 17

To create a new Task, right click on the vertical gray bar at the left of the Code Editor and choose *Add Task...* from the popup menu.

A new Task is automatically created for each commented line that includes the keywords *TODO* or *FIXME*.

The screenshot shows the Code Editor with COBOL code. Line 16 contains a TODO comment: *TODO: retrive desktop heigh/witdh and adapt window. This comment is highlighted in yellow. Below the editor is the Tasks view, which lists a single task:

Tasks				
	Description	Resource	Path	Location
<input type="checkbox"/>	*TODO: retrive desktop heigh/witdh and adapt window	menu.cbl	/Project1/source	line 17

To reach the source line associated to the Task, double click on the Task row in the Tasks view.

History

The History view lists all the changes made to a text file. Each line corresponds to a saved copy of the file. From here it's possible to compare two different copies of the same file to see what's different and also to restore a previously saved copy.

History				
PROG1.cbl				
Revision Time				
<input type="checkbox"/>	22/04/13 14:14			
<input type="checkbox"/>	22/04/13 12:58			
<input checked="" type="checkbox"/>	22/04/13 12:55			
<input type="checkbox"/>	22/04/13 12:55			
<input type="checkbox"/>	22/04/13 12:54			

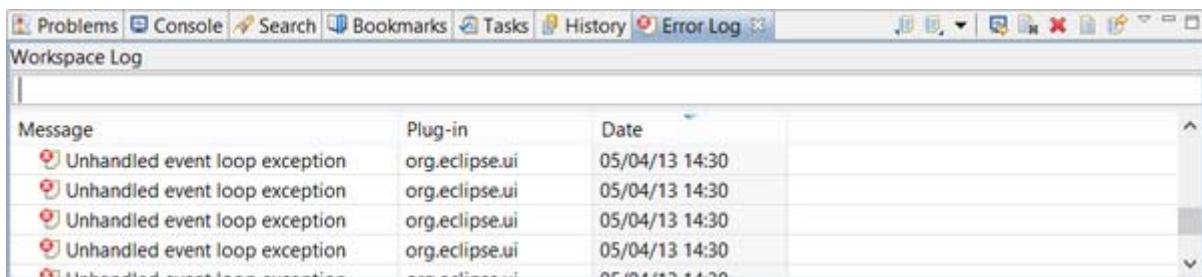
To populate the History view, right click in the Code Editor and choose *Team / Show Local History* from the popup menu.

To manage history entries, right click on their row in the view and choose the proper option from the popup

menu.

Error Log

The Error Log view shows Java Exceptions produced by the IDE plugins.



To clear the Error Log, right click on it and select 'Delete Log' from the pop-up menu.

Customization

Each one of the IDE views can be moved, resized, hidden or detached. The IDE perspective is fully customizable.

To detach a view, right click on its label and select *Detached* from the popup menu. The same menu item can be used to restore the view into the IDE window.

To move a view left click on its label, then, holding the left mouse button down, drag it to the desired position in the IDE window and release the left mouse button.

To hide a view right click on its label and choose *Close* from the popup menu, or just click on the X icon available on the label.

To show a view open the *Window* menu and choose *Show View*, then choose the desired view from the list.

Views can be easily minimized, maximized and restored through keyboard shortcuts.

- to minimize a View press **CTRL+ALT+M** then release these buttons and press the letter that identifies the view
- to maximize a View press **CTRL+ALT+X** then release these buttons and press the letter that identifies the view
- to restore a View press **CTRL+ALT+R** then release these buttons and press the letter that identifies the view

Letters that identify views are listed in the following table:

View	Letter
Structural	S
File	F
Data	A
Outline	O

View	Letter
Problems	R
Console	C
Search	E
Bookmarks	K
Tasks	T
Error Log	L
History	H
Debug	D
Variables	V
Breakpoints	B
Properties	P

For example: in order to minimize the Structural View, press CTRL+ALT+M, then S.

Working with Projects

Creating a new Project

In order to use isCOBOL IDE to create and maintain COBOL programs, one or more projects must be created. A Project is a container for programs that share the same settings and resources.

To create a new Project in a workspace, right click in the isCOBOL Explorer area and choose *New / isCOBOL Project* from the pop-up menu. A wizard will ask for project name and location, compiler options and additional jar libraries. Most of these settings can be changed later, after the project has been created, by clicking on *Project* menu and selecting *Properties* from the list.

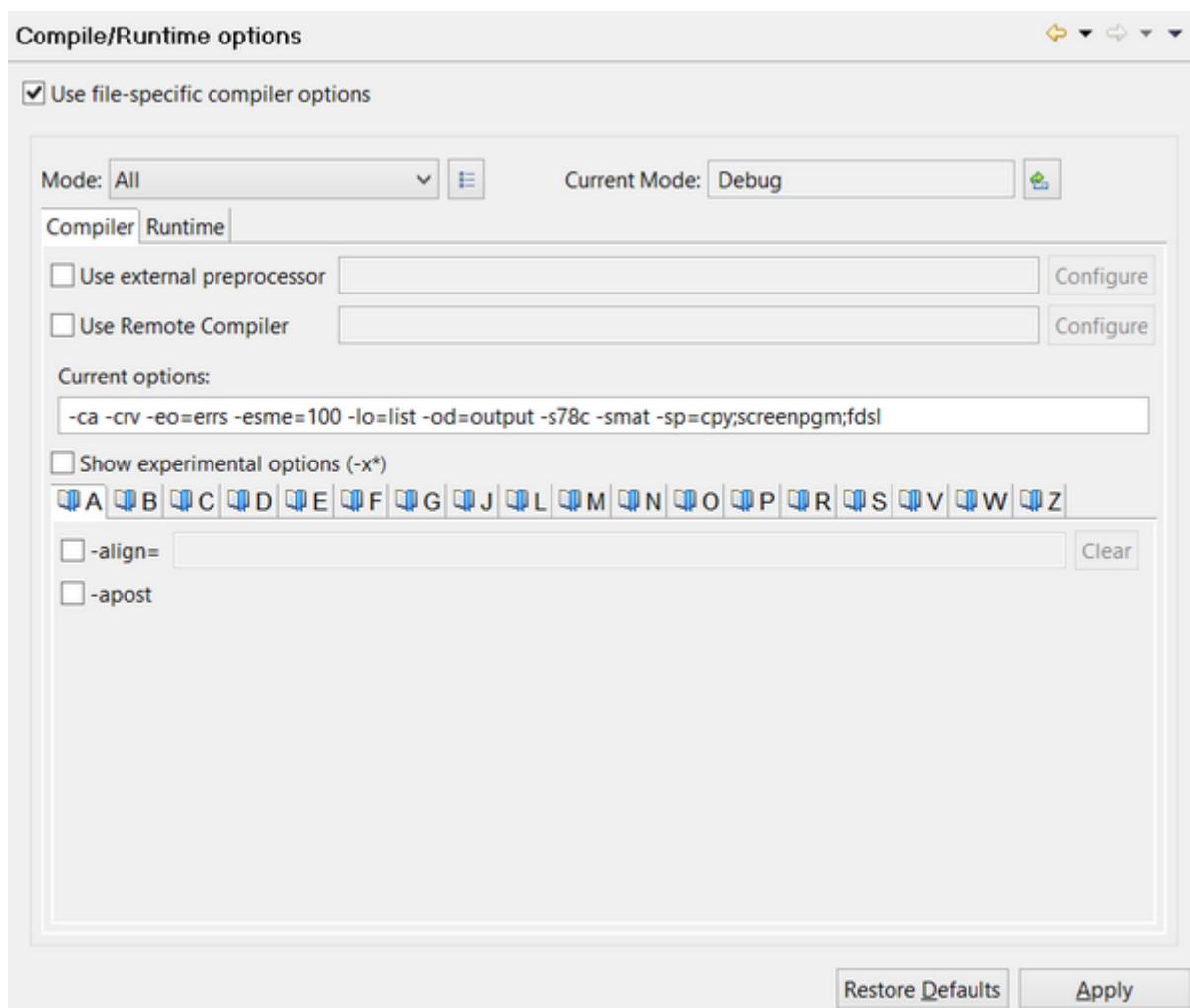
Setting Project properties

Project settings affect all the programs of the project. In order to set them

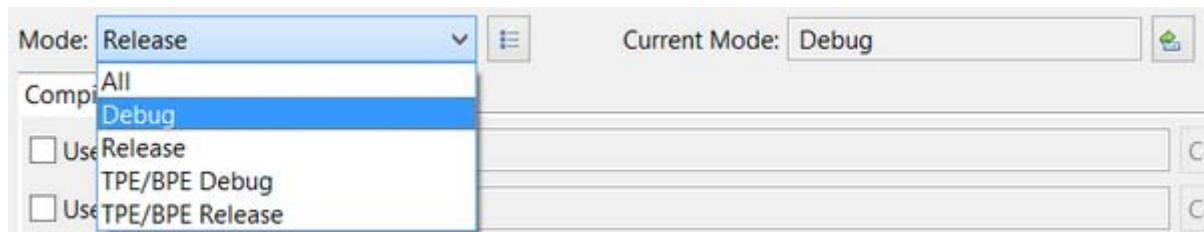
- right click on the project name in the isCOBOL Explorer
- select *Properties* from the pop-up menu.

Compile and Runtime options

Compile and Runtime options are found under the *isCOBOL Settings* item. Here you can specify compile options as well as runtime settings.

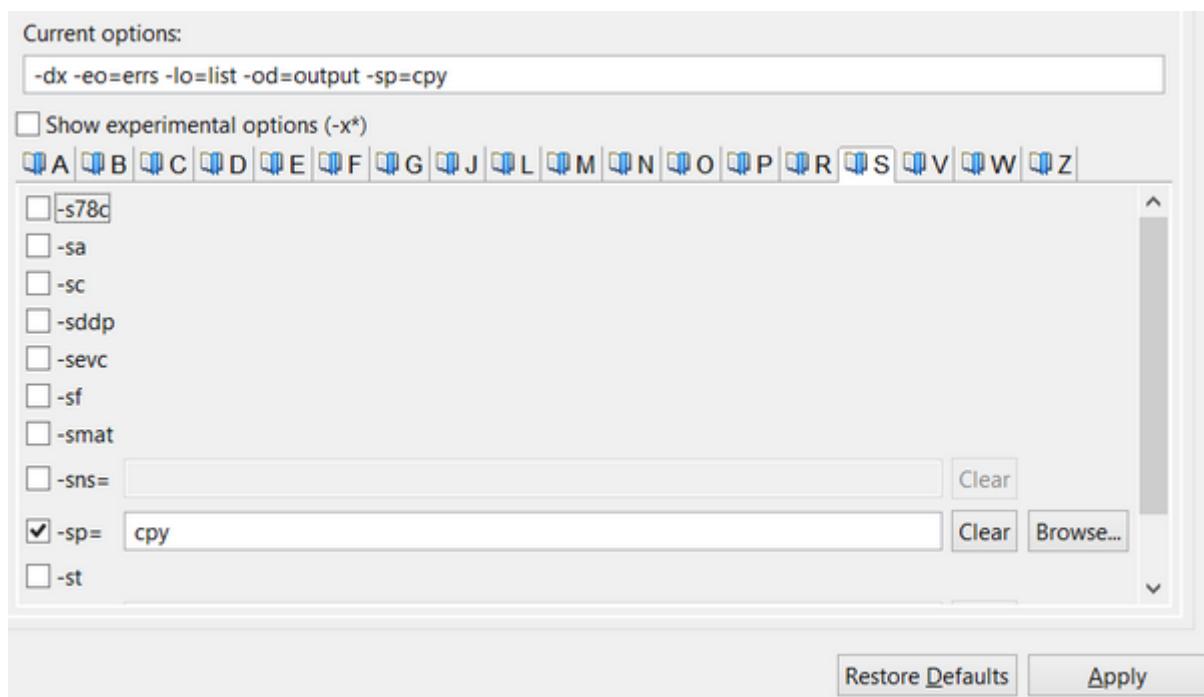


You can create different sets of options. Choose the set you wish to configure from the combo-box.

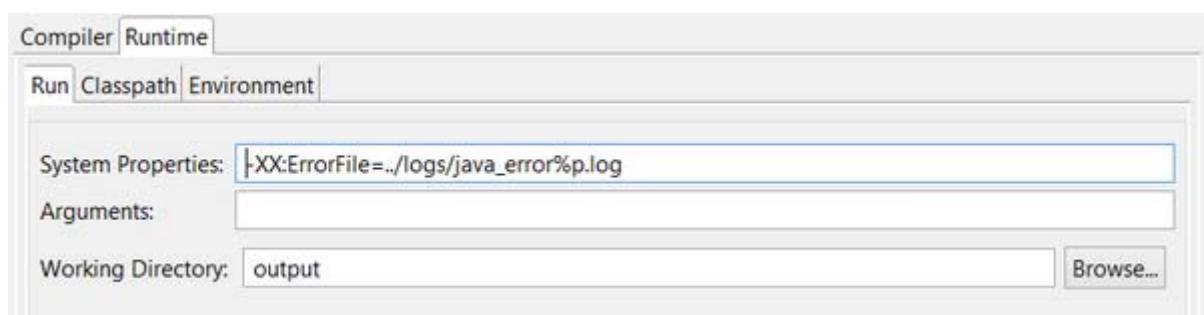


You can switch the active mode by clicking on the *Change* button on the right in this panel, or by clicking on the Project menu and selecting *Change Current Mode*.

In the *Compiler* page, you can set compile options by checking them with the mouse. They're distributed on different pages depending on their name. The list of options that are currently active appears in the text box just above these pages and can be edited from there also.



In the *Runtime* page, you can set command-line options and parameters.



Class Path

The *Class Path* panel allows you to add references to jar libraries or class folders to the project. This is important if you're going to create programs that interact with Java objects.



Screen Designer

The Screen Designer panel allows you to set some defaults for the Screen Designer (see [Screen Programs](#) for more information on this tool).

Property	Value
cell	true
cell measure	PIXELS
cell measuring control	LABEL
cell measuring font	Default-Font
cell measuring style	NONE
cell-height	10
cell-width	10
color	{257}
control font	Default-Font
height-in-cells	false
unit	CELLS
width-in-cells	false

When the "Show only supported features in WD2" option is checked, the properties that are not supported by WD2 are not shown in the *Properties* view and the controls that are not supported by WD2 are not shown in the *Component Palette*. This happens in the new screens that you create. If you open an existing screen that contains some components not supported by WD2, then these components are not modifiable, you can only delete them.

When you save these settings, the IDE asks if you want to apply the changes to the programs already created in the project. This behavior can be configured in the Preferences. See [Setting Screen Designer preferences](#) for details.

These settings are applied to Screen Programs as well as Screens.

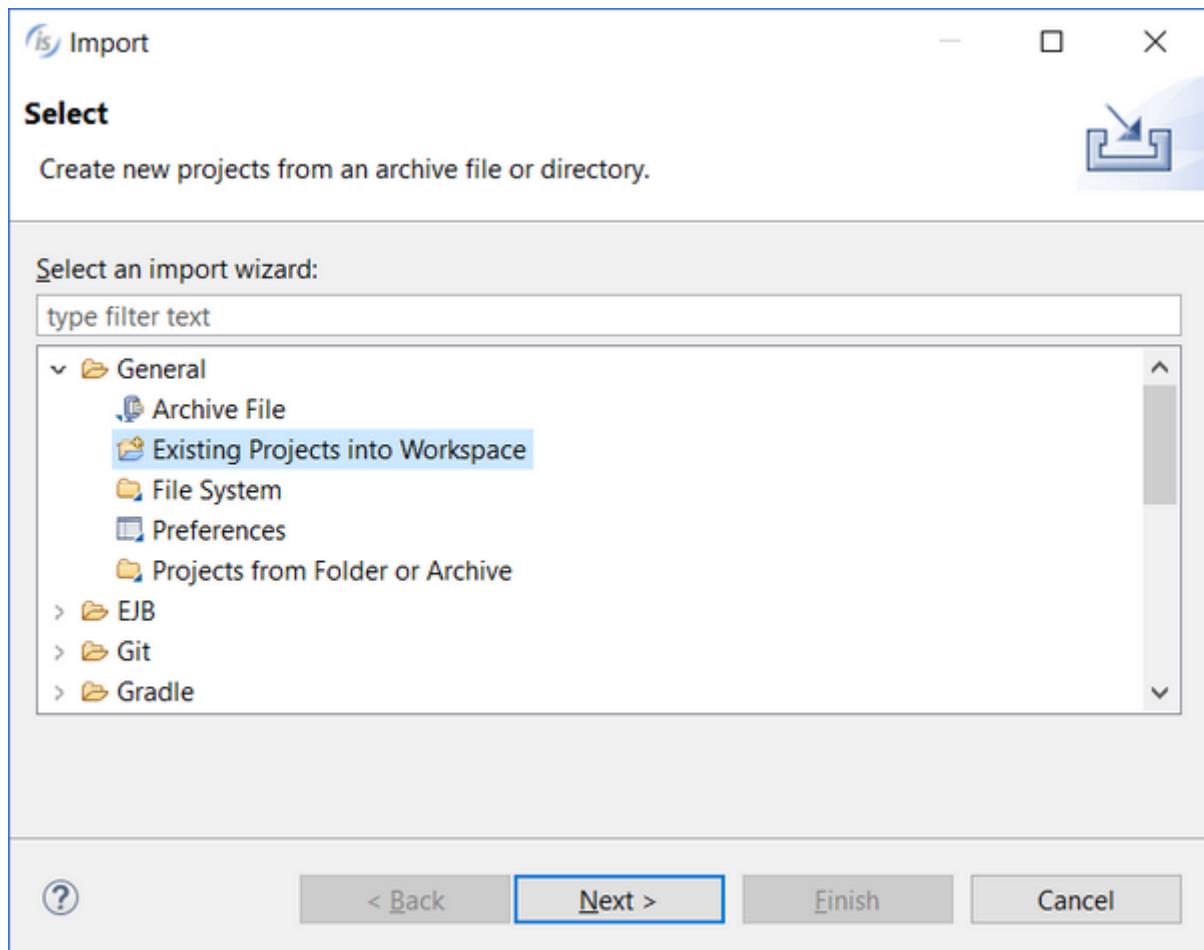
Alternate isCOBOL SDK

If you linked one or more external isCOBOL SDK to the IDE, you can choose which one to use for the current project. Select *isCOBOL Settings > Installed isCOBOL* in order to get the list of available SDKs and check the one you wish to use.

See [Binding additional isCOBOL SDK](#) for more information about how to add external isCOBOL SDKs to the IDE.

Adding an existing Project to the current Workspace

isCOBOL IDE can also import an existing project that is part of another workspace into the current workspace. To import another project, right click in the isCOBOL Explorer area and choose *Import* from the pop-up menu. Then select *General / Existing Projects into Workspace* from the tree.



A wizard procedure will ask for the location of the source workspace (the directory containing a folder named `.metadata`). Once the workspace is found, all of its projects are listed and you can choose the ones you wish to

import.

Creating a new program

The isCOBOL IDE creates and maintains two kinds of programs:

Standard Programs

Standard programs are written from scratch by the user. The user uses the Code Editor to maintain the source code.

To create a new text file (source file or copybook) in the project, right click in the isCOBOL Explorer area and choose *New / Source File* or *New / Copy File* from the pop-up menu.

Note: Every new program is created with the -sf option set by default amongst its compiler options. This means that the IDE will try to compile the source code as a Free format source code. If you plan to write (or paste from other sources) code written in a different format (Ansi or Terminal) you should edit your compiler options accordingly. See [Compiling](#) for details.

Screen Programs

Screen Programs are designed by the user through the Designers views (see [The isCOBOL IDE Perspective](#) for details). The isCOBOL IDE automatically generates the COBOL code for these programs. Screen Programs will be covered in-depth in [Screen Programs](#).

Screen Programs and standard programs can coexist in the same Project.

Adding existing programs to the Project

Standard Programs

In order to add existing source and copy files to the current Project, copy them into the proper project folders using operating system commands, then right click on the project name in the isCOBOL Explorer and choose *Refresh* from the pop-up menu.

Screen Programs

For information on how to add an existing Screen Program to the Project, see [Adding existing Screen Programs to the current Project](#).

Code Editing

Standard Programs

Once a source file or copy file is selected from the isCOBOL Explorer, its content is shown in the Code Editor, at the center of the IDE window. From here it is possible to edit the code.

Screen Programs

Even if the COBOL code is shown in the Code Editor when you select a source file or a copybook, do not modify it directly. In order to modify Screen Programs you need to operate in the Designers views, and then regenerate the code. The editing of the generated code is allowed only outside the tagged areas if the *Regenerate tagged area only* option is set in the *Preferences*.

Tagged areas are special blocks of code delimited by the following comments:

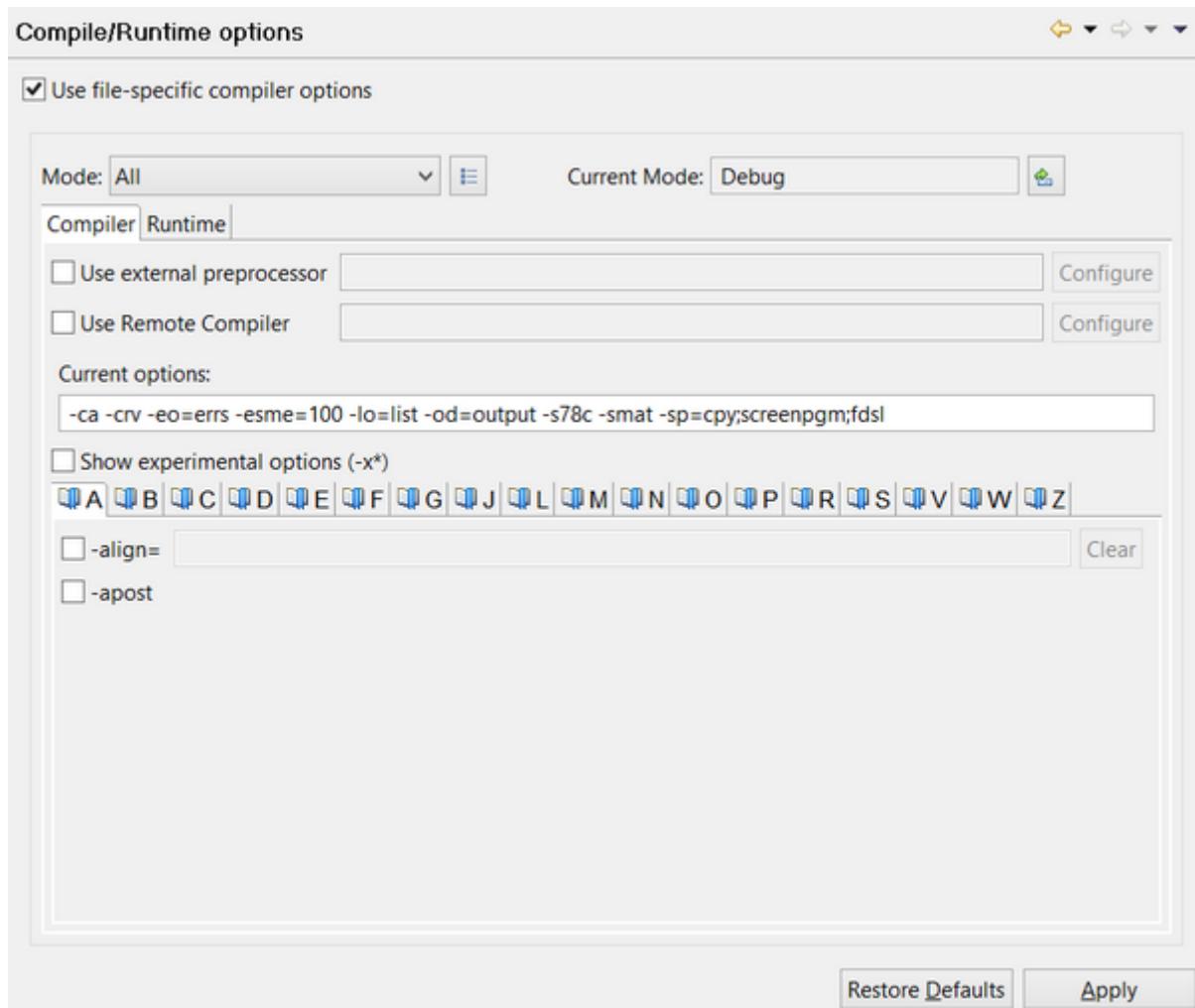
```
* BEGIN {isCOBOL}area-name  
* END {isCOBOL}area-name
```

When the IDE generates the code for Screen Programs,

- if *Regenerate tagged area only* option is set, only tagged areas are overwritten.
- if *Regenerate tagged area only* option isn't set, the whole source code is overwritten.

Compiling

In order to compile programs correctly it is necessary to set the proper compiler options. If this was not done during project creation, it can be done now by clicking on *Project* menu and choosing *Properties*, then *isCOBOL Settings* and *Compile/Runtime options*. Compiler options are distributed in multiple pages of a tab-control. Find and check the ones you need.



In this options panel, it is possible to define different sets of options. By default, the "Debug" set is active. You

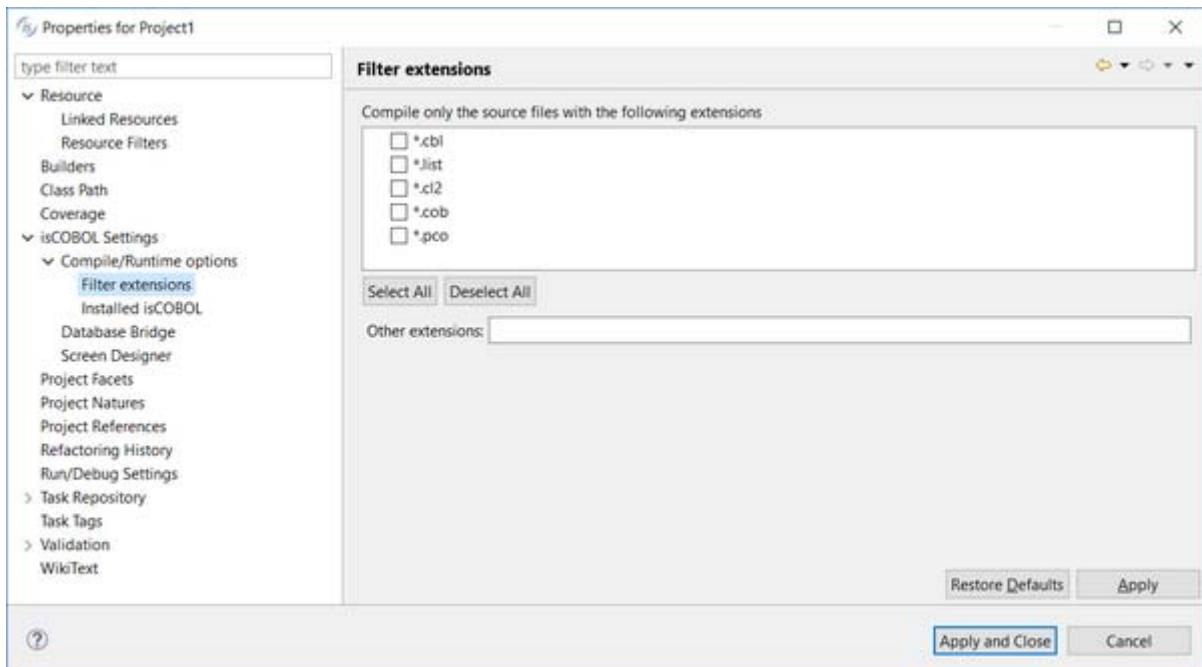
can define options that should be used when compiling for "Debug", for "Release" or for both of them. In this panel you also choose the active set of options that will be used the next time you compile a program.

Placing the mouse cursor over the name of a compiler option will show a pop-up hint with a short description of its effect. To retrieve additional information on compiler options and their effects, please consult [Compiler Options](#).

Note: when specifying -sp value, if you need to add a folder that is part of another project in the same workspace, you have to use the special syntax \${Project-Name}/folder. For your convenience, a *Browse* button is provided after the -sp field. Use the browse function to explore workspace folders and disk folders, select the desired copybook folder and have the correct value in the -sp field.

If *Use file-specific compiler options* is checked, then you can specify different compiler options for every single file in the project source directory, otherwise every file inherits settings from this page. To specify custom options for a specific file, right click on the source file name in the File view of the isCOBOL explorer and choose *Properties* from the popup menu. Then choose *settings* from the tree.

It's also possible to filter the file types that the IDE will try to compile. By default, there is no filter, so every file placed in the *source* folder is subjected to compilation.



The isCOBOL IDE automatically compiles the programs as soon as source files are added to a project or are saved to disk after editing. This keeps the entire project built at all times. This feature can be disabled by clicking on *Project* menu and unchecking *Build Automatically*.

To manually compile a program, click on the following button:



Alternatively, use the keyboard shortcut:

Ctrl+Alt+F9

The program compilation is performed in two steps:

1. compile COBOL to java
2. compile java to class

Errors produced by the first step are traced in the [Problems](#) view.

Errors produced by the second step are traced in the [Error Log](#) view.

Before compiling, it's possible to switch the current mode, choosing between "Debug", "Release" or a custom mode, if any. To switch the current mode for the current project, click on *Project* in the menu bar and choose *Change Current Mode*. To switch the current mode for all the projects in the workspace, click on *Project* in the menu bar and choose *Change Current Mode for All Projects*.

Reloading isCOBOL's copybooks

When you upgrade the IDE or change the isCOBOL Library associated to your project, the isCOBOL copybooks in the *cpy* folder are left unchanged. In order to use new features added in the upgrade, you must manually reload the copybooks. Right click on the *cpy* folder and choose *Reload isCOBOL's copybooks* from the pop-up menu. isCOBOL's standard copybooks (like "isgui.def") are re-loaded from the *isdef* folder of the linked isCOBOL Library, while IDE's exclusive copybooks (like "isreport.def") are re-loaded from the *dropins* folder of the IDE.

Remote Compiling

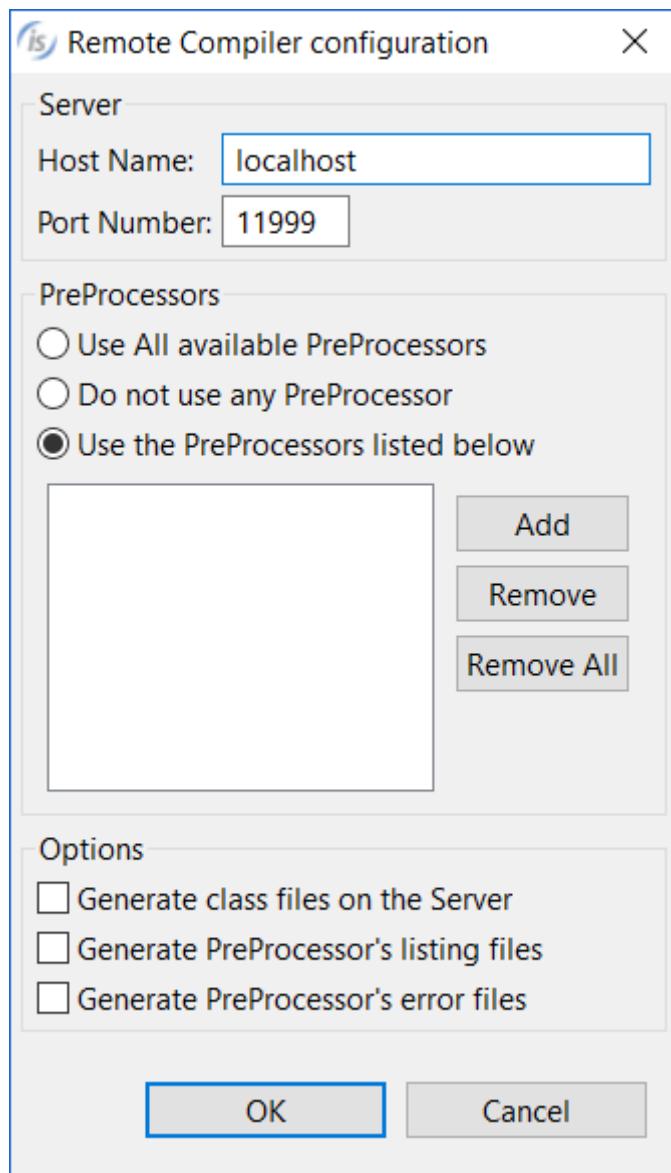
isCOBOL IDE allows you to compile programs remotely. Assuming that there is a Remote Compiler listening on a server, you can instruct the IDE to delegate the compilation to it.

To activate this feature

1. click on *Project* menu
2. choose *Properties*
3. expand *isCOBOL Settings* and select *Compile/Runtime options*
4. check the *Use Remote Compiler* option
5. click on the *Configure* button and input the necessary information

Host Name and *Port* are the address and the port of the server where the Remote Compiler is listening.

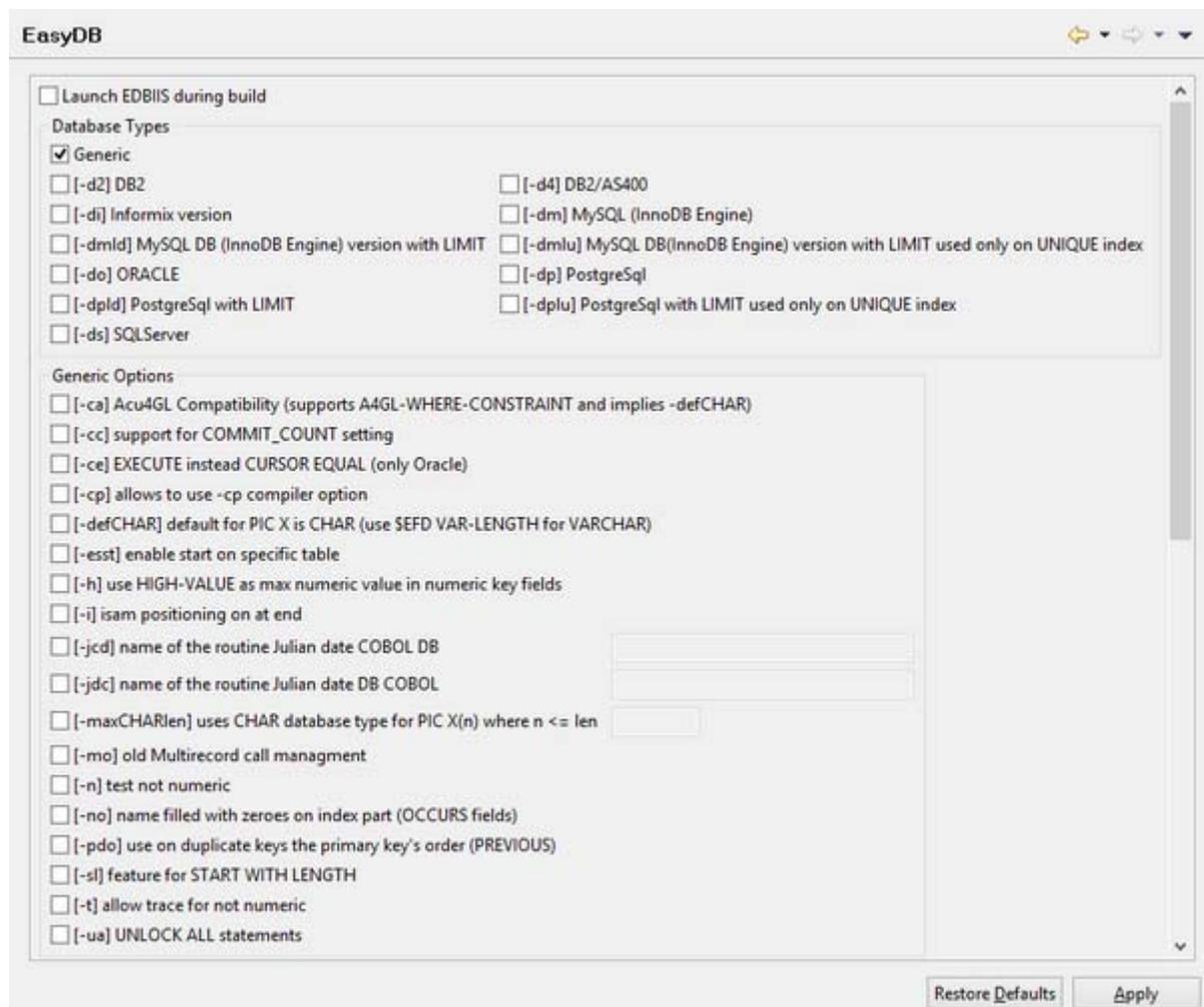
PreProcessors Names is the list of preprocessors that must be invoked to precompile the source file (click on the *Add* button to add items to the list). Remaining settings are optional.



Now, each time you issue a compilation, it will be done remotely. To disable the feature and compile locally, uncheck the *Use Remote Compiler* option.

Generating Database Bridge subroutines

isCOBOL IDE can automatically generate EDBI subroutines when the project is built. If this feature was not activated during project creation, it can be done now by clicking on *Project* menu and choosing *Properties*, then *isCOBOL Settings* and *Database Bridge*.



Check the option *Launch EDBIIS during build* to activate the feature.

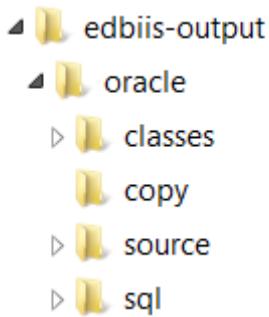
Also check the proper options depending on the database you want to interface.

EDBIIS can be launched also manually at any time by clicking on *Project* menu and selecting *Launch Edbiis*.

Database Bridge must have been properly configured in order to work. See [Binding Database Bridge to the IDE](#) for details.

Note: since EDBIIS process EFD dictionaries in order to produce subroutines, ensure that *-efd* compiler option is set for the project. See [Compiling](#) for details on how to set compiler options.

If the subroutine generation is successful, a new subfolder appears in the project tree.



edbiis-output contains a folder for each database you checked in the preferences (in the above picture, Oracle is shown). The database folder contains four folders:

- *classes*: compiled classes of subroutines
- *copy*: subroutine copyfiles
- *source*: subroutine source code
- *sql*: table schema generated by EDBIIS (documentary only)

If the subroutine generation fails, check the Error Log view for errors in the EDBIIS setup and the Console view for errors in the EDBIIS activity.

External Preprocessors

An External Preprocessor is an executable file that takes two parameters as input:

```
<input-file>, <output-file>
```

The preprocessor analyzes the input file named *<input-file>* and produces an output file named *<output-file>*.

To integrate an External Preprocessor in the isCOBOL IDE Builder, the user has to select the project in the isCOBOL Explorer View, right-click on it and select the menu-item 'Properties'. A dialog will be opened.

Expand 'isCOBOL Settings' and select the item 'Compile/Runtime options' in the list on the left of the dialog and check the option 'Use External Preprocessor'. The 'Configure' button will become enabled. Click on it to open the 'Preprocessor Configuration' dialog. In this dialog the user can select the executable program from his file system, clicking on the 'Browse' button.

The user can also set any necessary environment variables. The user can add, remove or edit these variables using the 'New', 'Edit' and 'Remove' buttons on the right side of the dialog.

The user is able to configure a particular Preprocessor for a single file inside his project's source folder. To do this, right-click on the file and select 'Properties -> Compiler options'.

When the isCOBOL IDE Builder is configured to integrate an external Preprocessor, it executes the following steps for each file inside the project's source folder that needs to be compiled:

1. As explained above, the isCOBOL IDE Builder assumes that the program set as Preprocessor takes two parameters in as input: the name of the input file and the name of the output file. The Builder checks whether or not a Preprocessor has been set for the specified file for the project and executes it passing

the name of the source file as the input file name, and the name generated as follows as the output file name:

- a. <project-location>/translated/<input-file-name-without-extension>.cbl
 - b. The extension of the original file is replaced with the 'cbl' extension.
 - c. The input file name is a path relative to the project's source folder.
 - d. The output file name is an absolute path.
 - e. In this way, the Preprocessor program creates the output file inside a standard project's folder named 'translated'. The 'translated' folder is created automatically if it does not exist.
2. The Builder checks whether or not the exit code of the Preprocessor program is equal to 0 (OK) and whether or not the output file has been created.
 3. The Builder compiles the output file in the 'translated' folder and puts the output of compilation (.class and|or .java files) into the project's output folder.

The working directory of the executed Preprocessor program is the project's source folder.

The standard output/error streams of the Preprocessor program are redirected to the Eclipse 'Console' view.

If the program does not exist or cannot be executed, the Builder will add an entry to the 'Error Log' or 'Problems' view.

The compiler errors and their line numbers are related to the output file generated by the Preprocessor program.

The user is able to directly compile only the files inside the project's source folder, not those inside the 'translated' folder. These files are only the output of the Preprocessor, and they should be used only for debugging. For these reasons, an error message will be shown if the user assigns the 'translated' name to the source folder in the project's properties.

When the original source file contains some EXEC statements, the isCOBOL syntax checker does not recognize them. For this reason, a number of errors may be shown when the user opens the file.

To avoid this problem, the user should check the '-noexec' compiler option in the project's properties. Setting this option, the syntax checker will skip all EXEC statements.

Pro*COBOL Example

Procob

The syntax of the 'Procob' Preprocessor is as follows:

```
procob iname=inputfile oname=outputfile [...]
```

The script needed to integrate the 'Procob' Preprocessor should be like this:

On Windows:

```
runprocob.bat:  
procob iname=%1 oname=%2
```

On UNIX/Linux

```
runprocob.sh:  
procob iname=$1 oname=$2
```

Integration of more Preprocessors:

Using a script file, it is possible to execute two or more Preprocessors. For example, if the user wants execute both the 'Procob' and 'Kixclt' Preprocessors, he should create a script like this:

On Windows:

runprocobkixclt.bat:

```
kixclt -o %2.pco %1.cl2  
procob iname=%2.pco oname=%2.cbl  
delete %2.pco
```

On UNIX/Linux

runprocobkixclt.sh:

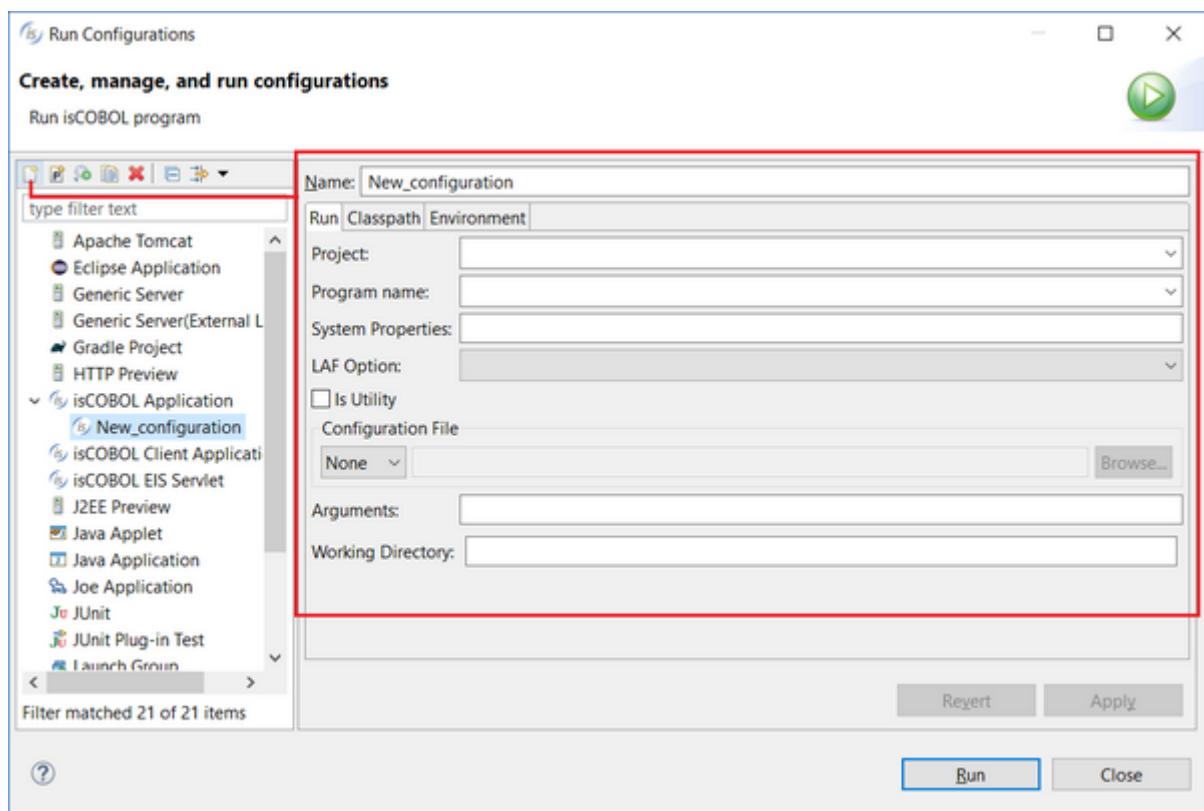
```
kixclt -o $2.pco $1.cl2  
procob iname=$2.pco oname=$2.cbl  
rm -f $2.pco
```

Run and Debug

In order to run or debug a program from inside the IDE, highlight the main source file name in the isCOBOL Explorer or click on the editor window for the main program file, and click on the *Run* menu and choose *Run As -> isCOBOL Application* or *Debug As -> isCOBOL Application*.

To set additional runtime options, create a "Run/Debug Configuration" by clicking on the *Run* menu and choosing *Run Configurations...* or *Debug Configurations...*

The following panel will appear:



After clicking on the New button, a panel will appear on the right.

Ensure that the fields are filled as follows:

Name: name of the configuration (any name can be used).

Project: name of the project in which programs are stored.

Program name: name of the main program that must start.

System properties: COBOL configuration file and other Java options (i.e. -DisCOBOL.conf=C:/etc/isCOBOL.properties -Xmx256m)

Arguments: arguments that must be passed to the program, if they exist.

Working Directory: the working directory for the runtime process. Every relative path in the program will refer to this directory.

One or more Configurations can be created to test different launches of the program (i.e. Testing with different configuration files, testing with different arguments, etc.).

Once the Configuration has been created, the program can be run or debugged.

To run:

- click on the *Run* menu
- choose *Run As*
- choose the Configuration to be used.

To debug:

- click on the *Run* menu
- choose *Debug As*
- choose the Configuration to be used.

Run and Debug features can also be reached through the proper buttons on the IDE toolbar:



The following keyboard shortcuts are also available:

CTRL+F11	Run As
F11	Debug As

Note - When the user runs or debugs a program, if there are some unsaved editors, a message box appears, asking if the user wants to save the editors before the launch. In addition, if the option Build (if required) before launching is checked in *Preferences: Run/Debug -> Launching*, a build of the project is performed.

While debugging the program, available debug options and commands are shown in the Debug View, just below the editor, and the current statement is highlighted in the Code Editor.



Note: when you create a Run or Debug configuration you can set the working directory for the runtime session. When you run the program without a configuration, the working directory is automatically set to the "output" folder of the project.

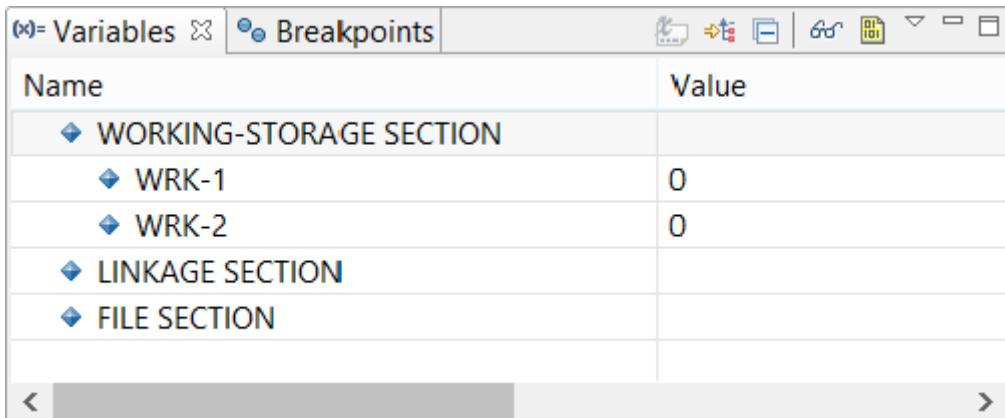
Debug Perspective

A better interface for the debug process is offered by the Debug Perspective. To switch to the Debug Perspective:

- click on the *Window* menu
- choose *Open Perspective*
- choose *Debug*

The Debug Perspective contains two views that allow you to manage Variables and Breakpoints.

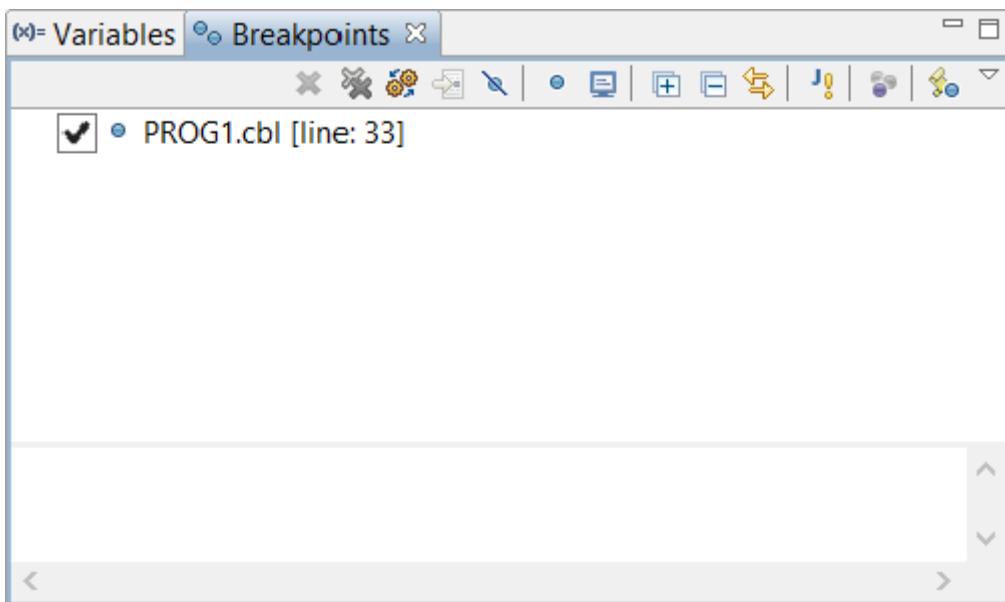
The Variables view allows the user to monitor the content of variables.



The screenshot shows the 'Breakpoints' view in the IDE. At the top, there are tabs for 'Variables' and 'Breakpoints'. The 'Breakpoints' tab is selected. Below the tabs is a toolbar with various icons. The main area displays a table with two columns: 'Name' and 'Value'. The 'Name' column lists sections: 'WORKING-STORAGE SECTION', 'LINKAGE SECTION', and 'FILE SECTION'. Under 'WORKING-STORAGE SECTION', there are entries for 'WRK-1' and 'WRK-2', both with a value of '0'. The table has scroll bars at the bottom.

Name	Value
WORKING-STORAGE SECTION	
WRK-1	0
WRK-2	0
LINKAGE SECTION	
FILE SECTION	

The Breakpoints view allows the user to toggle breakpoints and monitors.



When the debugging session terminates, the IDE switches back to the isCOBOL perspective.

Update the source code while debugging

isCOBOL IDE allows you to change the source code of a paragraph while you're debugging a program.

This feature requires that programs are compiled with the -dx option and without the -sns and -big options.

To take advantage of this feature:

1. start debugging the program by clicking on the following button:



2. identify the code that you wish to change and apply your changes.

Note: The following changes are not permitted:

- a. Add, remove or rename a graphical control
- b. Add, remove or rename a data item
- c. Add, remove or rename a paragraph or section
3. click on *Project* menu and choose *Build Project*
4. continue debugging until you reach the changed code

Note: all the changes made to a paragraph code will be active starting from the next time the paragraph is executed. If you change the paragraph code while you're debugging it, then the changes will not have effect.

Debug a remote application

isCOBOL IDE allows to debug a program that is running in a local or remote Application Server. The Application Server must have been started having `iscobol.rundebug *` set to either 1 or 2 in the configuration.

1. click on *Run* menu
2. choose *Debug Configurations....*
3. choose *Remote isCOBOL Application* from the tree



4. compile the fields as follows
 - o *Connect* page
 - a. **Name:** free text, type any name. This name will appear in the *Debug History* menu for future uses
 - b. **Host:** server name or IP address where a remote debugger is listening
 - c. **Port:** port where a remote debugger is listening
 - o *Source* page
 - a. Browse for the source files of the programs you want to debug

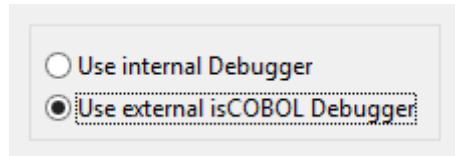
Press the *Debug* button to start debugging.

External Debugger

By default, when you debug a program, the IDE switches to the Debug perspective. If you prefer to use the standard isCOBOL Debugger, started outside of the IDE, you can configure it as follows:

1. click on *Window* in the IDE menu bar
2. choose *Preferences*
3. expand the *isCOBOL* tree
4. choose *Run/Debug*

- switch from “Use internal Debugger” to “Use external isCOBOL Debugger”



- click OK

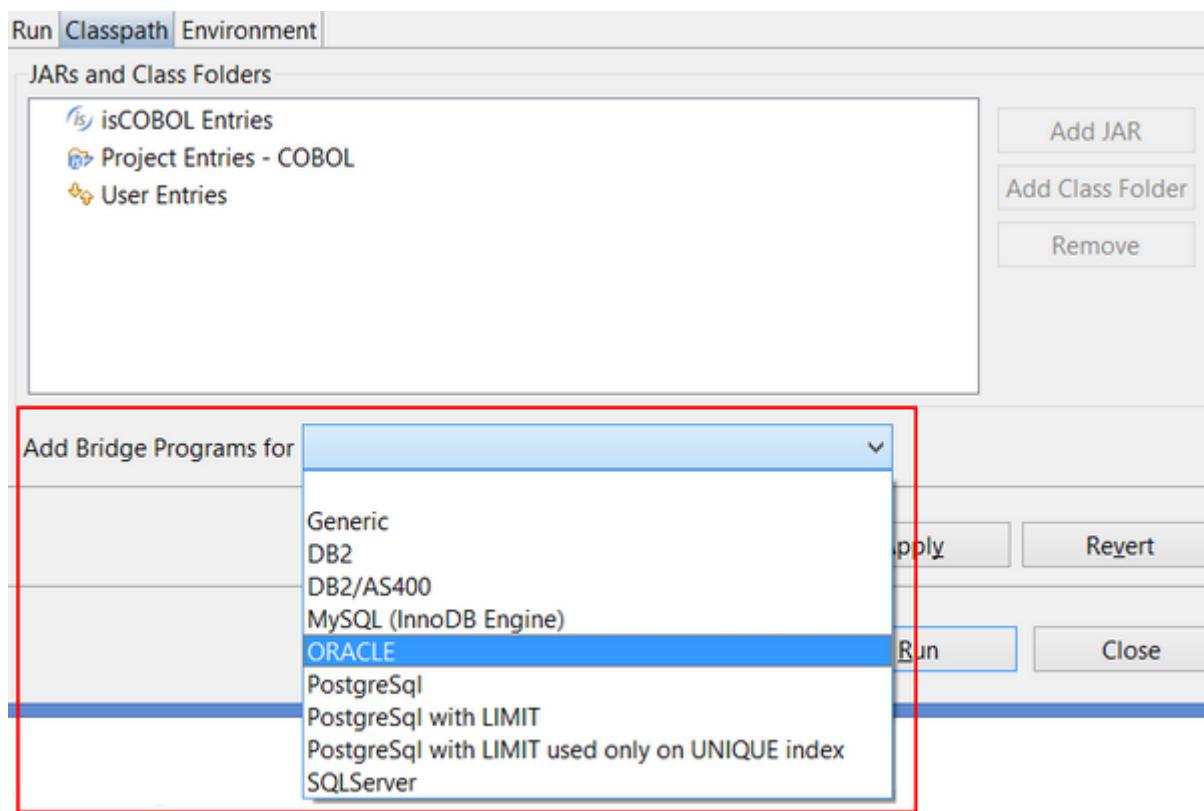
For more information about isCOBOL Debugger, please consult [Debugger](#).

Running with Database Bridge

If you plan to run your programs having Database Bridge as file index, then additional work is required.

- Ensure that EDBI subroutines have been generated in your project (see [Generating Database Bridge subroutines](#) for details).
- Add the JDBC driver library provided by your database vendor to the project Classpath. Click on *Project* menu, select *Properties*, select *Class Path* from the tree. Click on the *Add Jar* button and browse for the JDBC jar file.
- Set *iscobol.file.index=easydb* in the configuration (it can be done anywhere in the configuration, it doesn't matter if your program performs SET ENVIRONMENT before opening a file or if you prefer to edit the *iscobol.properties* file). Set *iscobol.jdbc.driver*, *iscobol.jdbc.url* and any other applicable JDBC setting for your database connection in the configuration (it can be done anywhere in the configuration, it doesn't matter if your program performs SET ENVIRONMENT before opening a file or if you prefer to edit the *iscobol.properties* file).
- Edit the Run Configuration of your program and, in the *Classpath* page, bind the proper database

routines set:



5. You're now able to run your program using Database Bridge.

Running as Client Application

isCOBOL IDE offers the ability to launch the program in a thin client environment.

Server setup

1. Click on *Window* menu
2. Select *Preferences*
3. Choose *isCOBOL / Tools / Server / Class Path* from the tree
4. Click the *Add Class Folder* button and choose the output folder of your project
5. Optionally add other jars and folders that are necessary to the program (e.g. a JDBC driver library)
6. Click OK
7. Click on *isCOBOL Tools* menu
8. Select *isCOBOL Server / Start*

Client setup

1. Click on *Run* menu
2. Select *Run Configurations...*
3. Select *isCOBOL Client Application* from the tree and click on the *New* button at the top of the tree
4. Compile the fields as follow:

- o **Name:** free text, this is the name that will appear in the Run As list once we save modifications
- o **Arguments:** name of the program in the project output folder that you want to run. Be sure to type an upper-case name without .class extension. The name of the program might be followed by command-line arguments you want to pass to the program.

Note: you can use the same panel to create launch commands that point to existing isCOBOL Server processes listening in the network rather than connecting to the internal isCOBOL Server started by the IDE. In this case you have to compile also the other fields.

5. Click on *Run* button

Running as Web Application

isCOBOL IDE also offers the ability to launch the program as a web application. This feature is useful if you're developing programs that you plan to install into a servlet container (like Tomcat) to work as web applications.

To run the program as a web application:

1. Click on *Run* menu.
2. Select *Run As*.
3. Select *isCOBOL EIS WD2*.

To debug the program as a web application:

1. Click on *Run* menu.
2. Select *Debug As*.
3. Select *isCOBOL EIS WD2*.

CSS files

isCOBOL IDE loads automatically the 'iscobol.css' file embedded in the wd2 plugin. It is not necessary to copy it in the isCOBOL project. If the user copies manually the 'iscobol.css' file under the 'resources/css' folder of the isCOBOL project, that file will be used.

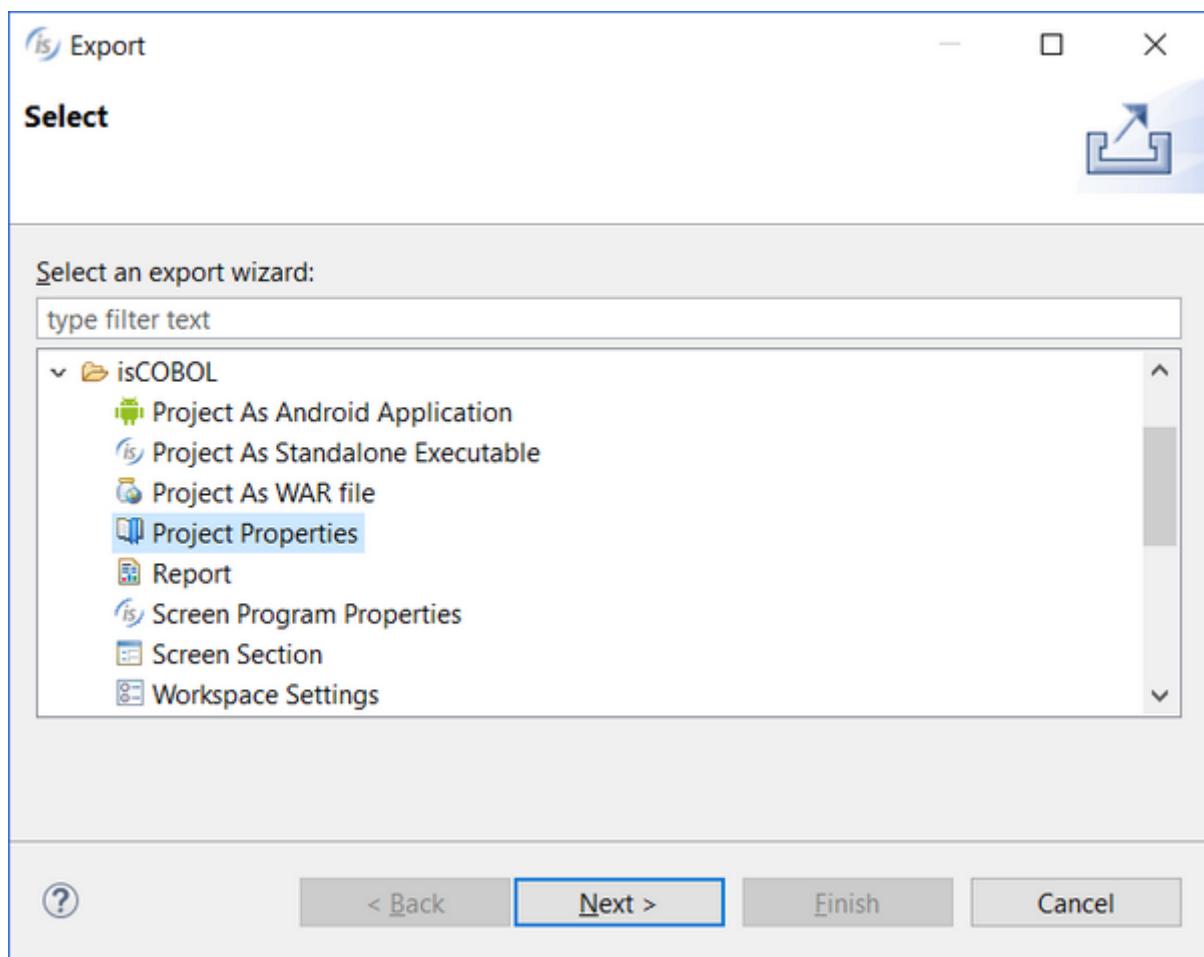
The css file pointed by [iscobol.wd2.additional_stylesheet](#), if present, must be placed in the *resources/css* folder of the project.

Import / Export Project Properties

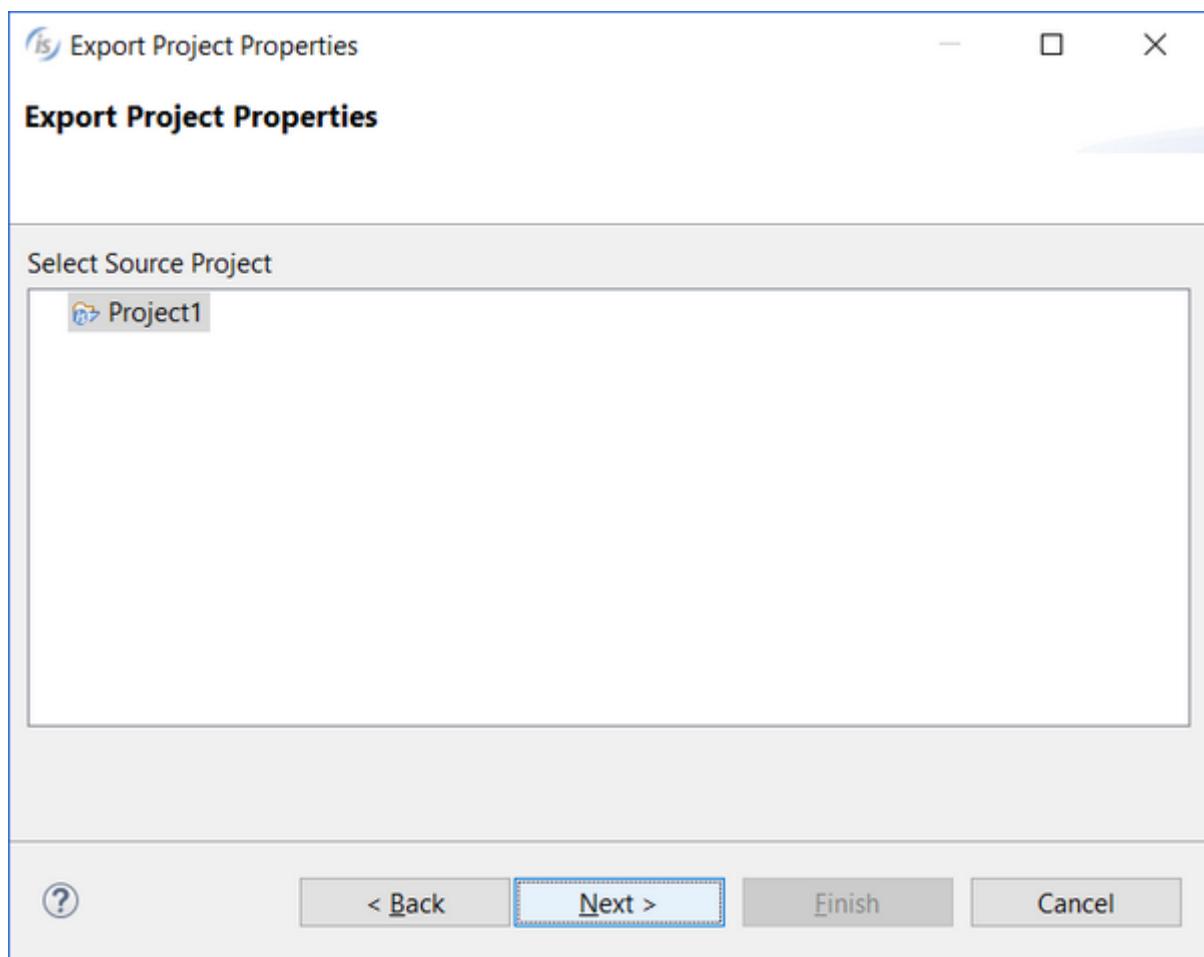
isCOBOL IDE exports the current project settings to share them with other projects.

To export project settings:

1. Right click on project name in the isCOBOL Explorer area.
2. Choose *Export* from the pop-up menu.
3. Choose *isCOBOL / Project Properties* from the tree.



The next panel will prompt for a file name. You can choose any name and extension. The IDE produces an xml output.



To import project options, the steps are very similar:

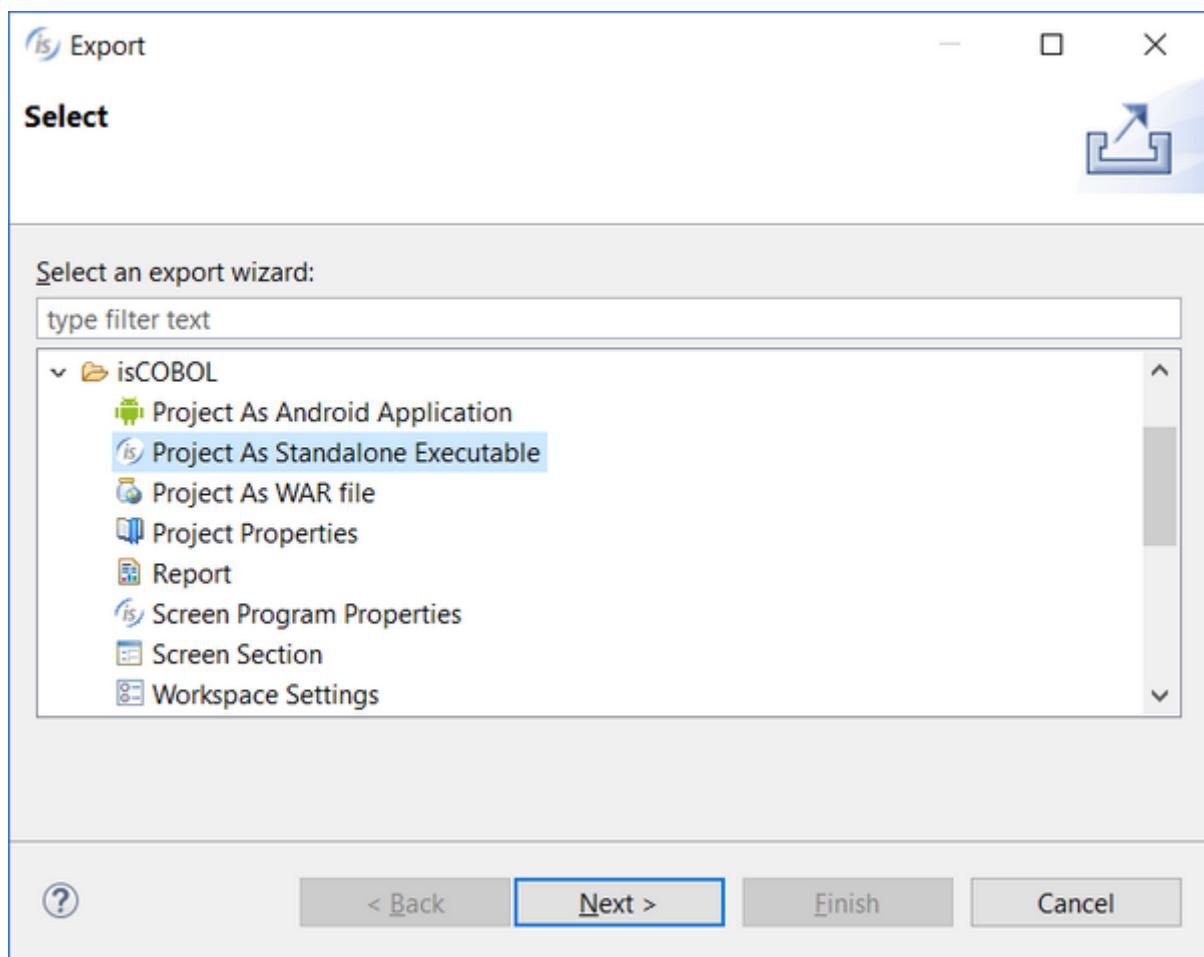
1. Choose *Import* from the pop-up menu.
2. Choose *isCOBOL / Project Properties* from the tree.
3. Choose the disk file containing the project settings. The file must have been created with the export procedure described above.

Deployment facility (JAR)

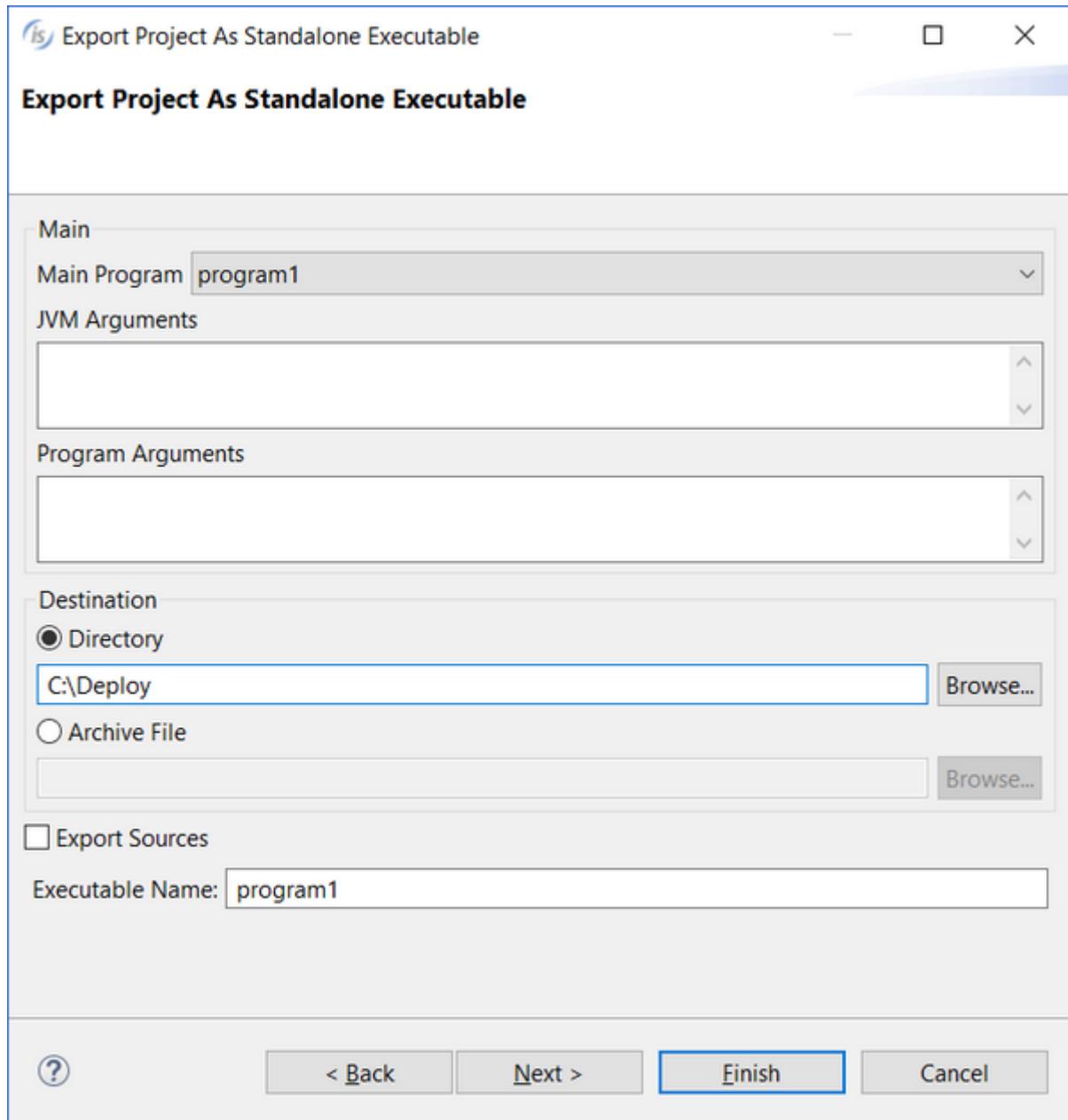
isCOBOL IDE allows you to build a stand-alone package of your project in order to facilitate the deployment.

This is useful to deploy desktop applications.

1. Right click on project name in the isCOBOL Explorer area.
2. Choose *Export* from the pop-up menu.
3. Choose *isCOBOL / Project As Standalone Executable* from the tree.



4. Click Next.
5. Select the desired project from the list.
6. Click Next.
7. Fill the fields with the required information.



- Main Program* - select the main program from the ones in the project.
 - JVM Arguments* - optionally specify arguments for the JVM (e.g. -Xmx option).
 - Program Arguments* - optionally specify command-line parameters for the COBOL program.
 - Destination* - store resulting files in a folder or in a tar archive.
 - Export Sources* - optionally include source files in the package.
 - Executable Name* - name of the jar that will allow you to start the COBOL application (the jar extension is automatically applied, do not specify it).
8. If the project needs external jar files, click *Next* and choose to export them as well before clicking *Finish*. Otherwise click *Finish*.

The package is generated in the folder (or in the archive) specified by *Destination* (see point 7). The package consists of the following folder structure:

- o *output* - contains COBOL program class files.
- o *resources* - contains the configuration file and other resources for the project.
- o *libs* - contains necessary jar libraries such as the isCOBOL runtime and third party libraries (see point 8).
- o *source* - contains the source files (only if you checked Export Sources in point 7).
- o *bin* - contains the main jar that allows you to start the COBOL application (see *Executable Name* in point 7).

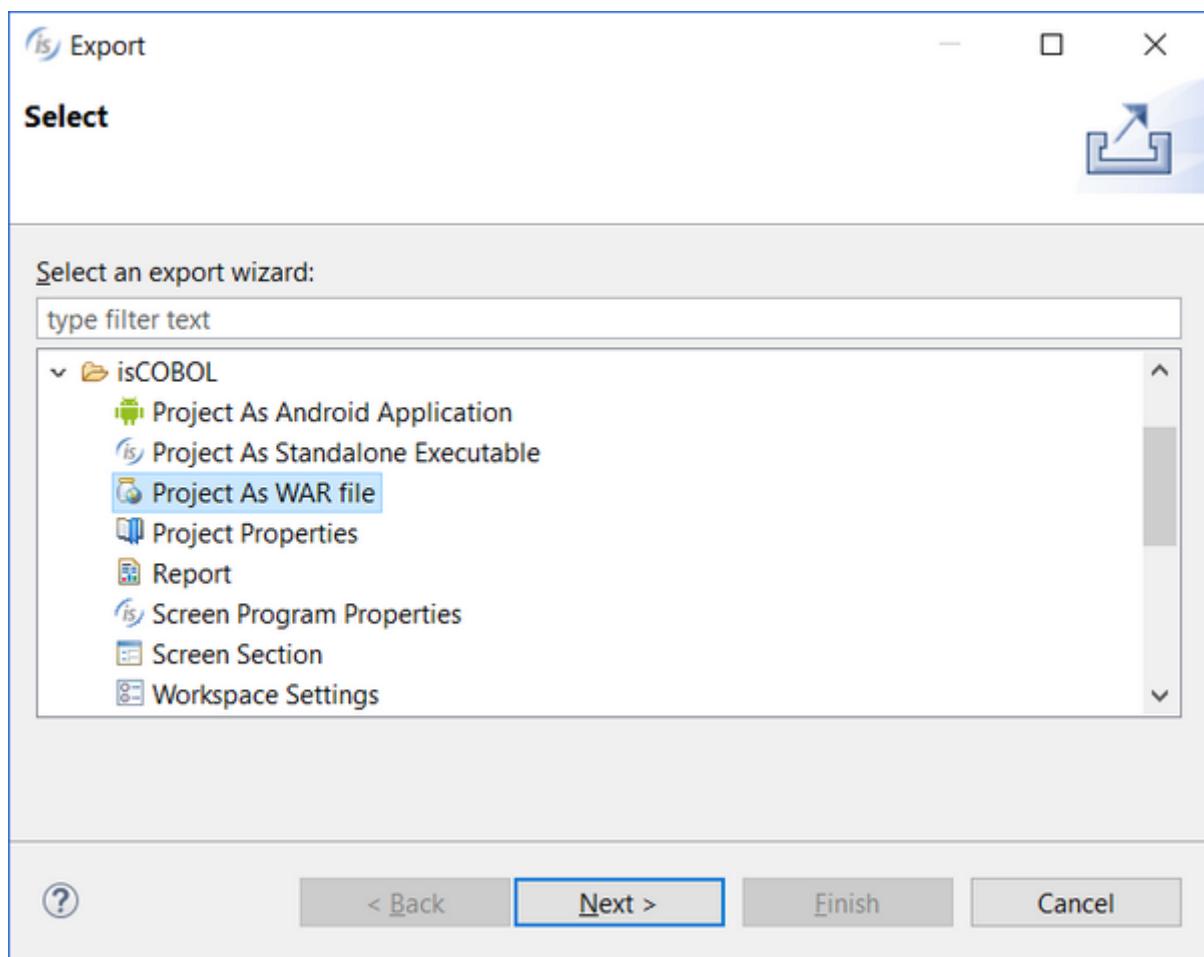
If jar files are associated with the JVM in the system, then you can start the application by double clicking on *Executable Name.jar*. Otherwise, change to that directory and run the command: `java -jar Executable Name.jar`.

Deployment facility (WAR)

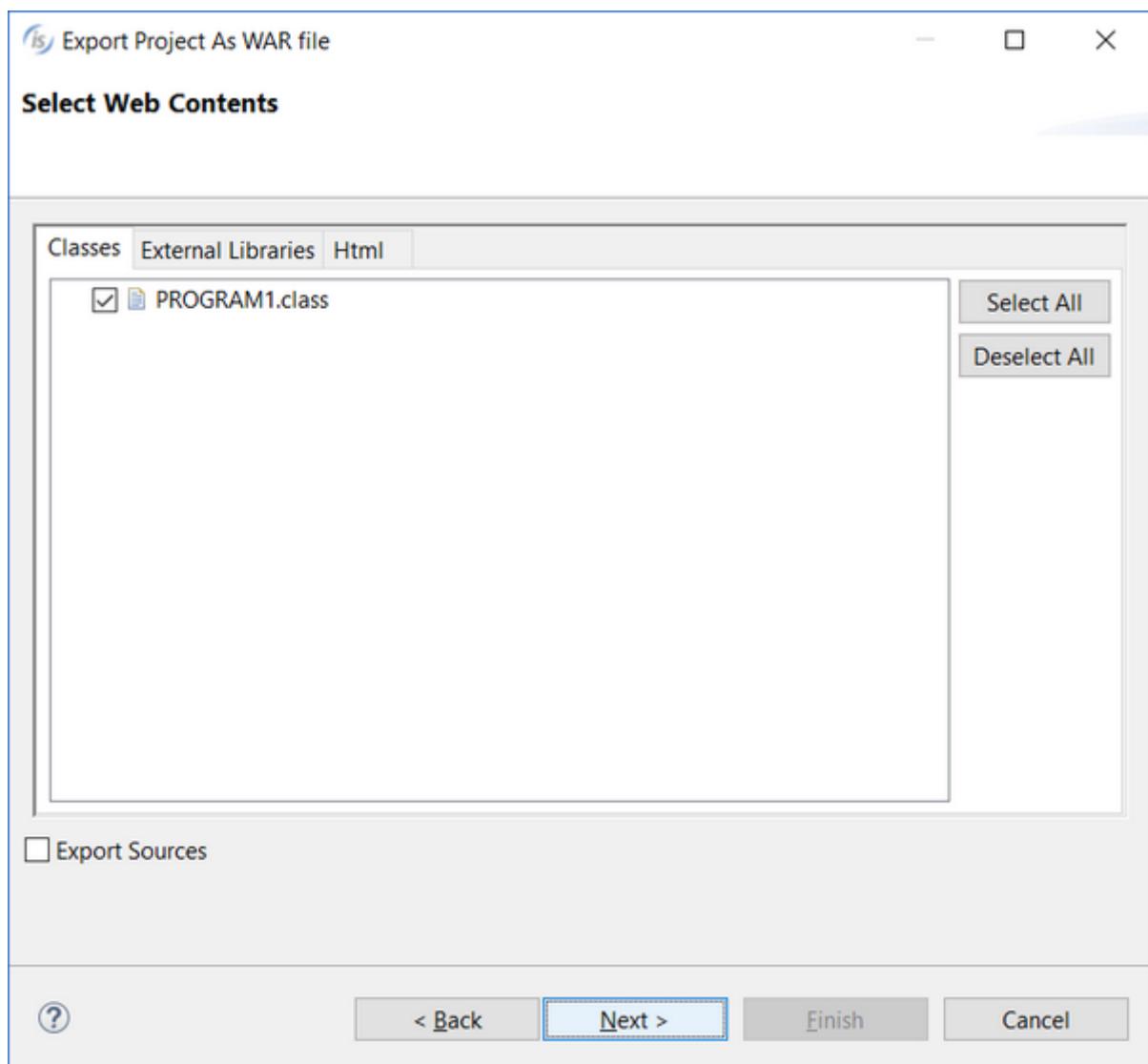
isCOBOL IDE allows you to build a WAR library of your project in order to facilitate the deployment.

This is useful to deploy web applications, either standard HTML applications or WD2 applications.

1. Right click on project name in the isCOBOL Explorer area.
2. Choose *Export* from the pop-up menu.
3. Choose *isCOBOL / Project As WAR File* from the tree.

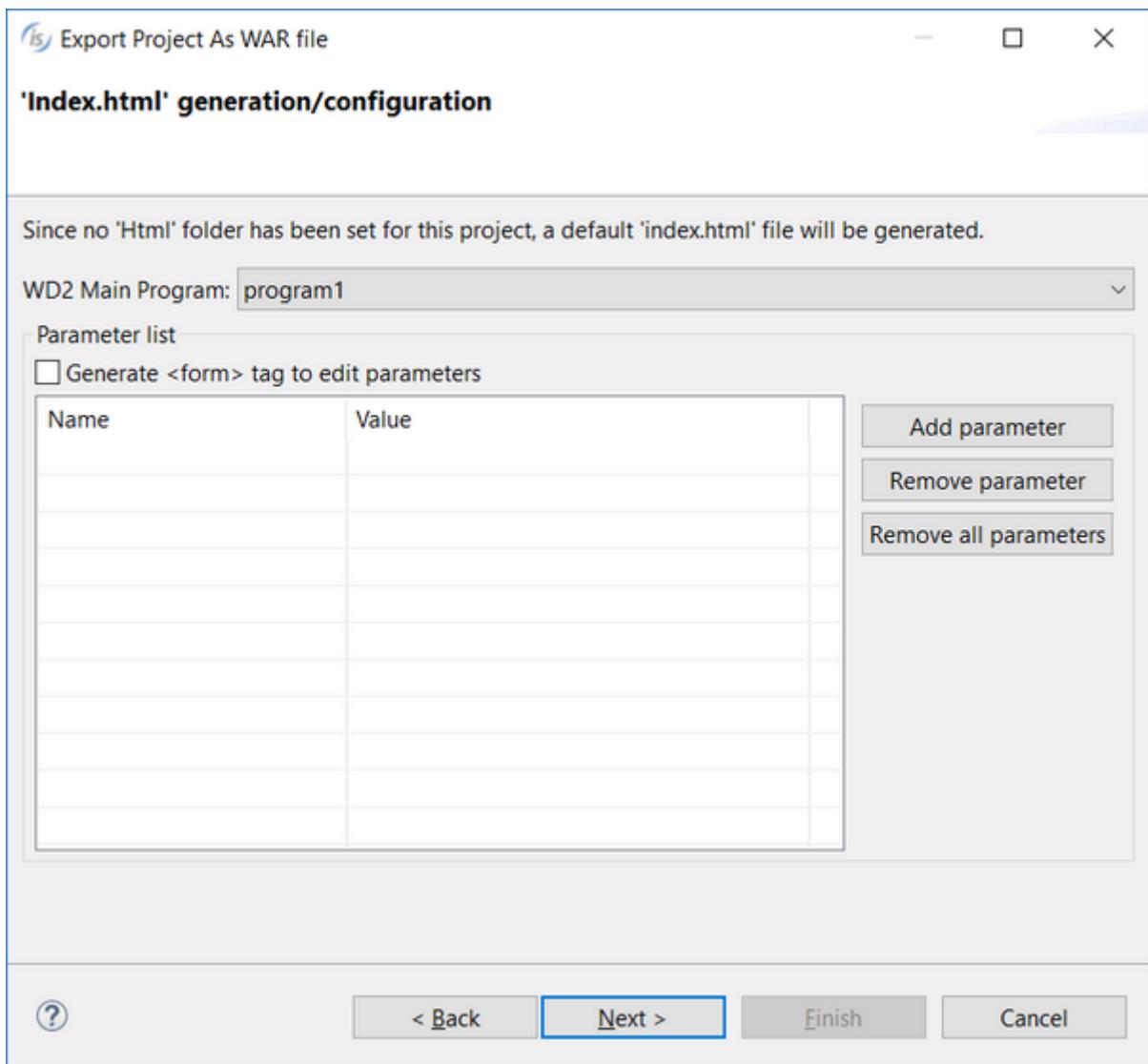


4. Click Next.
5. Select the desired project from the list and choose between *Export as Standard Html Application* or *Export as WD2 Application*.
6. Click Next.
7. In the next page you're asked to choose which programs and which external libraries from the project Classpath should be included in the WAR. Among external libraries custom libraries in the project Classpath are automatically checked, but only the essential runtime items are checked; if you need more runtime libraries for specific features (e.g. the iText library to print PDFs), then you have to check them before proceeding.
If you're exporting as "standard Html Application", then you're also asked to choose which HTML, JS and CSS files should be included in the WAR.



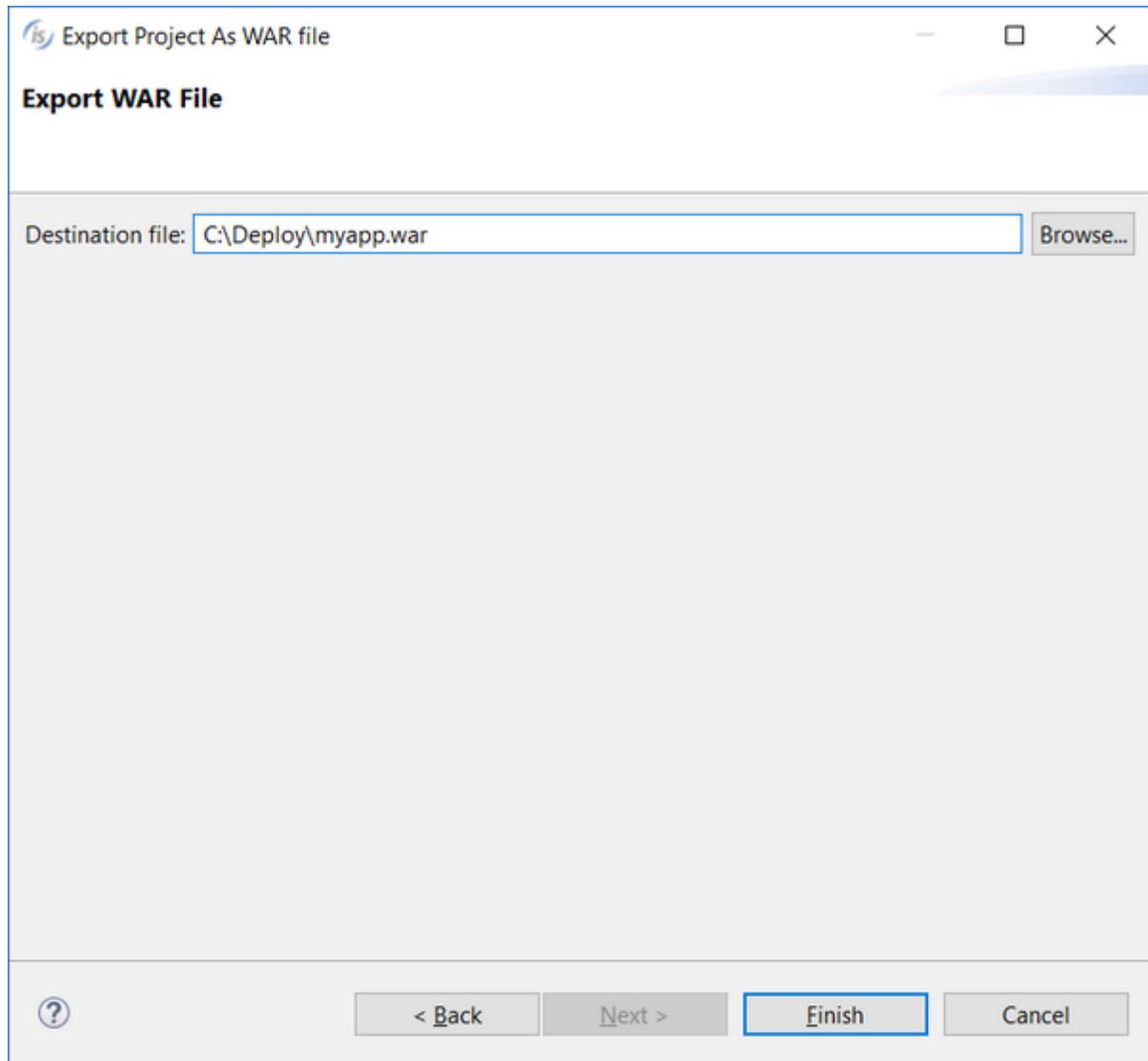
8. Click *Next*

If you're exporting as "WD2 Application" and the project doesn't include the *html* folder, then the following additional screen appears:



You have to tell which is the main program of your web application and you can make the IDE generate an HTML form that the user can use in order to set chaining parameters received by the main program. Click *Next* when you have provided the necessary information.
If the folder *html* is present in your project instead, then the content of that folder is included in the root folder of the war.

9. Choose the name and path of the WAR and click *Finish*.



The resulting WAR file is ready to be deployed in Tomcat or another servlet container.

Integrated Source Control feature

Each time a modification is saved, the IDE keeps track of it in the Local History. It's always possible to restore a previous version of a program or other project resources from the Local History.

In order to consult the history of a given resource

1. right click on the project resource
2. choose *Team > Show Local History* from the pop-up menu
3. Click on one of the entries in the History View in order to load the specific version of the resource in the editor

In order to compare the current resource with a previous version (available only for text files)

1. right click on the project resource
2. choose *Compare With > Local History...* from the pop-up menu

In order to restore a previous version of a resource from the Local History

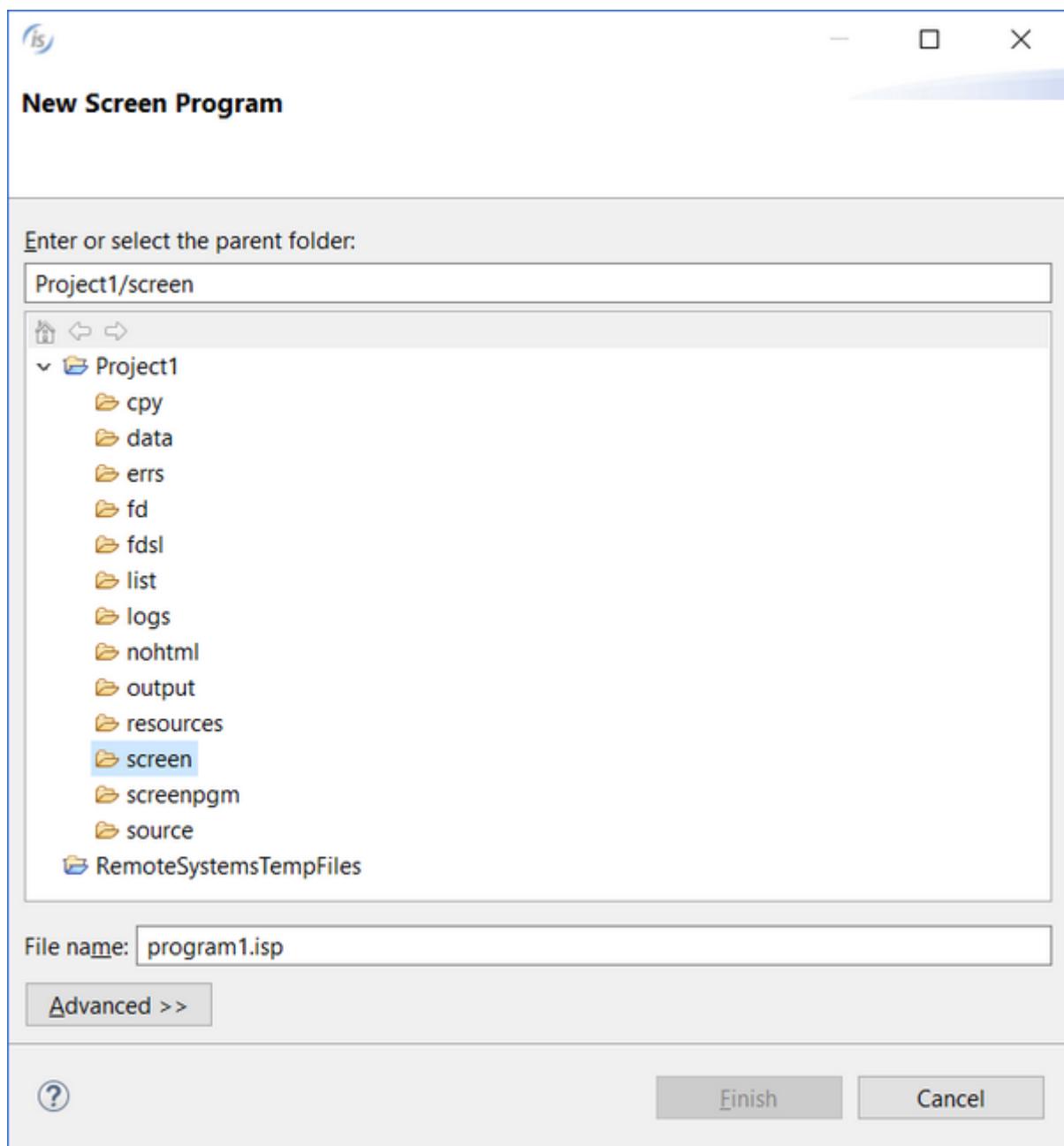
1. right click on the project resource
2. choose *Replace With*
3. choose either *Local History...* or *Previous from Local History* from the pop-up menu.

Screen Programs

Screen Programs have a program structure file and appear in the *Structural view* of the Project. This type of program contains IDE-generated code.

Creating a new Screen Program

To create a new Screen Program, right click on the project name in the isCOBOL Explorer area and choose *New / Screen Program* from the pop-up menu. The IDE will ask for program name. Ensure that the name terminates with the .isp extension and that it's placed in the *screen* folder of the project.

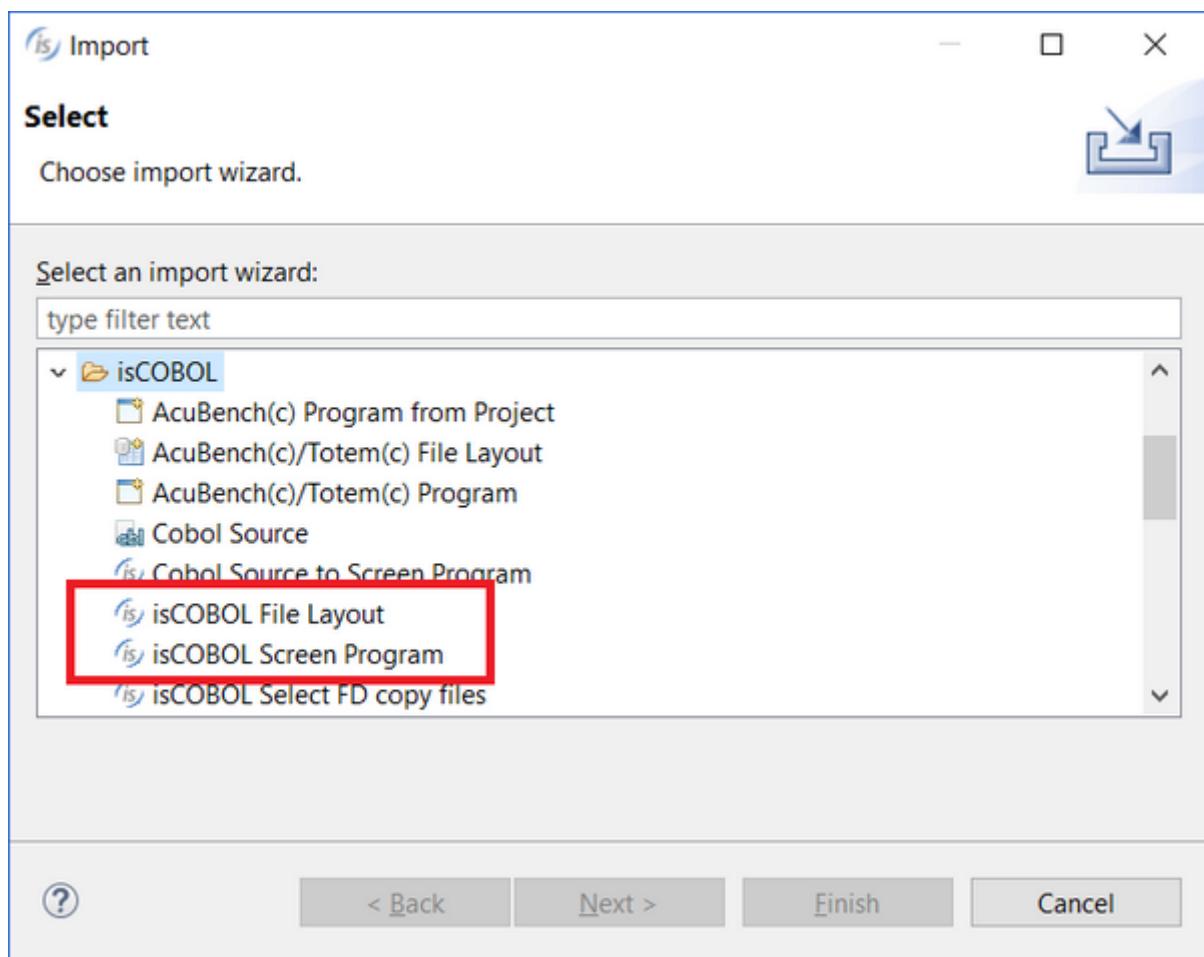


Adding existing Screen Programs to the current Project

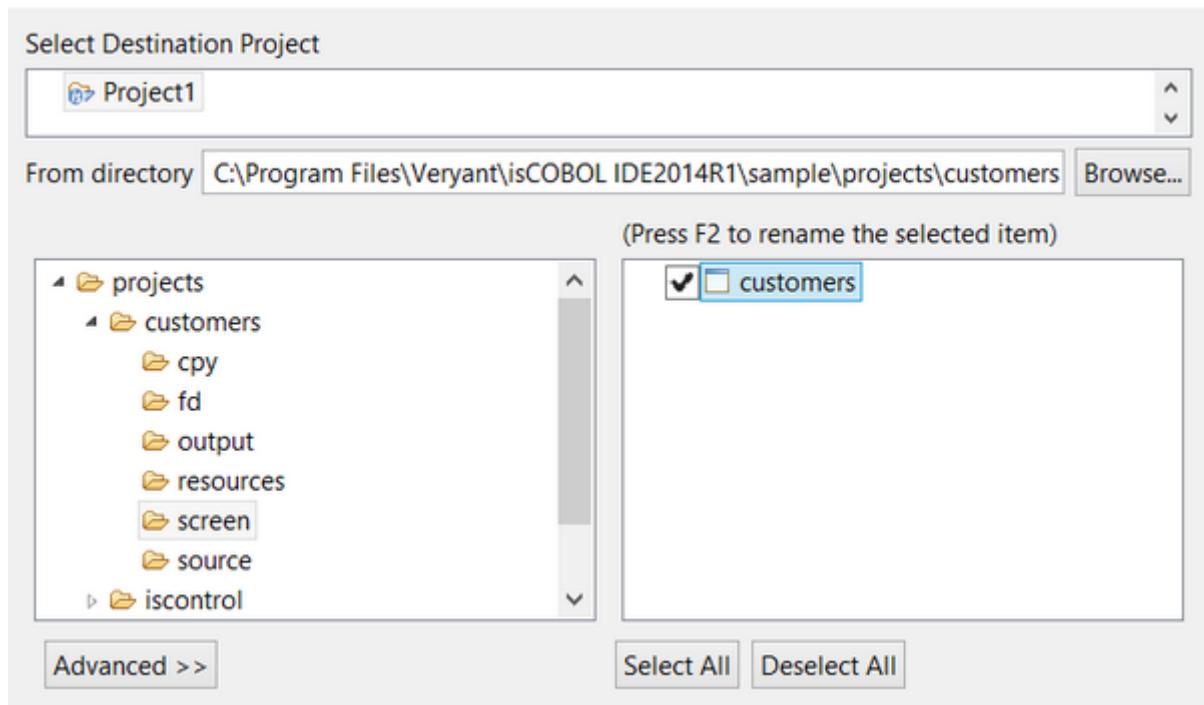
isCOBOL IDE can also import an existing Screen Program that is part of another project or workspace into the current project.

In order to add existing Screen Programs to the current Project, right click on the project name in the isCOBOL Explorer and choose Import from the pop-up menu. Then select *isCOBOL / isCOBOL Screen Program* from the tree.

Screen Programs may contain a Dataset with some file definitions. In this case, repeat the above steps, selecting *isCOBOL File Layout* from the tree, in order to import the necessary file definitions.



Use the *Browse* button to find the folder with the program or file layout that you wish to import. Select the desired item from the list on the right. Click on the *Advanced* button for advanced options.



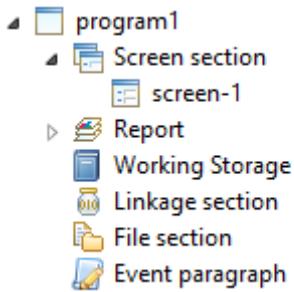
Advanced options for Screen Program are:

IDE 2013 R1 compatibility	Screen programs (*.isp) created by isCOBOL IDE 2013R1 or previous are different (and thus not compatible) than the ones created by IDE 2013R2 and later. Check this option if you need to import a isp generated by isCOBOL IDE 2013 R1 or previous. The IDE will perform the necessary conversion.
IDE 2016 R2 compatibility	Screen programs (*.isp) created by isCOBOL IDE 2016R2 or previous are different (and thus not compatible) than the ones created by IDE 2017R1 and later. Check this option if you need to import a isp generated by isCOBOL IDE 2016 R2 or previous. The IDE will perform the necessary conversion.
Create links in workspace	By default the imported resource is copied in the project folder and modifications are done on the copied resource, leaving the original resource unchanged. If you check this option instead, the resource is linked and modifications are done on the original resource.

Advanced options for File Layout are:

Create links in workspace	By default the imported resource is copied in the project folder and modifications are done on the copied resource, leaving the original resource unchanged. If you check this option instead, the resource is linked and modifications are done on the original resource.
---------------------------	--

Screen Program structure



Screen Section contains all the screens of the program. In this section it is possible to maintain the screens by drawing controls and setting their properties.

Report contains print reports. In this section it is possible to maintain the reports layout by drawing controls and setting their properties.

Working Storage contains variables, constants and copybooks that will appear in the program Working-Storage Section.

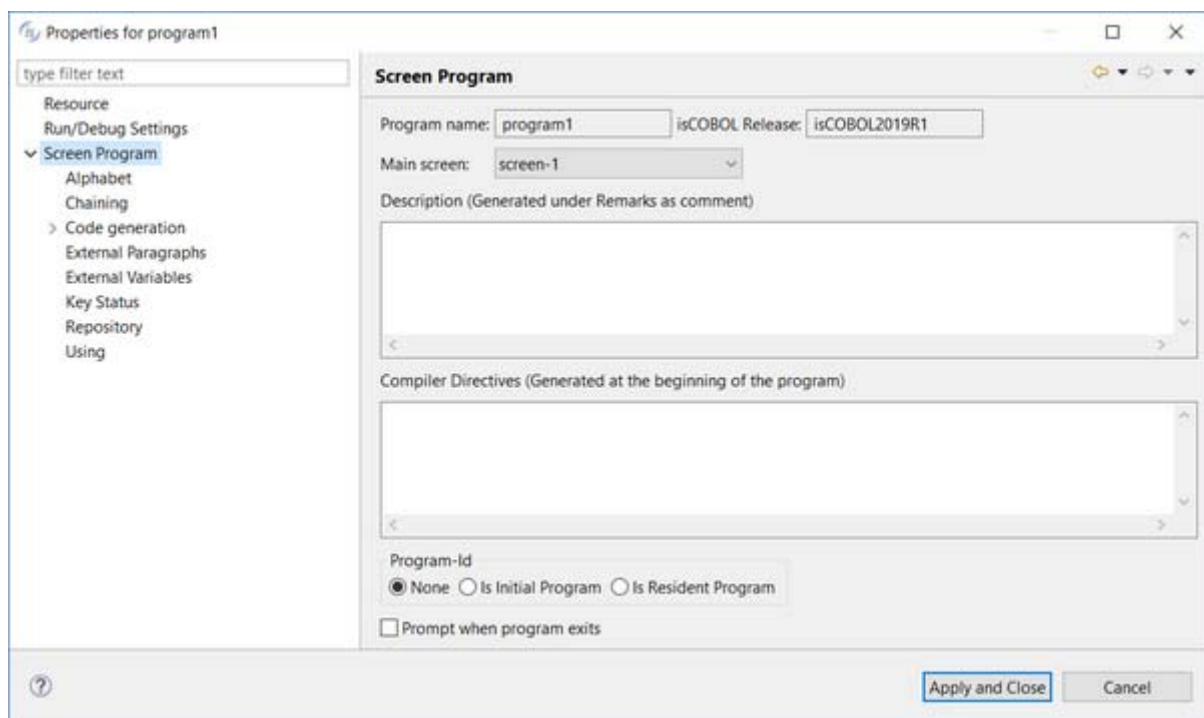
Linkage Section contains variables, constants and copybooks that will appear in the program's Linkage Section.

File Section contains FD/SL copybooks that will appear in the program. It also contains datasets that are files created through the IDE, in which FD/SL and I/O procedures will be automatically generated.

Event Paragraph contains the Procedure Division code written by the user. This code will be generated in addition to the code automatically generated by the IDE.

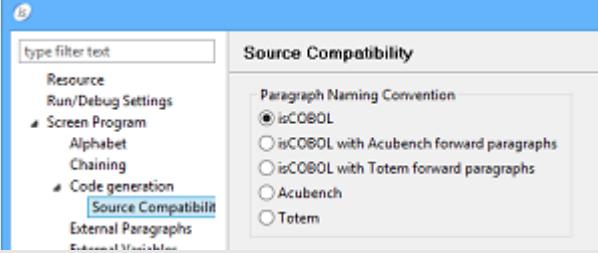
Screen Program properties

To set Screen Program properties, right click on the program name and choose "Properties" from the pop-up menu. The following panel appears.

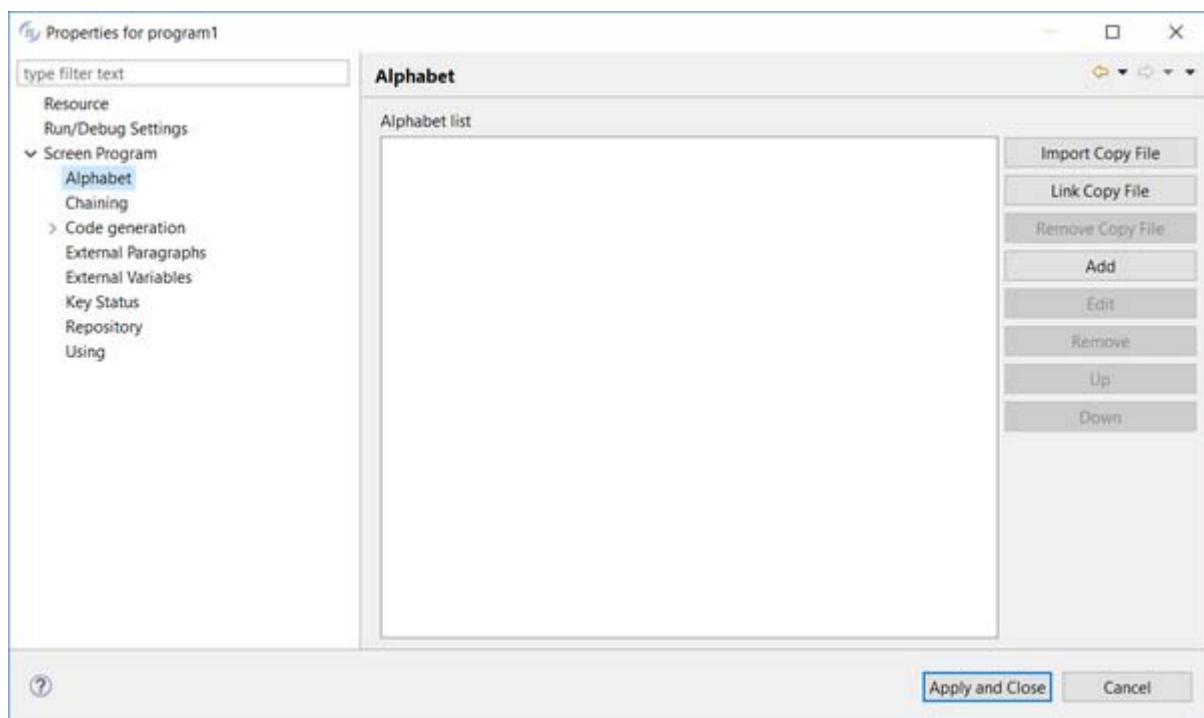


From the main screen you can set:

<i>Program name</i>	Name of the program
<i>Main screen</i>	Initial window
<i>Description</i>	Comments that are generated in the REMARKS paragraph.
<i>Program-Id</i>	Type of program (Resident, Initial, None).
<i>Prompt when program exits</i>	If set, additional code is generated to handle the program exit. The user will be asked to confirm the closing of the main window.
<i>Chaining</i>	Provides the list of data items defined in the Working-Storage Section and allows to choose which ones must be declared in the Procedure Division's CHAINING clause. The CHAINING clause is generated only if there are no Linkage Section parameters in the USING clause (see below).

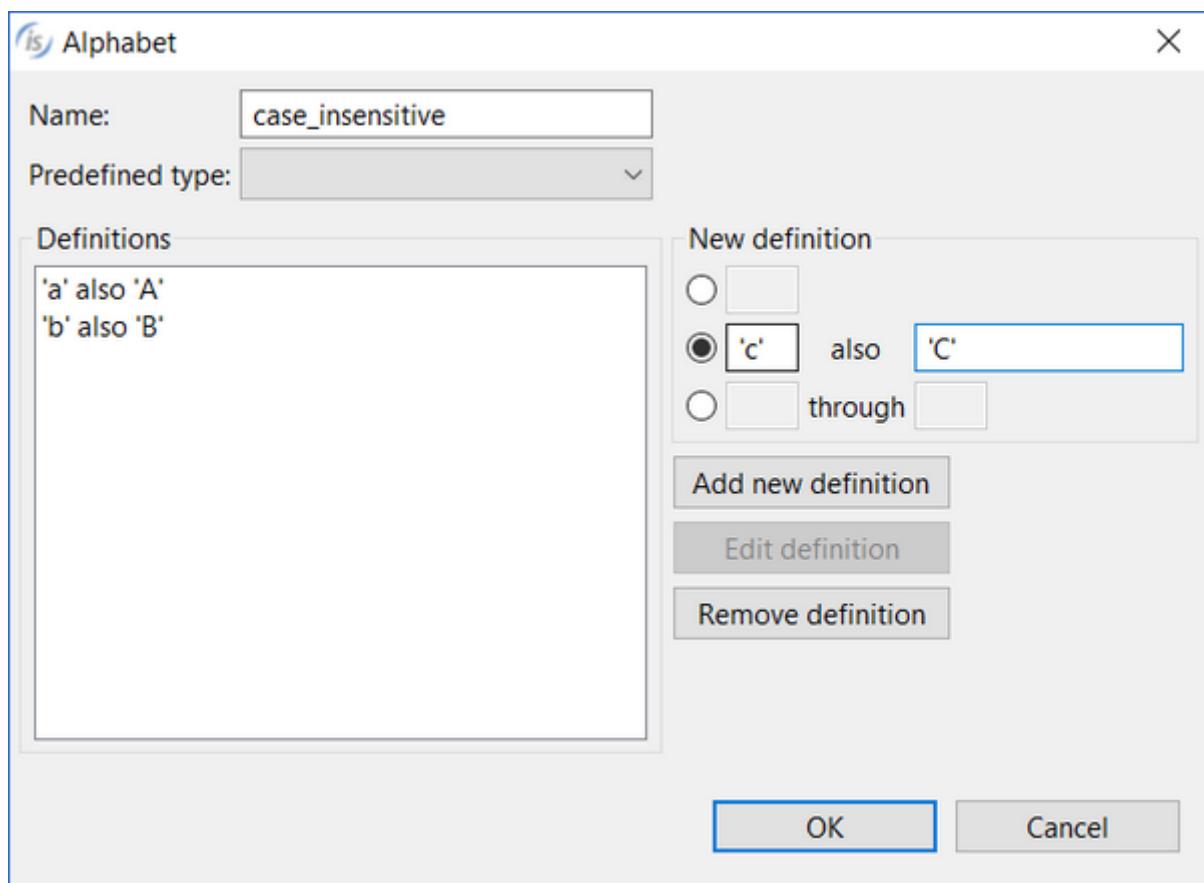
<i>Code generation</i>	<p>Specifies which parts of the source code must be generated. By default, the whole source is generated and these settings are inherited by the general Preferences.</p> <p>It's also possible to specifies the paragraph naming convention, for compatibility with other COBOLs:</p> 
<i>External Paragraphs</i>	Provides a list of the paragraphs found in the program and allows to check the ones that shouldn't be generated by the IDE. The selected paragraphs are considered external, meaning that they're externally provided by the user via code editing or copy files.
<i>External Variables</i>	Provides a list of the variables found in the program and allows to check the ones that shouldn't be generated by the IDE. The selected variables are considered external, meaning that they're externally provided by the user via code editing or copy files.
<i>Key Status</i>	Specifies the name and the picture of the crt status data-item, as well as some condition names for the most common values.
<i>Repository</i>	Specifies the program Repository .
<i>Using</i>	Provides the list of data items defined in the Linkage Section and allows to choose which ones must be declared in the Procedure Division's USING clause.

The *Alphabet* screen allows to provide alternate collating sequences to the program.



- Click on the **Add** button to add a new alphabet to the list (opens the dialog explained below)
- Click on the **Edit** button to edit the selected alphabet (opens the dialog explained below)
- Click on the **Remove** button to remove the selected alphabet from the list
- Click on the **Up** and **Down** buttons to move the selected alphabet up or down in the list controlling the order alphabets will appear in the program Special-Names

Add and *Edit* open a dialog where you can provide the alphabet name and the alphabet definitions. The screenshot below shows a work in progress case insensitive alphabet.

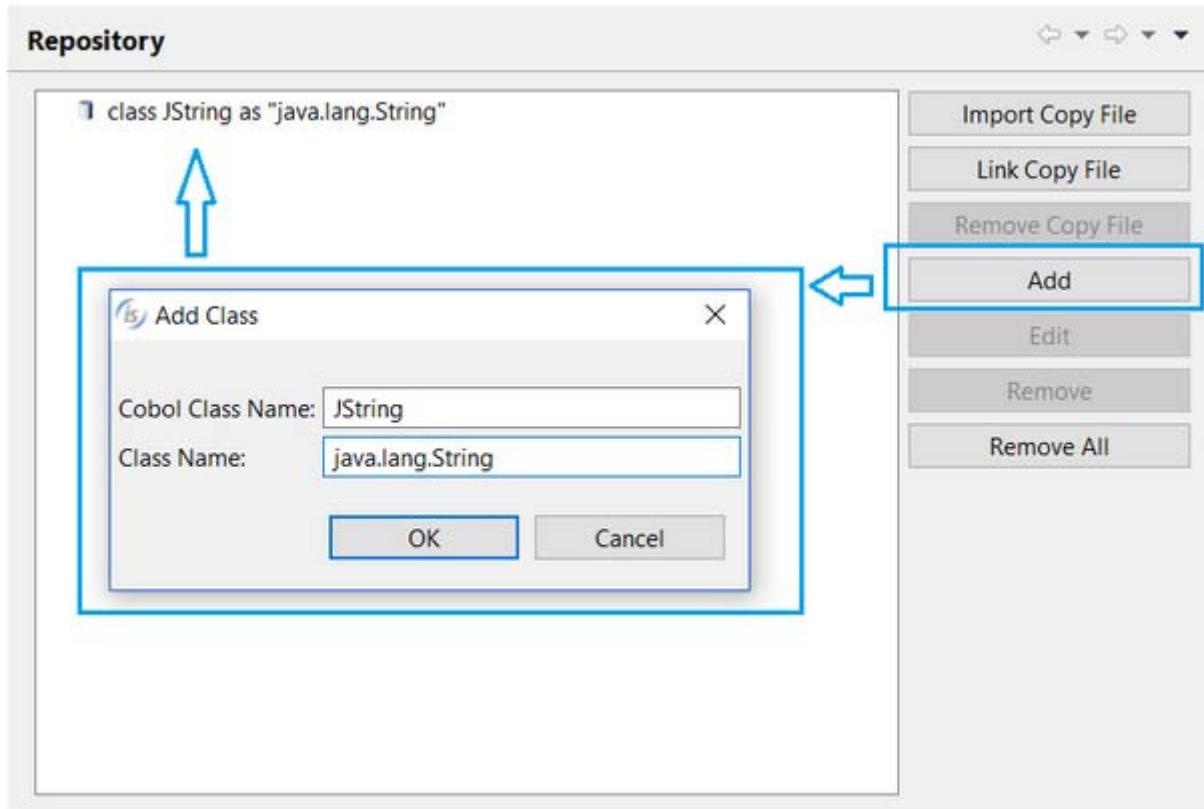


- Provide a name for your alphabet by filling the *Name:* field
- Compile *New Definition* fields and click the *Add new definition* button to add definitions
- Select a definition from the list and click *Edit definition* to modify it or *Remove definition* to remove it

Repository

If you plan to interact with Java objects in your program, you need to declare these objects. This can be done in the repository. To access the repository

- Right click on the program name in the isCOBOL Explorer.
- Select *Repository* from the list.



To add classes, click on the Add button and provide both a logical name and full class name. The class will be searched for in the project Class Path (see [Class Path](#) for more information). If not found, the IDE shows a red 'class not found' message above the fields.

It's also possible to link or import copybooks where classes are defined.

The content of this dialog is generated by the IDE in the REPOSITORY paragraph of the program allowing you to instance class methods in the COBOL code.

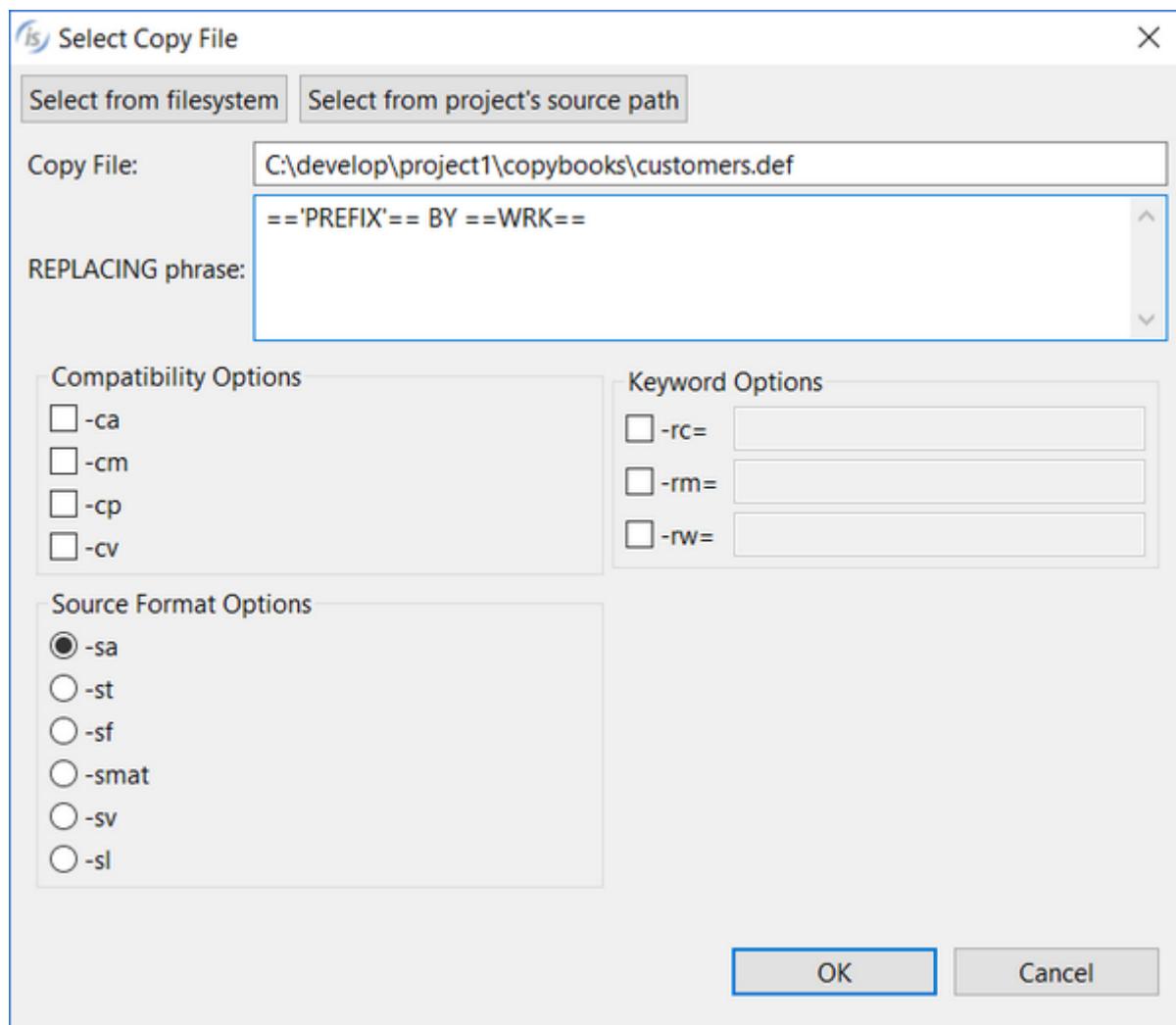
Working Storage and Linkage Section management

Add item	
01 wrk-item pic x(10)	Property
	blank when zero
	false
	className
	comment
	copy-entry
	copy-file
	copy-path
	external
	false
	identified by
	is special names
	None
	justified
	false
	level
	01
	name
	wrk-item
	occurs
	picture
x(10)	picture
	redefines
	same as
	sign is
	None
	synchronized
	false
	usage
	value

- To create a new group variable, click on the *Add item* button or choose *01-49 Level* from the *Add Item* menu.
- To create a sub-item into a group variable, click on the *Insert a sub-item* button.
- To create a stand-alone variable, choose *77 Level* from the *Add item* menu.
- To create a constant, choose *78 Level* from the *Add item* menu.
- To create a boolean constant, choose *88 Level* from the *Add item* menu.
- To create a renames, choose *66 Level* from the *Add item* menu.
- To link a copybook, choose *Link Copy File* from the *Add item* menu. Linked copy files cannot be edited from inside the IDE, they're just referenced.
- To import all the variables of a copybook, choose *Import Copy File* from the *Add item* menu. Once imported, the variables can be maintained from inside the IDE as if as they were created through it.

In the dialog shown by *Link Copy File* and *Import Copy File* it's possible to specify a **REPLACING** phrase as well as compatibility flags that the IDE should apply in order to parse the copybook content correctly.

Select from filesystem allows you to browse for a copybook in the whole file system; the full path of the copy book is used in this case. *Select from project's source path* allows you to browse for a copybook in the project folders; only the copybook name is used in this case.



- You can edit item characteristics from the table on the right. The level number can be changed only for items whose level is 01 to 49.

Drawing the Screen

To draw a control with default dimensions on the screen:

1. Click to select the graphical control you want to use. All the graphical controls are on the Controls Palette. Make sure you release the mouse button before going to step 2.
2. Click on the position you want to place the control, this is going to be the upper left corner of your graphical control. Release the mouse button to make the control appear.

To draw a control with custom dimensions on the screen:

1. Click to select the graphical control you want to use. All the graphical controls are on the Controls Palette. Make sure you release the mouse button before going to step 2.
2. Click and keep the left mouse button pressed on the position you want to place the control, this is going to be the upper left corner of your graphical control.
3. Drag the mouse to the lower right corner of your graphical control (it will feel like you are sizing the graphical control). Make sure you do not release the left mouse button in between step 2 and step 3.
4. Release the left button and see that the control has been successfully drawn.

By drawing the control you automatically set LINE, COL, LINES and SIZE properties. All other properties are set to a default value that can be configured by clicking on the *Window* menu, choosing *Preferences* and then selecting *isCOBOL / Screen Designer / Default* from the tree.

The control is automatically named with the following pattern: <screen_name>-<two-digits-control-type>-<progressive-number>. For example, when you draw the first entry-field on screen-1, it will be named "screen-1-ef-1". If you wish to assign a different prefix than the screen name to the control name, you can right click on the Screen Designer area and choose "Change Screen Prefix". When you change the prefix, all existing controls in the screen are automatically renamed, unless you check the option "Change screen's prefix only", in this case only the new controls will use the new prefix. You can always rename a single control by editing its "(name)" property.

To change the property values for one or more controls drawn on the screen:

1. Select the control(s).
 - o The control is automatically selected when drawn.
 - o To select more than one control, hold Ctrl and click on the desired controls.
2. Change the properties values in the *Properties* view. The full documentation of graphical control properties can be found at the Graphical User Interface Reference section of the isCOBOL Documentation.

Note: When you set the font property of a control to a native font (e.g. large-font) the IDE shows a preview of the control depending on the `iscobol.font` setting (e.g. `iscobol.font.large`) found in the configuration. Therefore, in order to obtain a preview that is consistent with the run time, ensure that the same font settings are used both for the IDE configuration and the Framework configuration. Changing these settings after the Screen Program has been created doesn't produce any effect since these settings are read only when the program is created.

To easily change LINE, COL, LINES and SIZE

1. Select the control by clicking on it.
2. Use arrow keys to move the control on the Screen pixel by pixel (LINE and COL are updated).
 - o Hold CTRL and use arrow keys to move the control on the Screen cell by cell (LINE and COL are updated).
 - o Hold SHIFT and use arrow keys to change the control dimensions pixel by pixel (LINES and SIZE are updated).
 - o Hold SHIFT + CTRL and use arrow keys to change the control dimensions pixel by pixel (LINES and SIZE are updated).

To easily change TITLE of controls that have this property (e.g. buttons):

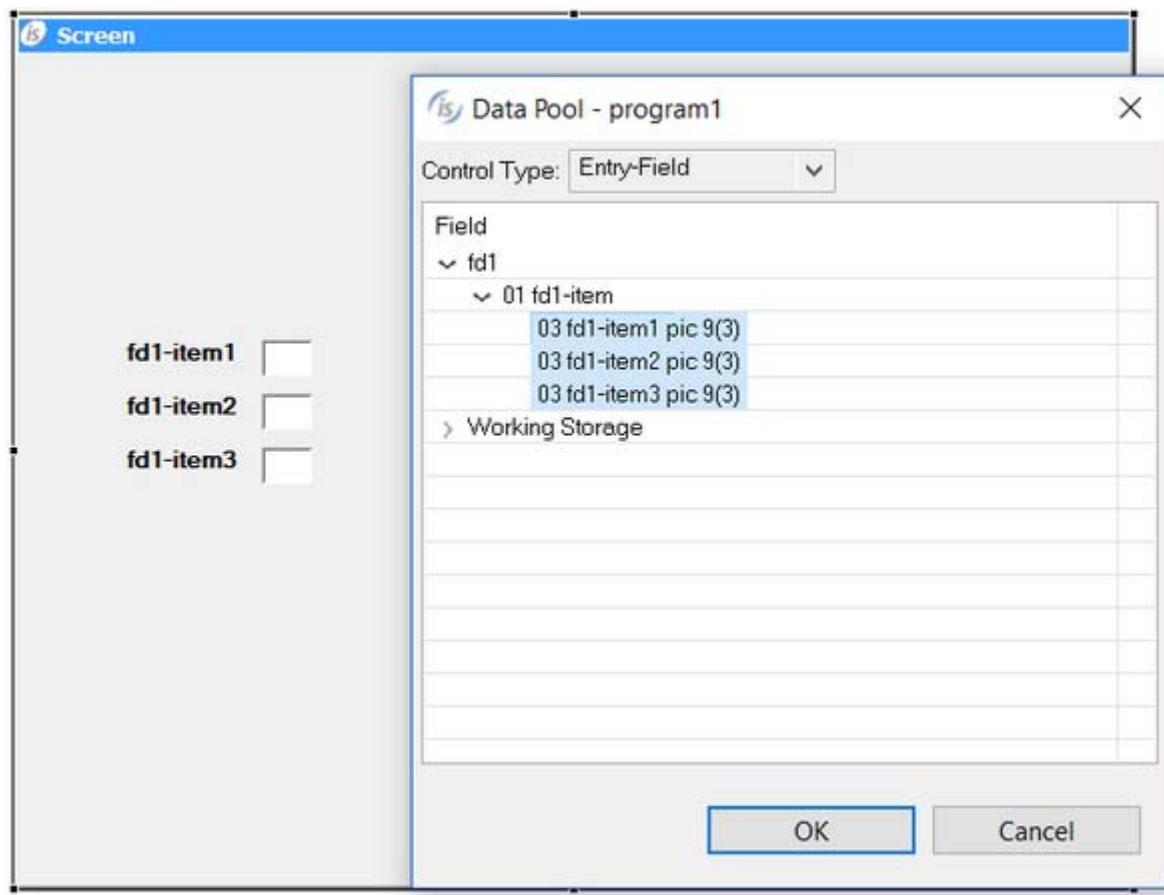
1. Select the control by clicking on it.
2. Press Enter to edit the current text or start typing to provide a new text.

If you double click on a control, the Event Paragraph editor is opened and the most common paragraph for that control is shown. It happens with all controls except for the Label. Double clicking on a Label allows to change its title instead.

The Data Pool feature

The quickest way to draw a control in the Screen Designer is by taking advantage of the Data Pool feature. The Data Pool allows you to draw a control starting from a data-item defined in the program.

Right click on the screen and select "DataPool" from the pop-up menu. The following dialog pops up.



Select the data-items that you wish to handle through graphical controls. Select the control type from the list on the top. Drag the data items on the screen with the mouse. When you release the left mouse button the desired controls appears on the screen.

The Group Item Designer

The Group Item Designer allows you to create groups of controls. By default, the IDE generates every control as a sub-item of the screen in the Screen Section of the source. The generated code looks like:

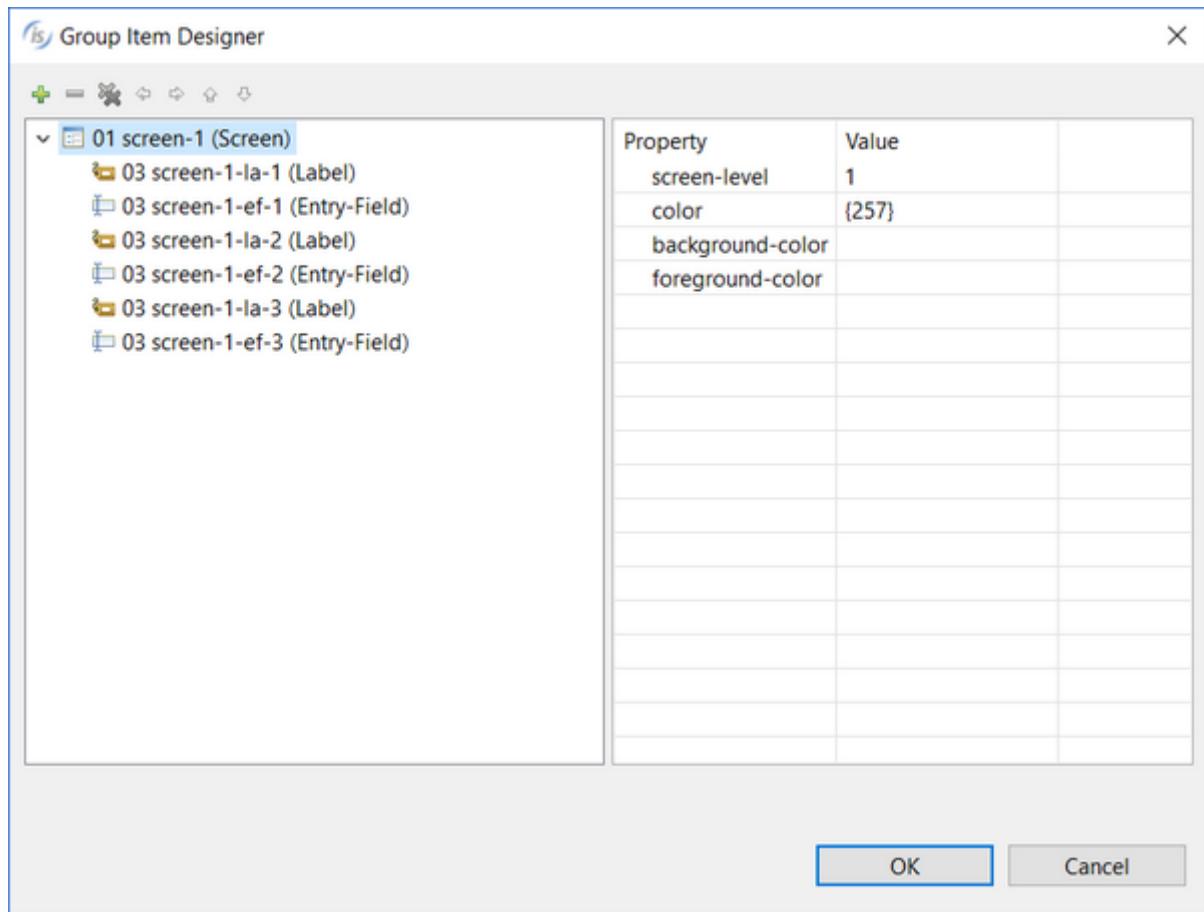
```
01 screen-1.  
 03 control-1.  
 03 control-2.  
 03 control-3.  
 03 control-4.  
 03 control-5.  
 03 control-6.
```

The Group Item Designer allows you to separate these controls into different groups. The new code may look like:

```
01 screen-1.  
  03 group-1.  
    05 control-1.  
    05 control-2.  
  03 group-2.  
    05 control-3.  
    05 control-4.  
    05 control-5.  
  03 control-6.
```

The advantage is that you can set some common properties (like visibility) or common embedded procedures for the group instead of setting them for each single control.

To take advantage of this feature, right click on the window in the Screen Designer and select "Group Item Designer" from the pop-up menu. The following dialog appears:



By default, the current Screen Section representation is shown.

- To create a new group, click on the "Add" button.
- To remove an item or a group, click on the Remove button. The item will not be generated in the program Screen Section.

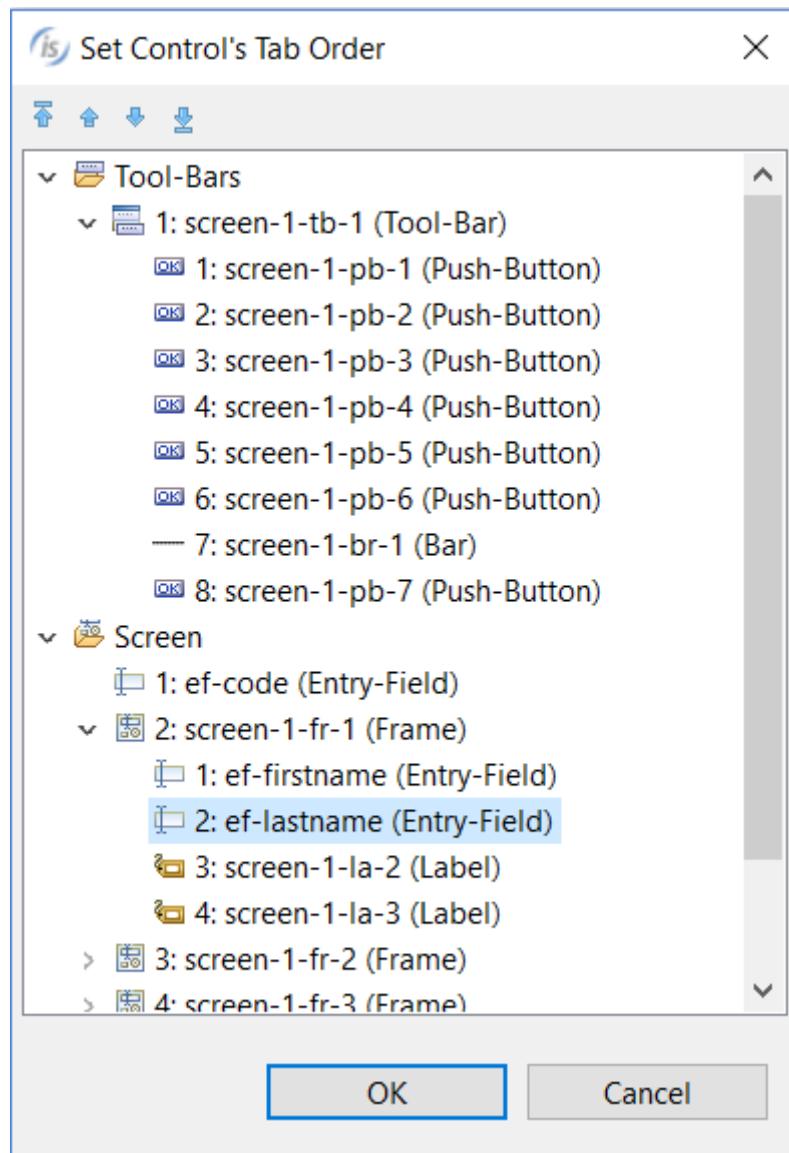
- To change the level number of a control or a group, click on the "Move Left" and "Move Right" buttons.
- To change the tab order of a control or a group, click on the "Move Up" and "Move Down" buttons.

Tab Order and ID Order

The tab order is the order in which controls are described in the program's Screen Section that matches with the order in which they get the focus when the user press the Tab button on the keyboard. This ordinal number allows to identify the control via the CONTROL-VALUE data item. Every control has a ID property. The ID is a unique number that allow to identify the control via the CONTROL-ID data item. See [SCREEN CONTROL](#) for more details about CONTROL-VALUE and CONTROL-ID data items.

To change the tab order of controls

1. right click in the screen area
2. choose *Tab Order* from the pop-up menu



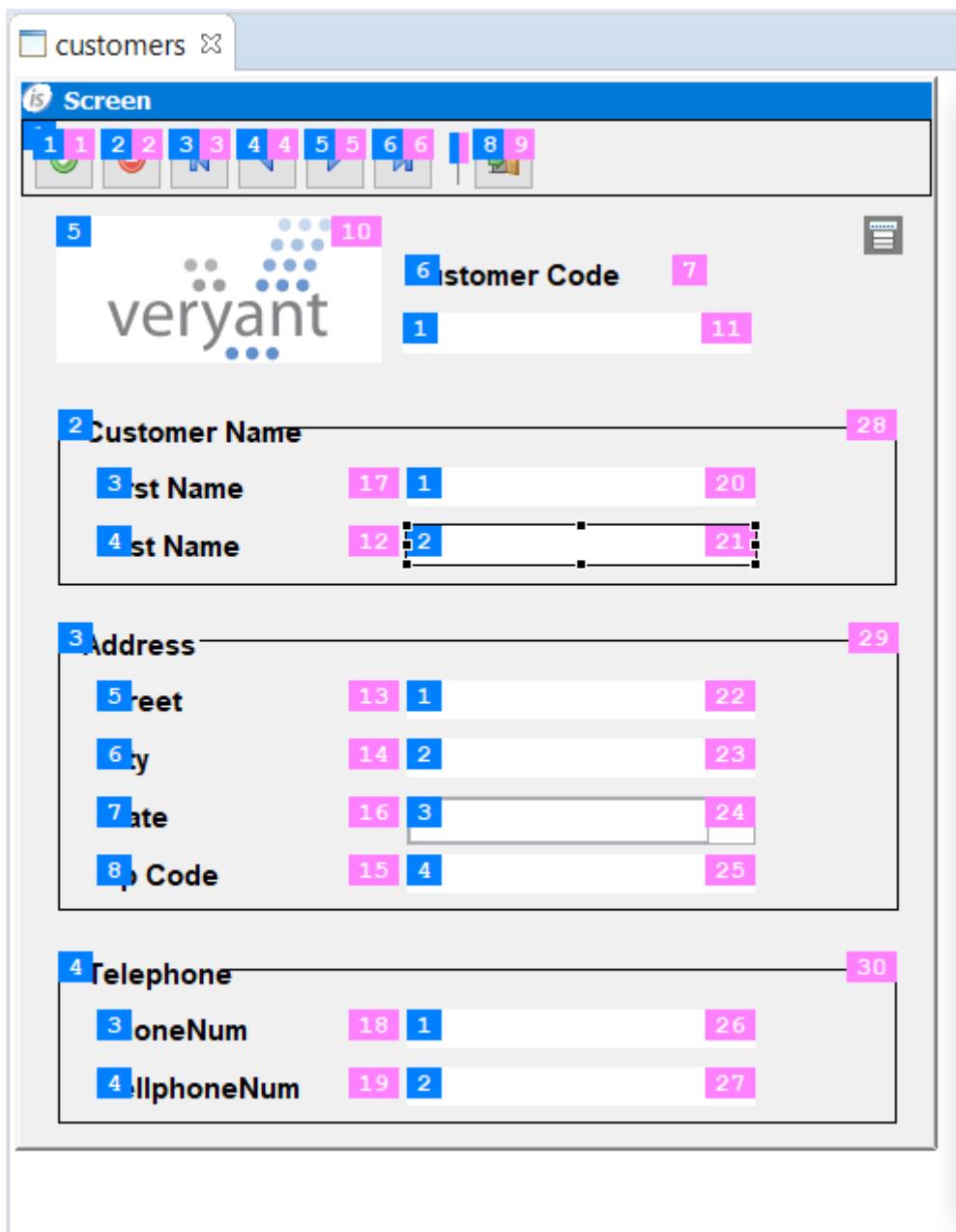
Use the buttons on top of the list or drag the items in the tree-view in order to change the tab order of the controls.

To change the ID of controls

1. select the control with a single click of the left mouse button
2. search for "id" in the *Properties* view and change its value

To have an overview of the current tab order and IDs:

1. right click in the screen area
2. select *Show Control Numbering* in the pop-up menu
3. activate the two items in this sub menu



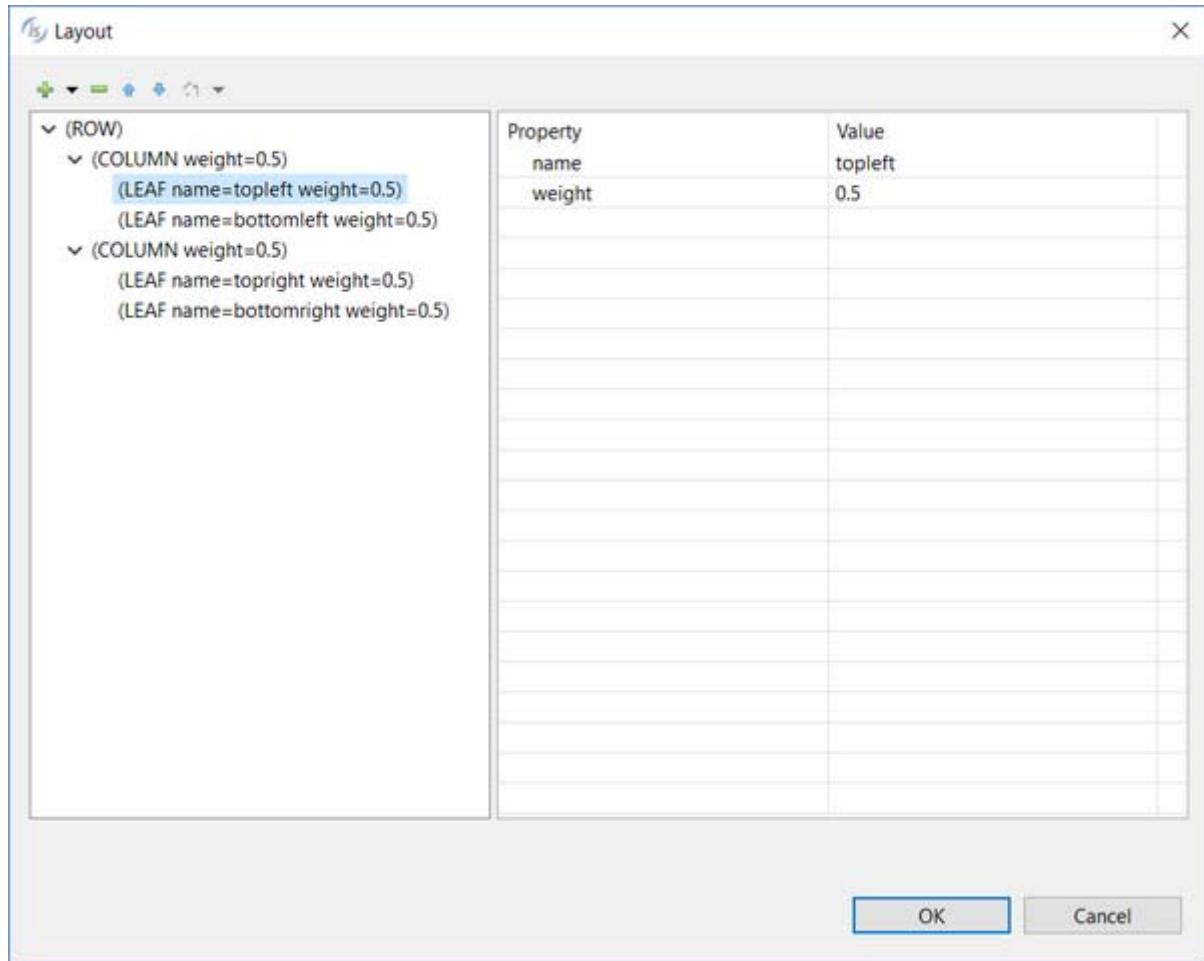
The tab order of controls is displayed on the left side, while control ID is displayed on the right side of the control.

How to create docking windows

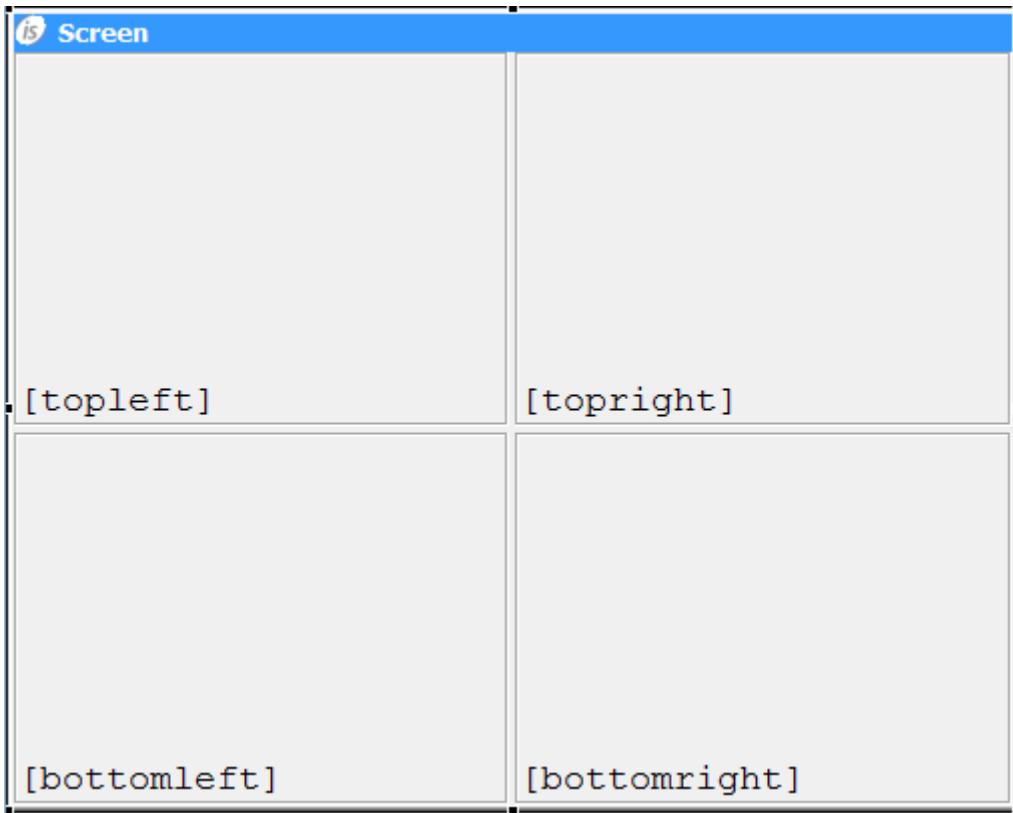
It's possible to create docking windows with the IDE by following these steps:

1. Click on the Screen title bar in order to load screen properties in the Property View.
2. Set the property "window type" to "DOCKING".
3. Right click on the Screen title bar and select "Docking Window Layout" from the pop-up menu.

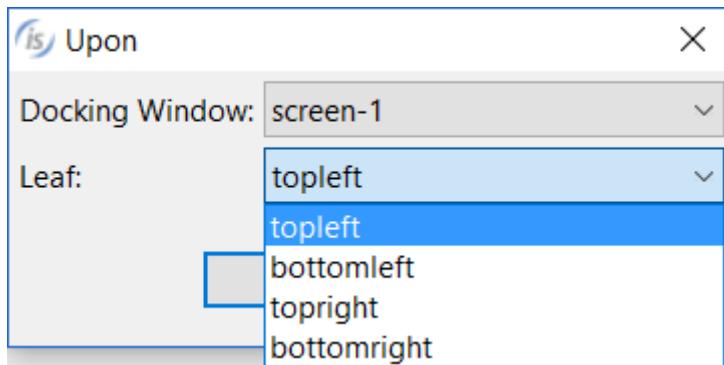
The Layout editor is open. Use the buttons at the top to create ROWS, COLUMNS and LEAFS. Provide a name and a weight for each item. The following screenshot, for example, shows a layout made of 4 different dockable windows where each dockable window takes a quarter of the docking window.



The above settings will make the Screen look like this in the Screen Designer.



4. Create a new Screen
5. Select the new Screen and set the property "window type" to "DOCKABLE".
6. Set the property "upon leaf" to one of the leafs defined in point 3.



7. Draw controls on the Screen and code their procedures.
8. Repeat from step 4 for each dockable window that you want to add to the docking window.

How to create MDI windows

It's possible to create MDI windows with the IDE by respecting the following rules:

- Each MDI window should be managed by a separate program.
- The program that hosts the MDI-PARENT window must call the programs that host the MDI-CHILD windows passing the window handle to them as parameter.
- The programs that host the MDI-PARENT window must include a data item of type *handle of window* in order to receive the MDI-PARENT window handle. Such data item must be used to set the field "upon variable" among *Variables* of the Screen.

Steps to create an MDI-PARENT window:

1. Work in *Structural View*
2. Expand *Screen Section* of the desired program
3. Select the Screen that you want to make an MDI-PARENT
4. Among *Properties*, set "window type" to "MDI-PARENT".

Steps to create an MDI-CHILD window:

1. Work in *Structural View*
2. Expand *Screen Section* of the desired program
3. Select the Screen that you want to make an MDI-CHILD
4. Among *Properties*, set "window type" to "MDI-CHILD"
5. Among *Variables*, set "upon variable" to the Linkage Section data item that receives the MDI-PARENT window handle.

Screen Designer Reference

This section lists all the Properties you can set for controls drawn on the window and for the window itself. If set to a value different than default, then properties are generated in the source code, otherwise they're not.

The Property view in the Screen Designer allows the user to set the initial value of a property, that is the value that appears in Screen Section. The value of a property can be inquired or modified in the code of Event Paragraph (that will become program Procedure Division) through INQUIRE and MODIFY statements.

For additional information about control properties and styles, consult [Controls Reference](#).

Property view items are grouped in the following categories:

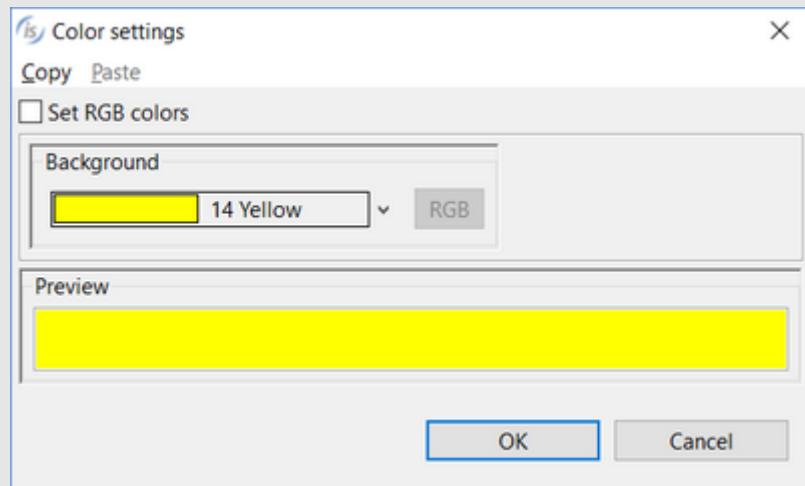
Properties	Control properties constant values and styles generated in the Screen Section of the program.
Events	Paragraphs that manage specific event procedures
Exceptions	Paragraphs that manage specific exception procedures
Procedures	Paragraphs that manage Events, Exceptions and focus change
Variables	Each variable of the program Data Division can be associated with the items below. When a variable is specified, it's generated in place of the corresponding constant value you may have set between Properties.

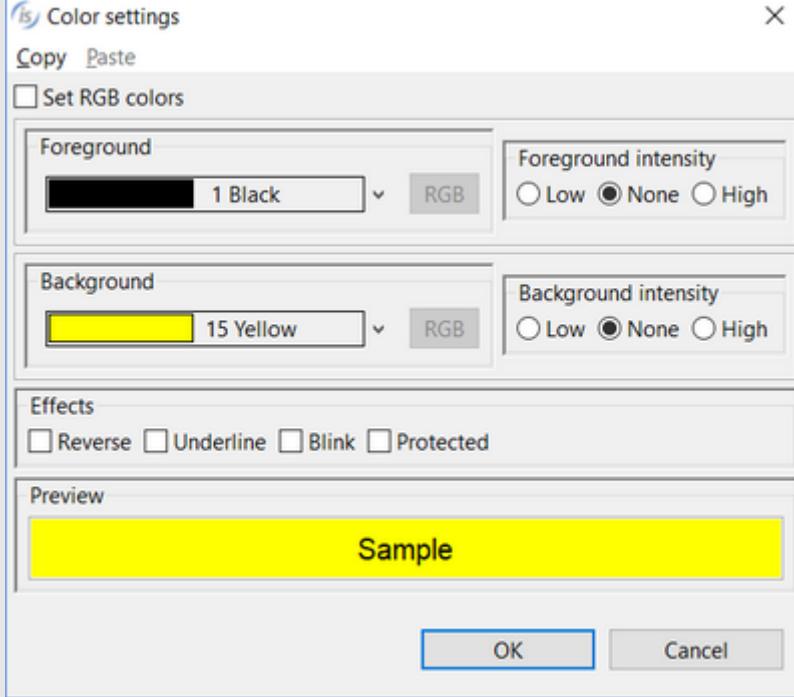
BAR

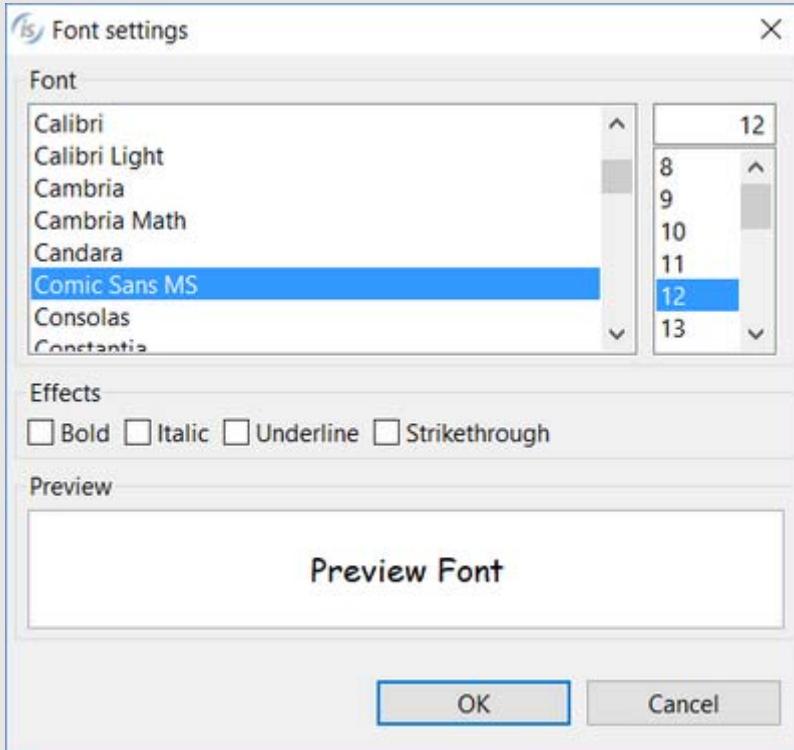
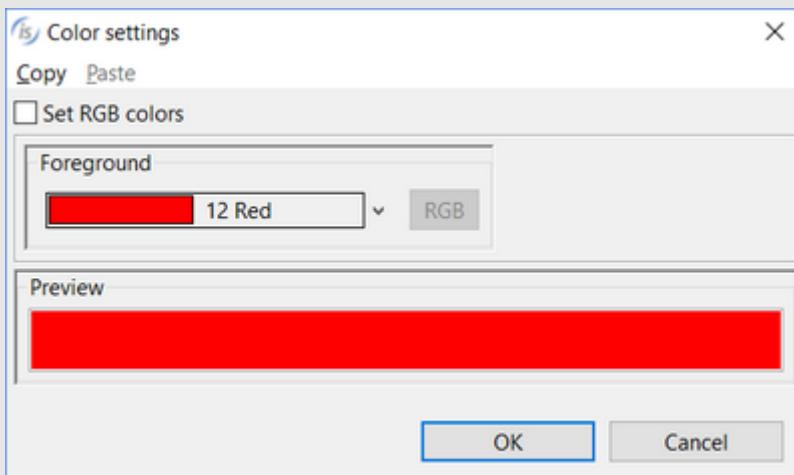
Refer to [BAR](#) for details about properties, styles and events of this control.

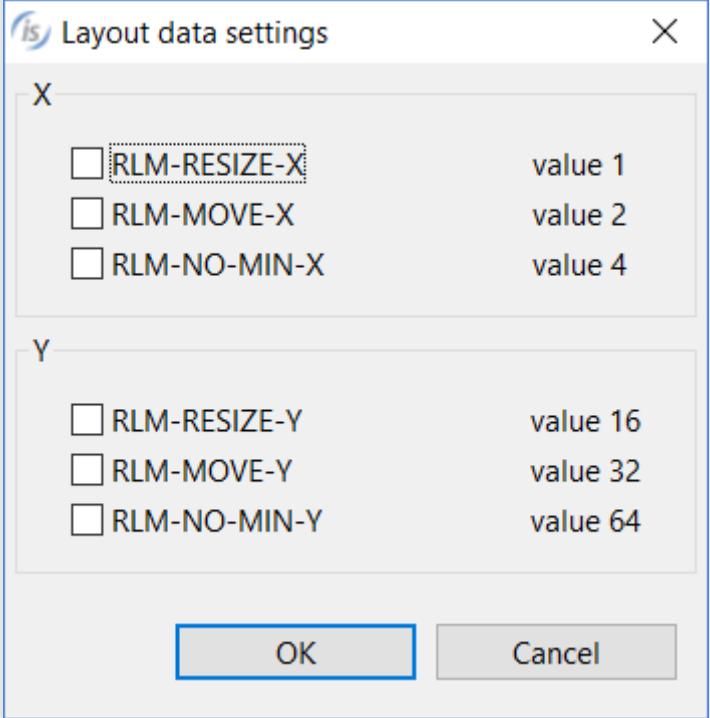
Properties

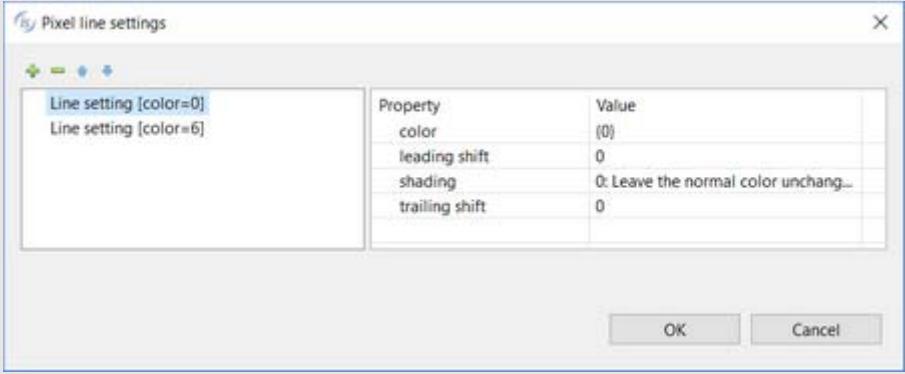
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
background-color	Opens a dialog that allows the user to choose the control background color.



color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
dashed	TRUE...The <i>Dashed</i> style is generated FALSE... The <i>Dashed</i> style is not generated
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor Permanent styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
dot-dash	TRUE...The <i>Dot-Dash</i> style is generated FALSE... The <i>Dot-Dash</i> style is not generated
dotted	TRUE...The <i>Dotted</i> style is generated FALSE... The <i>Dotted</i> style is not generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. In the 'Font' dropdown, 'Comic Sans MS' is selected. To its right is a font size dropdown set to 12. Below these are checkboxes for Bold, Italic, Underline, and Strikethrough, all of which are unchecked. A preview window shows the text 'Preview Font' in a bold, italicized, underlined, and strikethrough style. At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a 'Foreground' section with a color swatch labeled '12 Red' and an 'RGB' button. Below it is a 'Preview' window showing a red bar. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE... The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
horizontal	TRUE...The <i>Horizontal</i> style is generated FALSE... The <i>Horizontal</i> style is not generated

id	Specifies the control id. This property is set automatically when the control is drawn.
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules. 
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells

min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
pixel line settings	Opens a dialog that allows the user to set <i>Color</i> , <i>Shading</i> , <i>Leading Shift</i> and <i>Trailing Shift</i> for each line of pixel when the <i>Width</i> property is set to a value greater than 1.
 <p>The dialog box is titled "Pixel line settings". It contains a list of line settings: "Line setting [color=0]" and "Line setting [color=6]". To the right, there is a table with columns "Property" and "Value". The first row shows "color" with value "(0)". The second row shows "leading shift" with value "0". The third row shows "shading" with value "0: Leave the normal color unchanged". The fourth row shows "trailing shift" with value "0". At the bottom right are "OK" and "Cancel" buttons.</p>	
position shift	Specifies the value for the <i>Position-Shift</i> property
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width	Specifies the width in pixels for the BAR control. When this property is set to a value greater than 1 you can specify different settings for each pixels line (see pixel line settings, above)
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
No Events available.	
Exceptions	
No Exceptions available.	

Procedures

No Procedures available.

Variables

color variable	Numeric variable that hosts the color value.
column variable	Numeric variable that hosts the column value.
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state.
help-id variable	Numeric variable that hosts the help id.
hint variable	Alphanumeric variable that hosts the hint value.
id variable	Numeric variable that hosts the control id.
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property.
layout-data variable	Numeric variable that hosts the control resize rules.
lines variable	Numeric variable that hosts the lines value.
line variable	Numeric variable that hosts the line value.
max-height variable	Numeric variable that hosts the maximum height.
max-width variable	Numeric variable that hosts the maximum width.
min-height variable	Numeric variable that hosts the minimum height.
min-width variable	Numeric variable that hosts the minimum width.
position shift variable	Numeric variable that hosts the value for the <i>Position-Shift</i> property.
size variable	Numeric variable that hosts the size value.
visible variable	Numeric variable that hosts the visible state.
width variable	Numeric variable that hosts the value for the <i>Width</i> property.

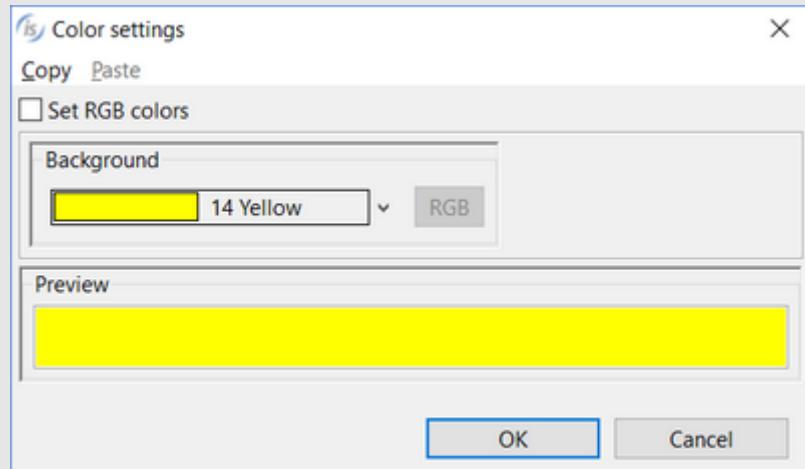
BITMAP

Refer to [BITMAP](#) for details about properties, styles and events of this control.

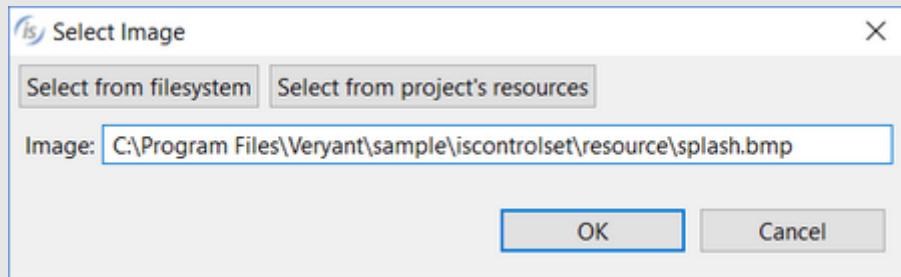
Properties

(name)	Specifies the control name. This property is set automatically when the control is drawn.
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.

background-color Opens a dialog that allows the user to choose the control background color.



bitmap Opens a dialog box that allows the user to select an image file to load into the control.



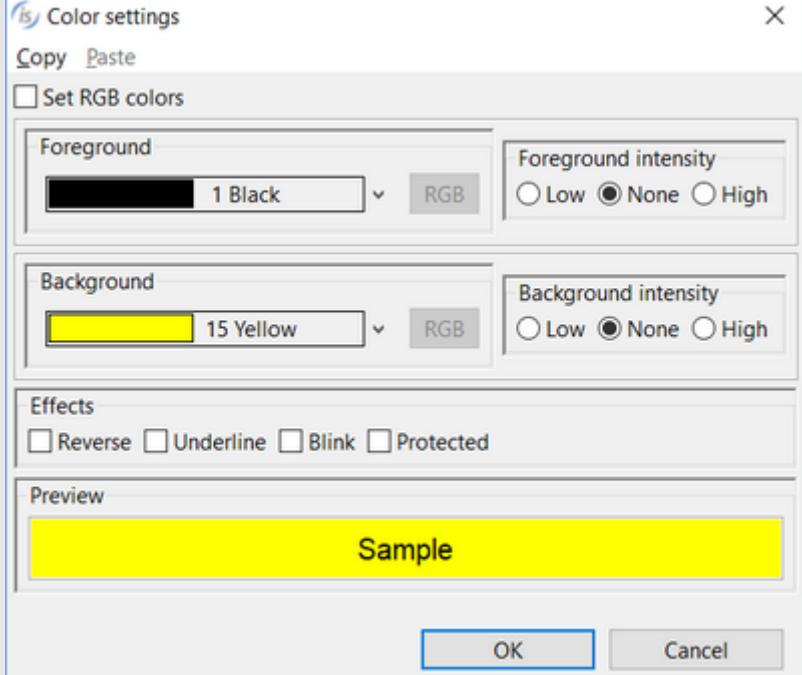
bitmap end Specifies the value for the *Bitmap-End* property.

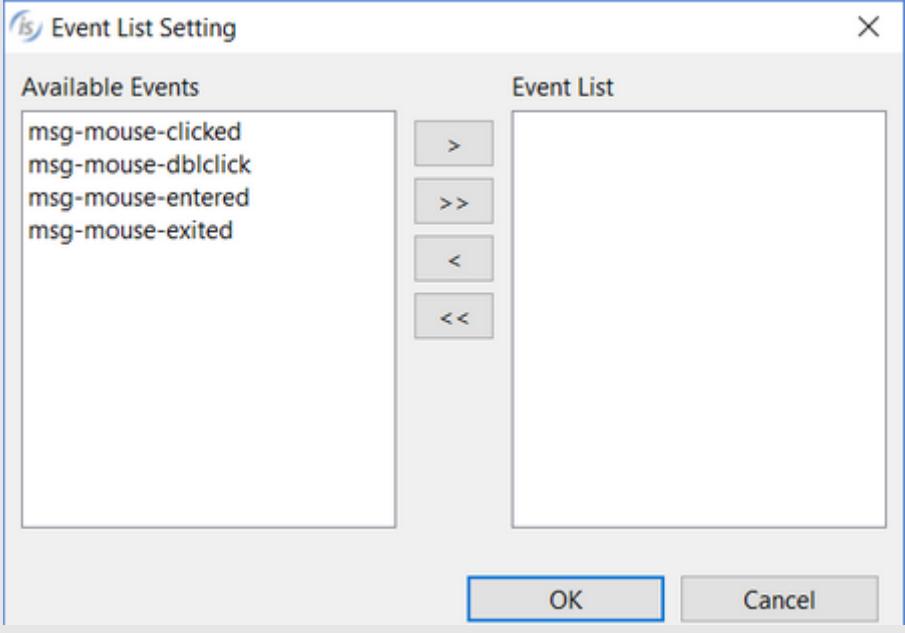
bitmap-number Specifies the value for the *Bitmap-Number* property.

bitmap scale Specifies the value for the *Bitmap-Scale* property.

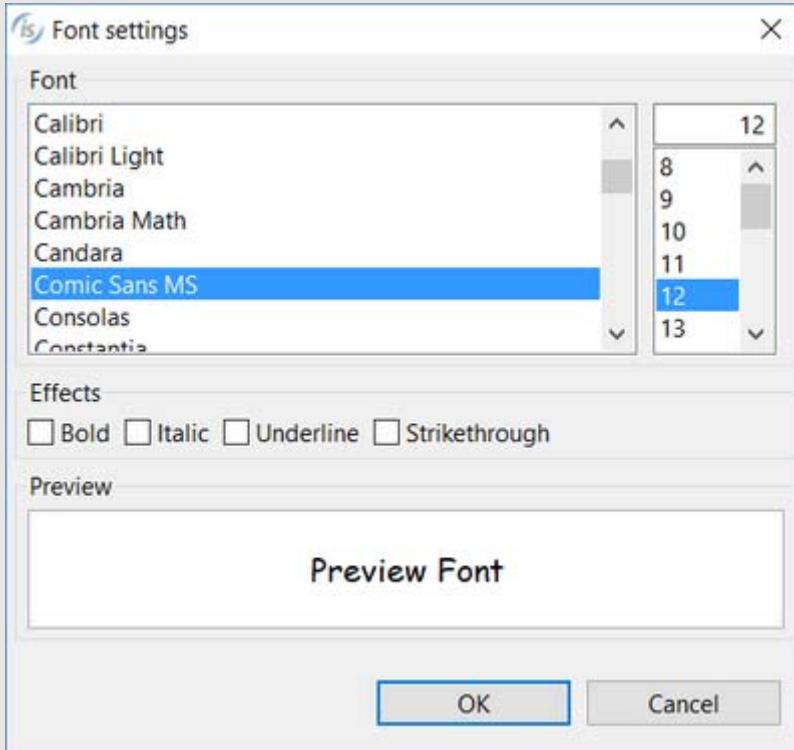
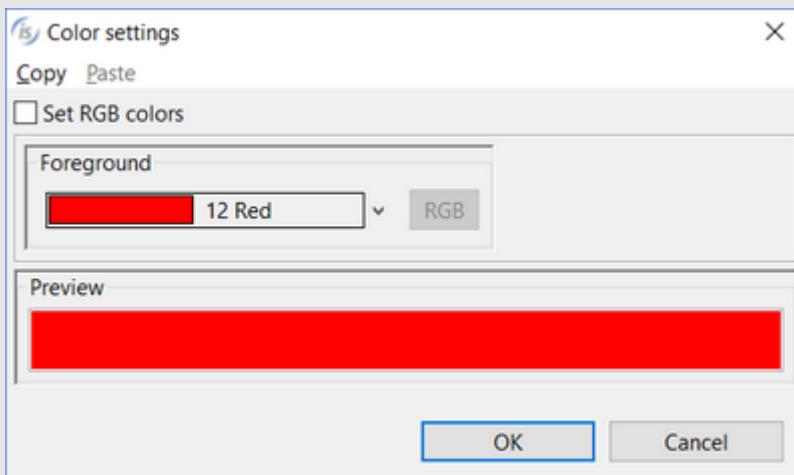
bitmap start Specifies the value for the *Bitmap-Start* property.

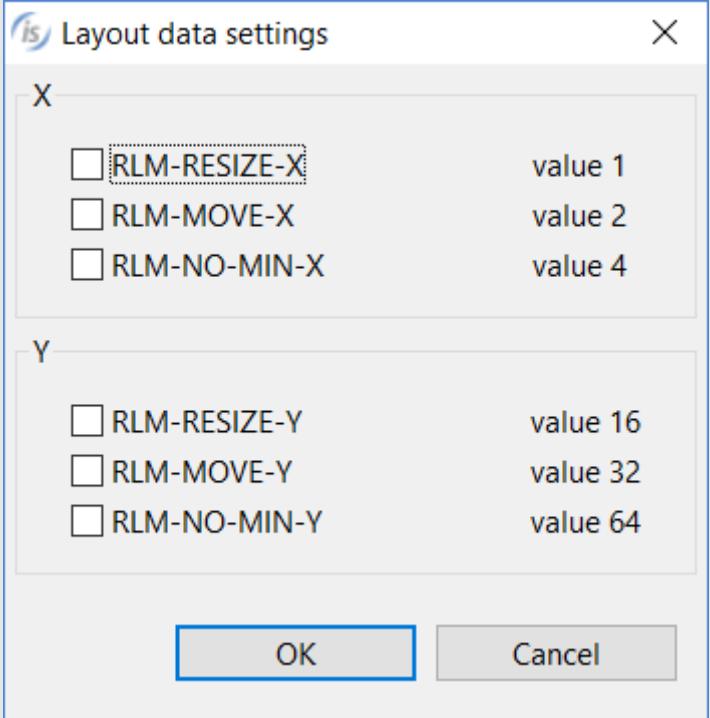
bitmap timer Specifies the value for the *Bitmap-Timer* property.

color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor <i>Permanent</i> styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. The "Available Events" section contains four items: "msg-mouse-clicked", "msg-mouse-dblclick", "msg-mouse-entered", and "msg-mouse-exited". Between the two sections are four buttons: a top-right button with a right-pointing arrow (>), a middle-right button with a double-right-pointing arrow (gg), a bottom-right button with a left-pointing arrow (<), and a bottom-left button with a double-left-pointing arrow (cc). At the bottom of the dialog are two buttons: "OK" (highlighted with a blue border) and "Cancel".</p>

exclude event list NONE... The *Exclude-Event-List* property is not generated.
 0... *Exclude-Event-List=0* is generated.
 1... *Exclude-Event-List=1* is generated.

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a title bar with a close button ('X'). Below it is a 'Font' section containing a list of font names: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical scroll bar and a size dropdown menu showing values 8, 9, 10, 11, 12 (selected), and 13. Below the font list is an 'Effects' section with checkboxes for Bold, Italic, Underline, and Strikethrough, all of which are unchecked. Underneath that is a 'Preview' section with a text area containing the text 'Preview Font'.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a title bar with a close button ('X'). Below it is a toolbar with 'Copy' and 'Paste' buttons, and a checkbox labeled 'Set RGB colors' which is unchecked. The main area is titled 'Foreground' and contains a color swatch set to '12 Red' and a 'RGB' button. Below this is a 'Preview' section with a large red rectangular area. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated. FALSE...The <i>Height-In-Cells</i> style is not generated.
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
id	Specifies the control id. This property is set automatically when the control is drawn.

key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules. 
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn.
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn.
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting. None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value. <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value. <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value.
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse.
max-height	Specifies the control maximum height as expressed in cells.
max-width	Specifies the control maximum width as expressed in cells.
min-height	Specifies the control minimum height as expressed in cells.

min-width	Specifies the control minimum width as expressed in cells.
no-tab	TRUE...The <i>No-Tab</i> style is generated. FALSE...The <i>No-Tab</i> style is not generated.
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn.
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn.
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting. None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value. <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value. <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value.
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn.
transparent color	Specifies the value of the color to be used as transparent color. Unlike other color properties, no dialog is shown here, so you have to type the color code by hand.
visible	NONE...The <i>Visible</i> property is not generated. TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated. FALSE... The <i>Width-In-Cells</i> style is not generated.
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division.
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division.
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division.
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division.
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division.
msg-mouse-clicked event	Allows the user to create a paragraph to handle the MSG-.MOUSE-CLICKED event in the Procedure Division.
msg-mouse-dblclick event	Allows the user to create a paragraph to handle the MSG-.MOUSE-DBLCLICK event in the Procedure Division.
msg-mouse-entered event	Allows the user to create a paragraph to handle the MSG-.MOUSE-ENTERED event in the Procedure Division.

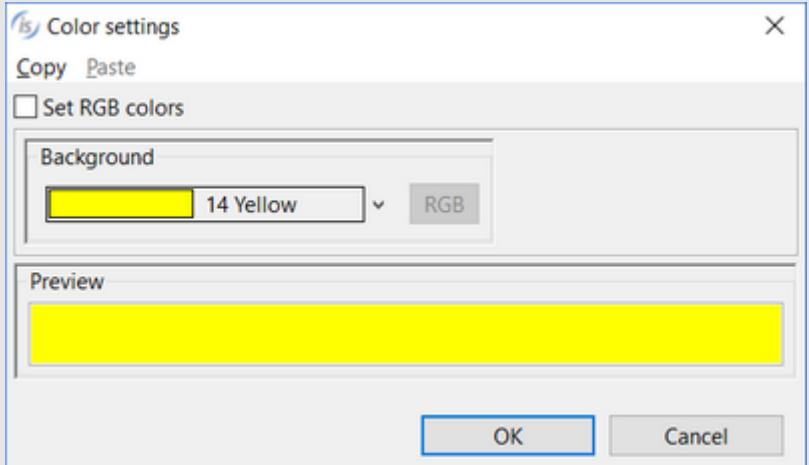
msg-mouse-exited event	Allows the user to create a paragraph to handle the MSG-MOUSE-EXITED event in the Procedure Division.
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division.
other event	Allows the user to create a custom paragraph.
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above.
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above.
other exception	Allows the user to create a custom paragraph.
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE.
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE.
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE.
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE.
Variables	
bitmap end variable	Numeric variable that hosts the value for the <i>Bitmap-End</i> property.
bitmap-number variable	Numeric variable that hosts the value for the <i>Bitmap-Number</i> property.
bitmap start value	Numeric variable that hosts the value for the <i>Bitmap-Start</i> property.
bitmap timer variable	Numeric variable that hosts the value for the <i>Bitmap-Timer</i> property.
color variable	Numeric variable that hosts the color value.
column variable	Numeric variable that hosts the column value.
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state.
help-id variable	Numeric variable that hosts the help id.
hint variable	Alphanumeric variable that hosts the hint value.
id variable	Numeric variable that hosts the control id.
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property.
layout-data variable	Numeric variable that hosts the control resize rules.

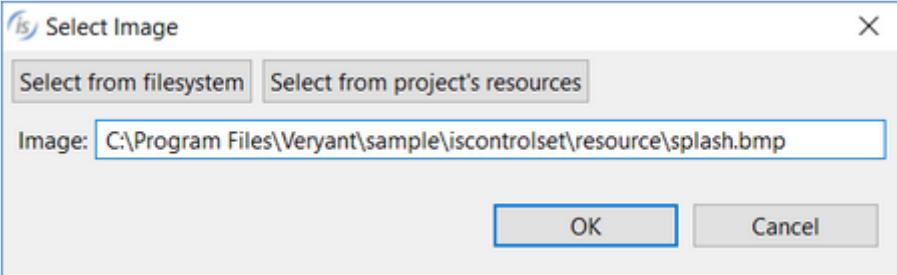
lines variable	Numeric variable that hosts the lines value.
line variable	Numeric variable that hosts the line value.
max-height variable	Numeric variable that hosts the maximum height.
max-width variable	Numeric variable that hosts the maximum width.
min-height variable	Numeric variable that hosts the minimum height.
min-width variable	Numeric variable that hosts the minimum width.
size variable	Numeric variable that hosts the size value.
transparent color variable	Numeric variable that hosts the value for the <i>Transparent-Color</i> property.
visible variable	Numeric variable that hosts the visible state.

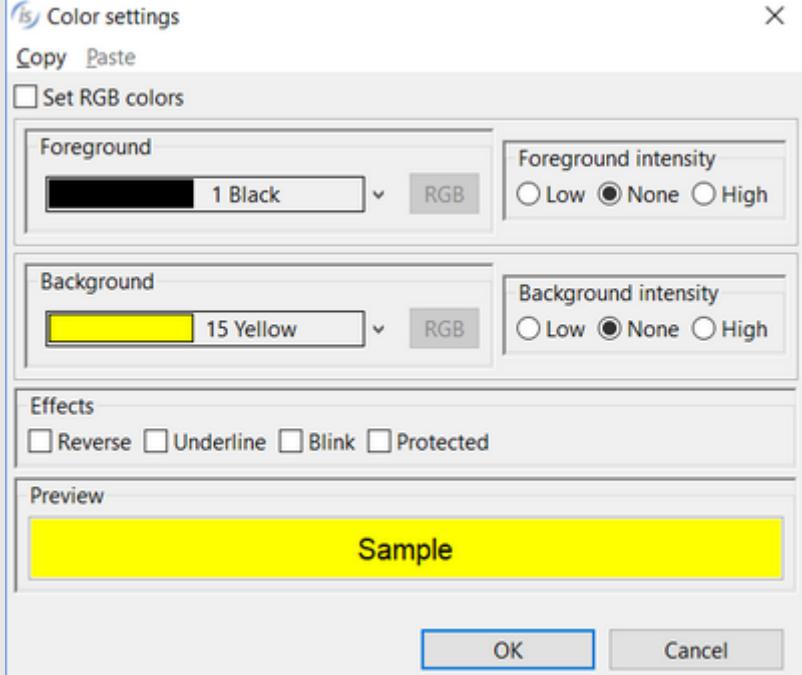
CHECK BOX

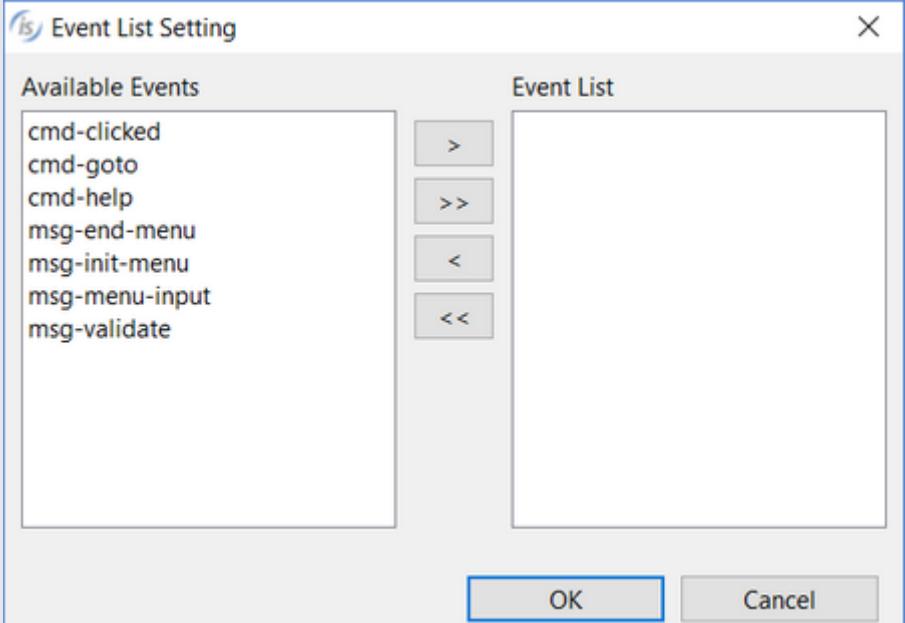
Refer to [CHECK-BOX](#) for details about properties, styles and events of this control.

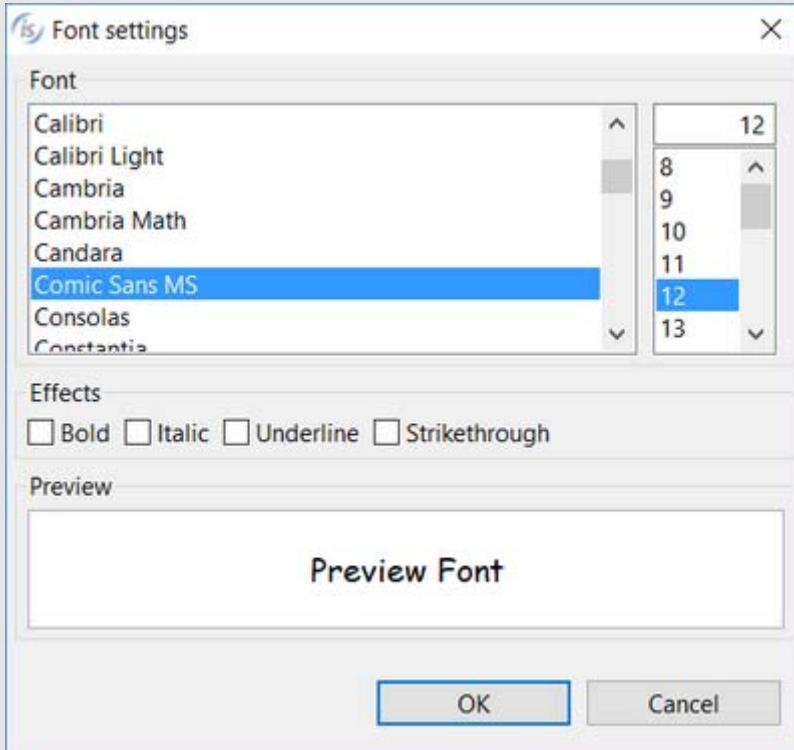
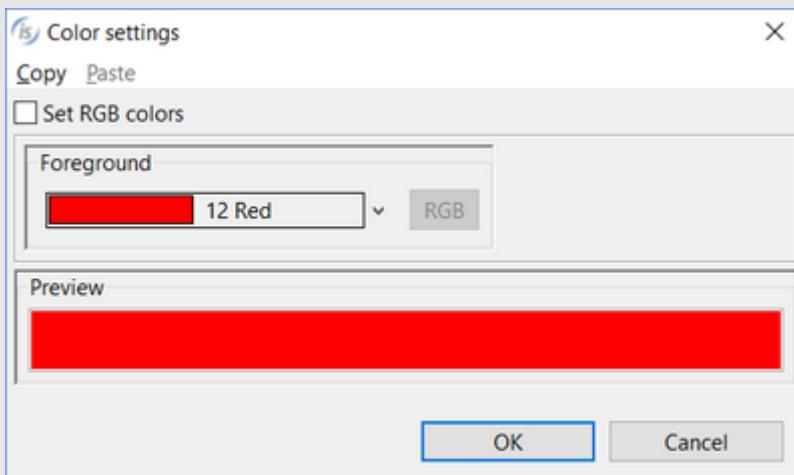
Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
auto-fit	TRUE... The image is scaled in order to fit the control boundaries. This is achieved through additional code generated in the Procedure Division. FALSE...The image is shown as is, if it's too large for control boundaries, it will be truncated.
background-color	Opens a dialog that allows the user to choose the control background color.

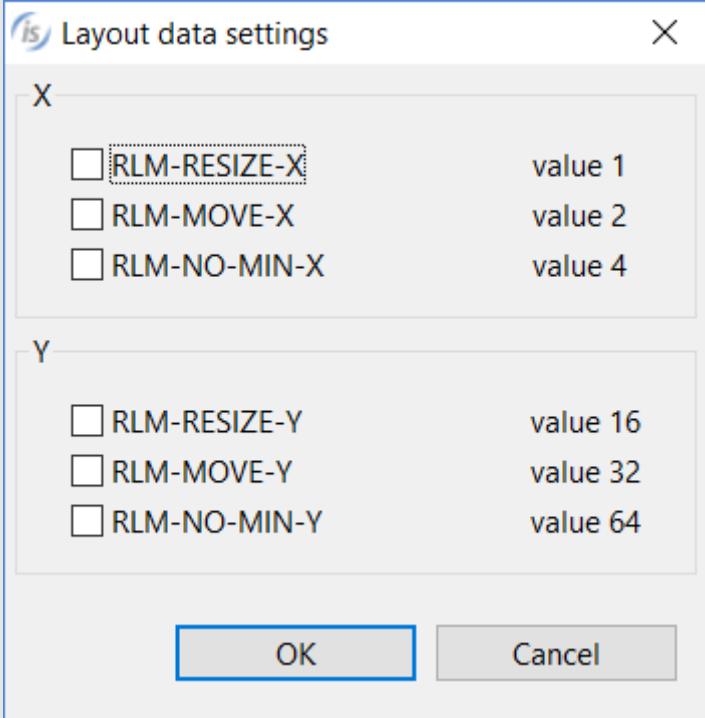


bitmap	Opens a dialog box that allows the user to select an image file to load into the control
	
bitmap-disabled	Specifies the value for the <i>Bitmap-Disabled</i> property
bitmap-disabled-selected	Specifies the value for the <i>Bitmap-Disabled-Selected</i> property
bitmap-frame	NONE... Neither <i>Framed</i> nor <i>Unframed</i> are generated FRAMED...The <i>Framed</i> style is generated UNFRAMED...The <i>Unframed</i> style is generated
bitmap-number	Specifies the value for the <i>Bitmap-Number</i> property
bitmap-pressed	Specifies the value for the <i>Bitmap-Pressed</i> property
bitmap-rollover	Specifies the value for the <i>Bitmap-Rollover</i> property
bitmap-rollover-selected	Specifies the value for the <i>Bitmap-Rollover-Selected</i> property
bitmap-selected	Specifies the value for the <i>Bitmap-Selected</i> property
bitmap-square	TRUE...The <i>Square</i> style is generated FALSE...The <i>Square</i> style is not generated
bitmap-width	Specifies the value for the <i>Bitmap-Width</i> property

color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor <i>Permanent</i> styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main panes: "Available Events" on the left and "Event List" on the right. The "Available Events" pane contains a list of event names: cmd-clicked, cmd-goto, cmd-help, msg-end-menu, msg-init-menu, msg-menu-input, and msg-validate. Between the two panes are four buttons: > (top), >> (middle), < (bottom), and << (bottom). At the bottom of the dialog are "OK" and "Cancel" buttons.</p>
exception-value	Specifies the value for the <i>Exception-Value</i> property
exclude event list	<p>NONE... The <i>Exclude-Event-List</i> property is not generated.</p> <p>0... <i>Exclude-Event-List=0</i> is generated.</p> <p>1... <i>Exclude-Event-List=1</i> is generated.</p>
flat	<p>TRUE...The <i>Flat</i> style is generated</p> <p>FALSE...The <i>Flat</i> style is not generated</p>

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a title bar with a close button ('X'). Below it is a 'Font' section containing a list of font names: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical scroll bar and a size dropdown menu showing values 8, 9, 10, 11, 12 (selected), and 13. Below the font list is an 'Effects' section with checkboxes for Bold, Italic, Underline, and Strikethrough, all of which are unchecked. Underneath that is a 'Preview' section with a text area containing the text 'Preview Font'.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a title bar with a close button ('X'). Below it is a 'Copy' and 'Paste' button, followed by a checkbox labeled 'Set RGB colors'. Underneath is a 'Foreground' section with a color swatch set to '12 Red' and an 'RGB' button. Below that is a 'Preview' section with a large red rectangular area. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
id	Specifies the control id. This property is set automatically when the control is drawn.

key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules. 
left-text	TRUE...The <i>Left-Text</i> style is generated FALSE...The <i>Left-Text</i> style is not generated
left-text-alignment	LEFT...Generates <i>Left-Text-Alignment=1</i> RIGHT...Generates <i>Left-Text-Alignment=0</i>
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse

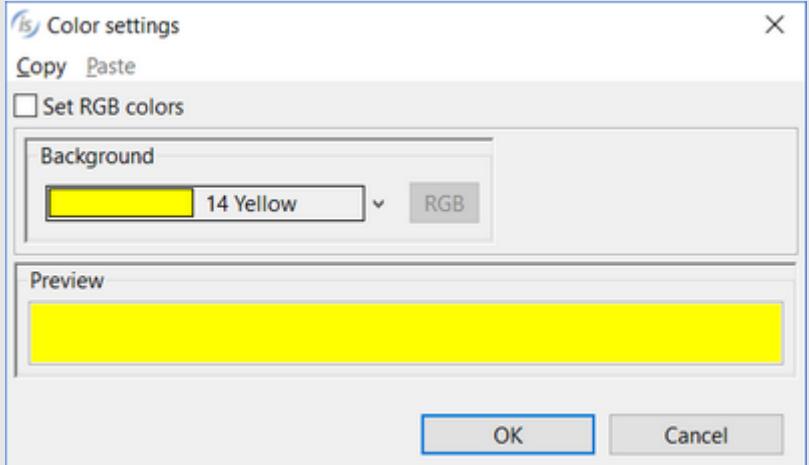
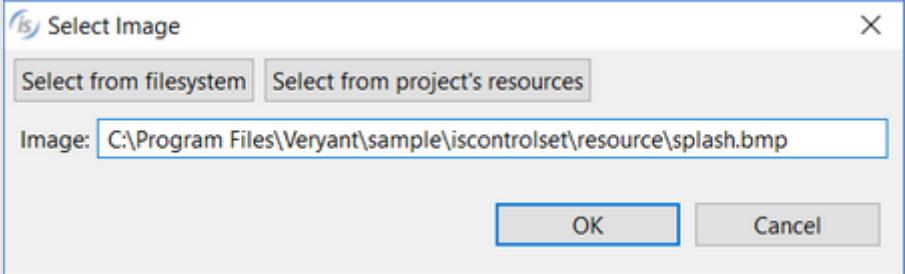
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
multiline	TRUE...The <i>Multiline</i> style is generated FALSE...The <i>Multiline</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
notify	TRUE...The <i>Notify</i> style is generated FALSE...The <i>Notify</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
self-act	TRUE...The <i>Self-Act</i> style is generated FALSE...The <i>Self-Act</i> style is not generated
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
termination-value	Specifies the value for the <i>Termination-Value</i> property
title	Specifies the value for the <i>Title</i> property
title-position	0...NONE 1...LEFT 2...RIGHT 3...TOP 4...BOTTOM 5...CENTER
transparent	TRUE...The <i>Transparent</i> style is generated FALSE...The <i>Transparent</i> style is not generated
value	Specifies the value for the <i>Value</i> property
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated

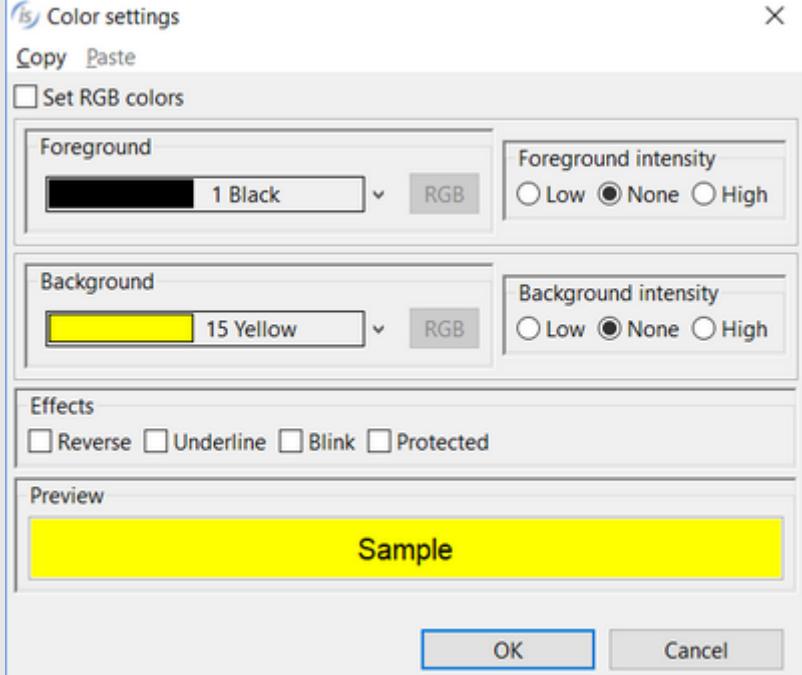
vtop	TRUE...The <i>Vtop</i> style is generated FALSE...The <i>Vtop</i> style is not generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-clicked event	Allows the user to create a paragraph to handle the CMD-CLICKED event in the Procedure Division
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-clicked exception	Allows the user to create a paragraph to handle the CMD-CLICKED event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Variables	

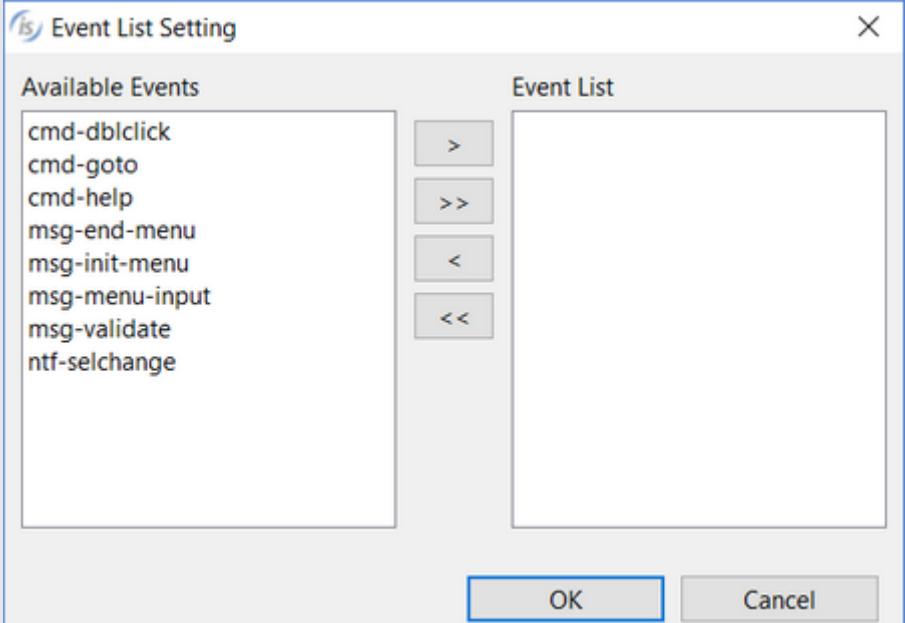
bitmap-disabled variable	Numeric variable that hosts the value for the <i>Bitmap-Disabled</i> property
bitmap-disabled-selected variable	Numeric variable that hosts the value for the <i>Bitmap-Disabled-Selected</i> property
bitmap-number variable	Numeric variable that hosts the value for the <i>Bitmap-Number</i> property
bitmap-pressed value	Numeric variable that hosts the value for the <i>Bitmap-Pressed</i> property
bitmap-rollover-selected variable	Numeric variable that hosts the value for the <i>Bitmap-Rollover-Selected</i> property
bitmap-rollover variable	Numeric variable that hosts the value for the <i>Bitmap-Rollover</i> property
bitmap-selected variable	Numeric variable that hosts the value for the <i>Bitmap-Selected</i> property
bitmap-width variable	Numeric variable that hosts the value for the <i>Bitmap-Width</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
exception-value variable	Numeric variable that hosts the value for the <i>Exception-Value</i> property
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the value for the <i>Hint</i> property
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
title variable	Numeric variable that hosts the value for the <i>Title</i> property
title-position variable	Numeric variable that hosts the value for the <i>Title-Position</i> property
value variable	Numeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

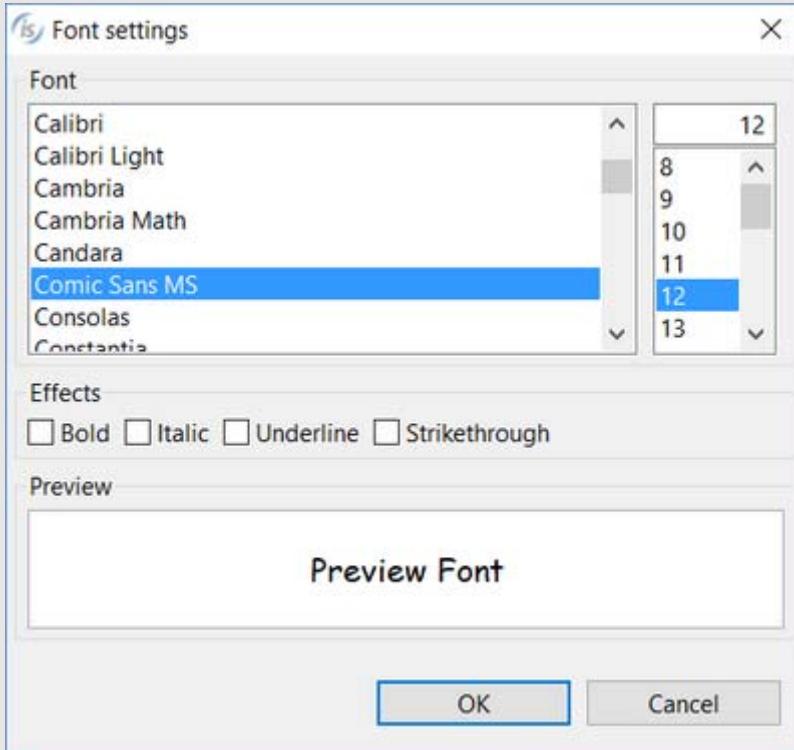
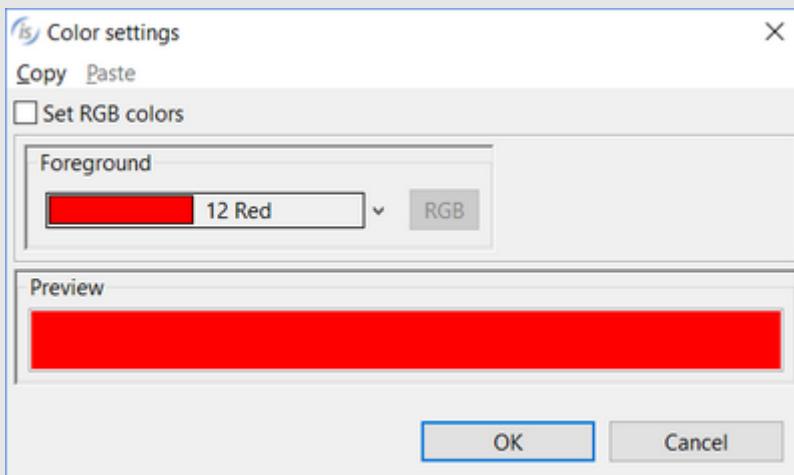
COMBO BOX

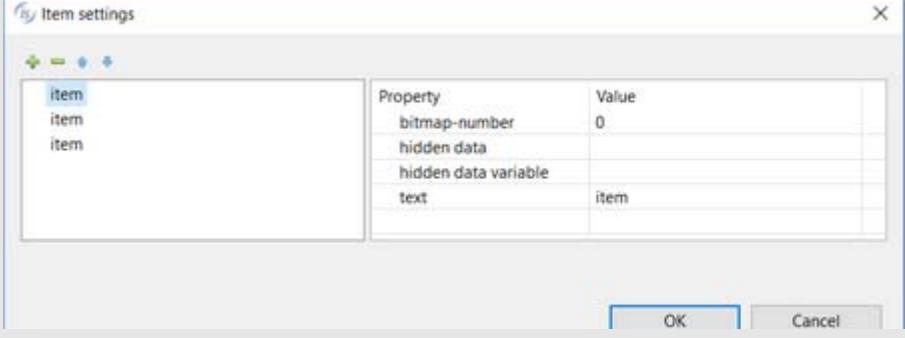
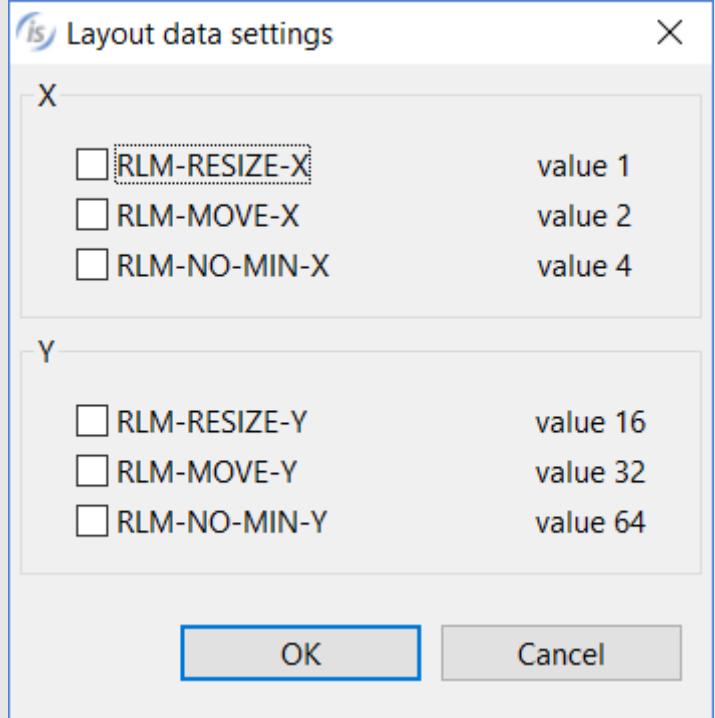
Refer to [COMBO-BOX](#) for details about properties, styles and events of this control.

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
background-color	Opens a dialog that allows the user to choose the control background color. 
bitmap	Opens a dialog box that allows the user to select an image file to load into the control 
bitmap-width	Specifies the value for the <i>Bitmap-Width</i> property
border	Allows the user to set one of the following three styles: 3-D BOXED NO-BOX
case	None....Neither the <i>Upper</i> nor <i>Lower</i> styles are generated UPPER.... <i>Upper</i> style is generated LOWER.... <i>Lower</i> style is generated

color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor <i>Permanent</i> styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. The "Available Events" section contains a list of event names: cmd-dblclick, cmd-goto, cmd-help, msg-end-menu, msg-init-menu, msg-menu-input, msg-validate, and ntf-selchange. Between the two sections are four buttons: a top-right button with a right-pointing arrow (>), a middle-right button with a double-right-pointing arrow (gg), a bottom-right button with a left-pointing arrow (<), and a bottom-left button with a double-left-pointing arrow (cc). At the bottom of the dialog are two buttons: "OK" (highlighted with a blue border) and "Cancel".</p>	
exception-value	Specifies the value for the <i>Exception-Value</i> property
exclude event list	<p>NONE... The <i>Exclude-Event-List</i> property is not generated.</p> <p>0... <i>Exclude-Event-List=0</i> is generated.</p> <p>1... <i>Exclude-Event-List=1</i> is generated.</p>

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. On the right, there's a vertical font size selector with values 8, 9, 10, 11, 12, and 13, where '12' is selected. Below the font list are sections for 'Effects' (Bold, Italic, Underline, Strikethrough) and 'Preview' (a text area containing 'Preview Font'). At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It includes a 'Set RGB colors' checkbox, which is unchecked. Below it is a 'Foreground' color picker set to '12 Red'. A preview window shows a red rectangle. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property
id	Specifies the control id. This property is set automatically when the control is drawn.

item-to-add	Opens a dialog that allows the user to set the text and icon for each single item
	
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn

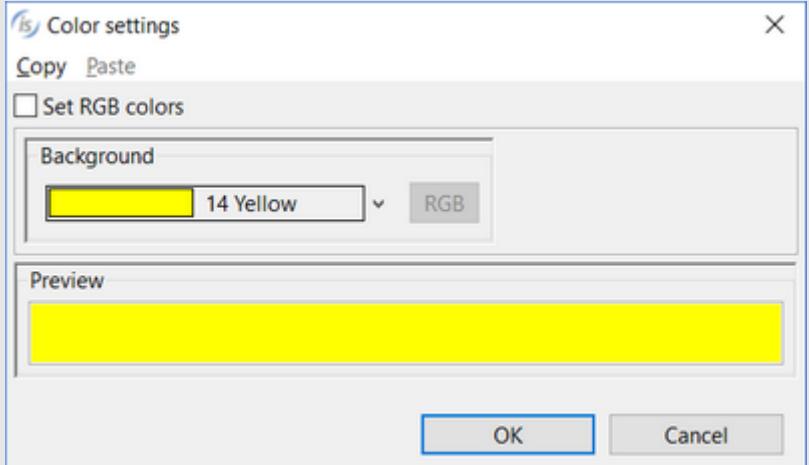
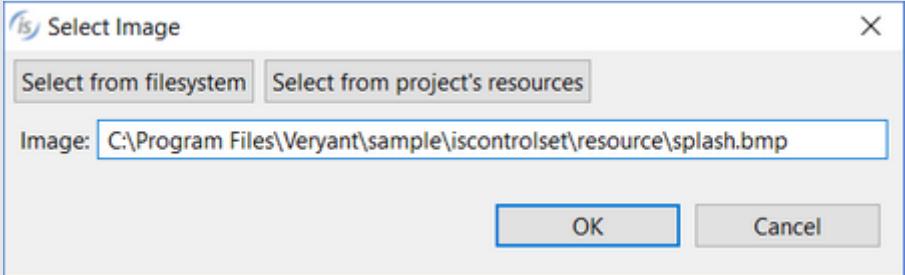
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
no-autosel	TRUE...The <i>No-Autoselection</i> style is generated FALSE...The <i>No-Autoselection</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
notify dblclick	TRUE...The <i>Notify-Dblclick</i> style is generated FALSE...The <i>Notify-Dblclick</i> style is not generated
notify selchange	TRUE...The <i>Notify-Selchange</i> style is generated FALSE...The <i>Notify-Selchange</i> style is not generated
placeholder	Specifies the value for the <i>Placeholder</i> property
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
selection-index	Specifies the value for the <i>Selection-Index</i> property
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
style	DROP-DOWN... The <i>Drop-Down</i> style is generated DROP-LIST... The <i>Drop-List</i> style is generated
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
termination-value	Specifies the value for the <i>Termination-Value</i> property
unsorted	TRUE... The <i>Unsorted</i> style is generated FALSE... The <i>Unsorted</i> style is not generated

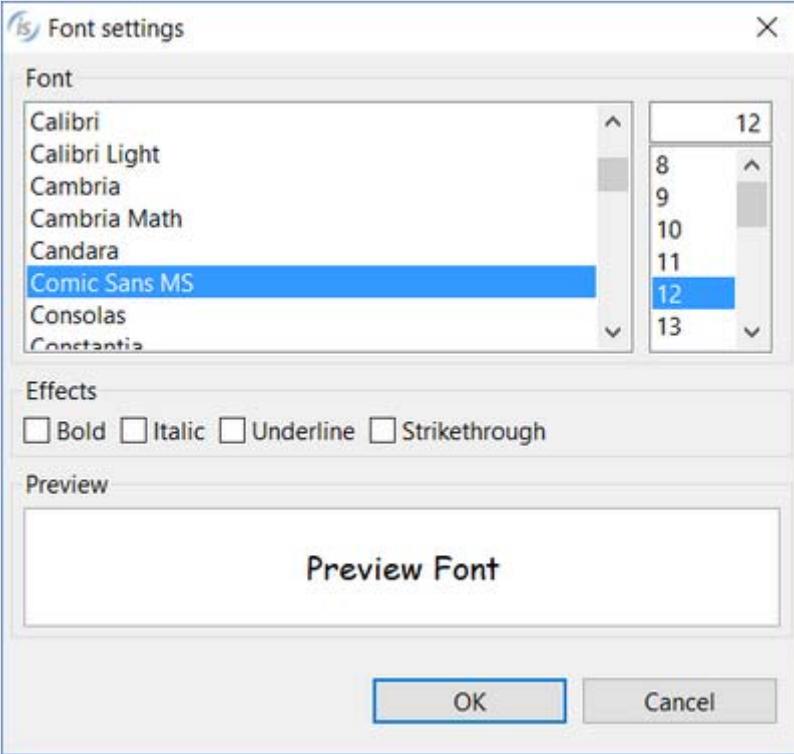
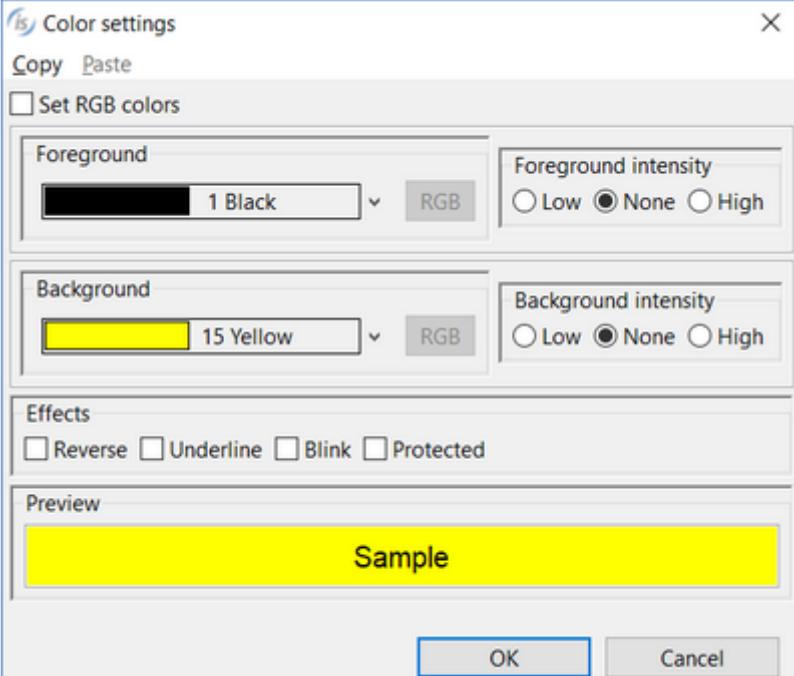
value	Specifies the value for the <i>Value</i> property
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-dblclick event	Allows the user to create a paragraph to handle the CMD-DBLCLICK event in the Procedure Division
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
ntf-selchange event	Allows the user to create a paragraph to handle the NTF-SELCHANGE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-dblclick exception	Allows the user to create a paragraph to handle the CMD-DBLCLICK event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-selchange exception	Allows the user to create a paragraph to handle the NTF-SELCHANGE event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.

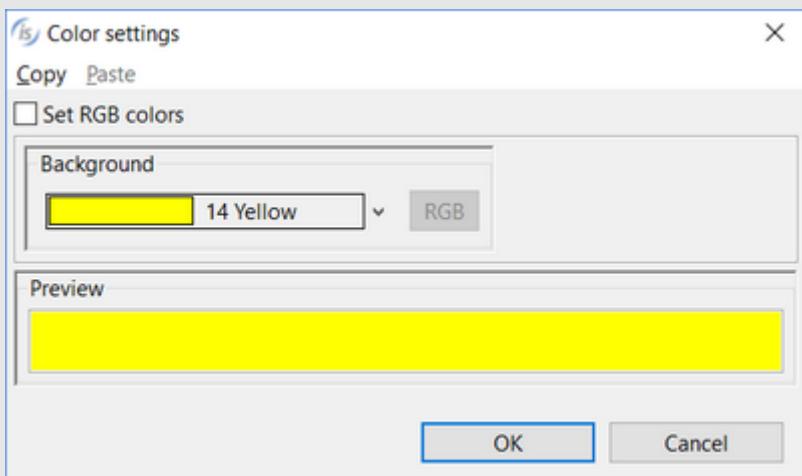
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Link To	Associates a paragraph with the control that will be executed when the control is double clicked
Variables	
bitmap-width variable	Numeric variable that hosts the value for the <i>Bitmap-Width</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
exception-value variable	Numeric variable that hosts the value for the <i>Exception-Value</i> property
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the value for the <i>Hint</i> property
id variable	Numeric variable that hosts the control id
item-to-add variable	Alphanumeric variable that hosts the value for the <i>Item-To-Add</i> property
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
value container	occurs item that hosts control items
value variable	Alphanumeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

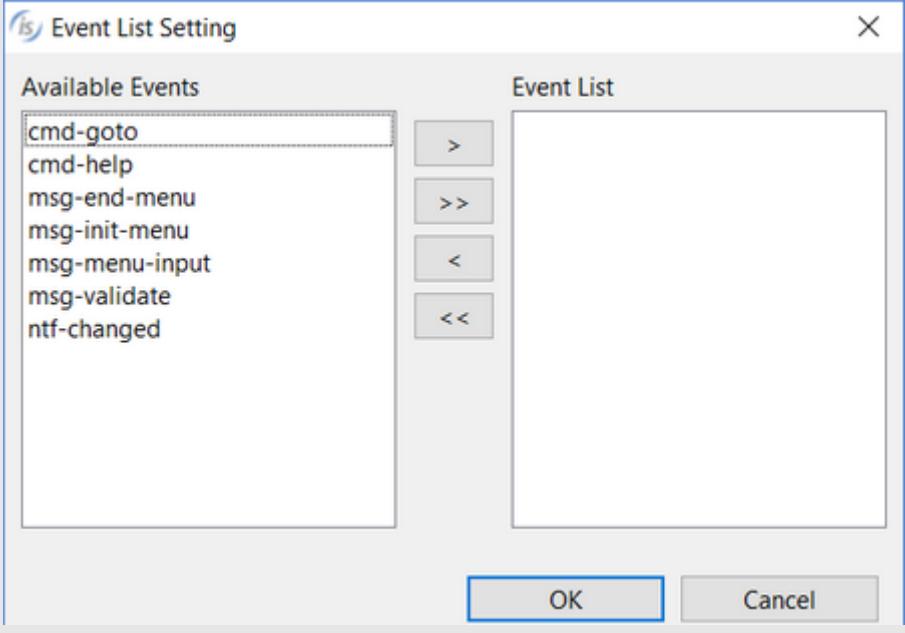
DATE ENTRY

Refer to [DATE-ENTRY](#) for details about properties, styles and events of this control.

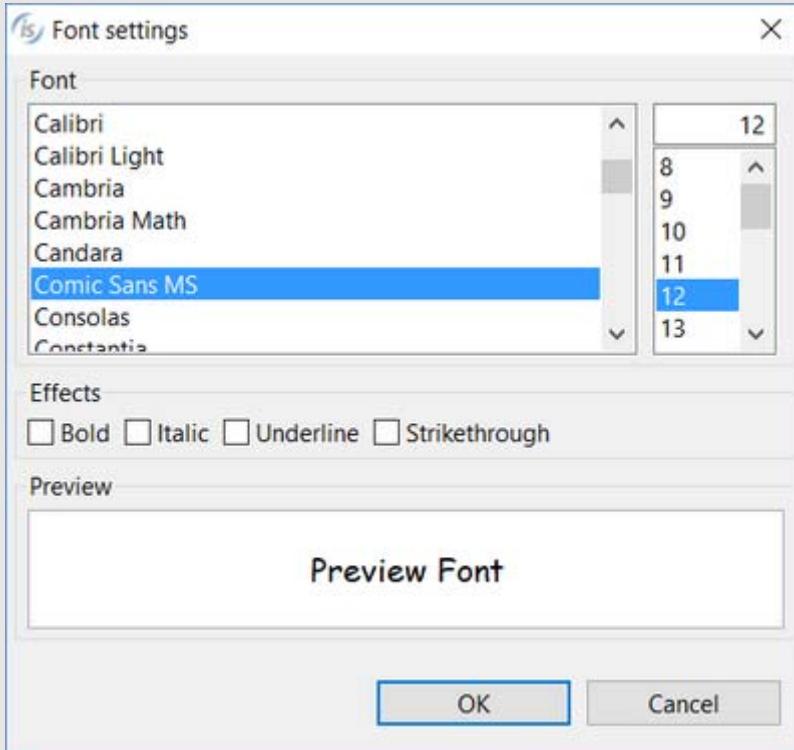
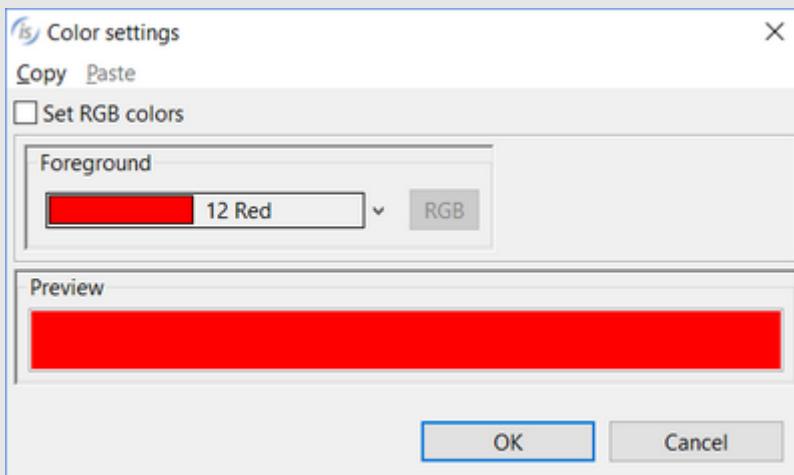
Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
allow empty	TRUE...The <i>Allow-Empty</i> style is generated FALSE...The <i>Allow-Empty</i> style is not generated
background-color	Opens a dialog that allows the user to choose the control background color. 
bitmap	Opens a dialog box that allows the user to select an image file to load into the control 
bitmap-number	Specifies the value for the <i>Bitmap-Number</i> property
bitmap-width	Specifies the value for the <i>Bitmap-Width</i> property

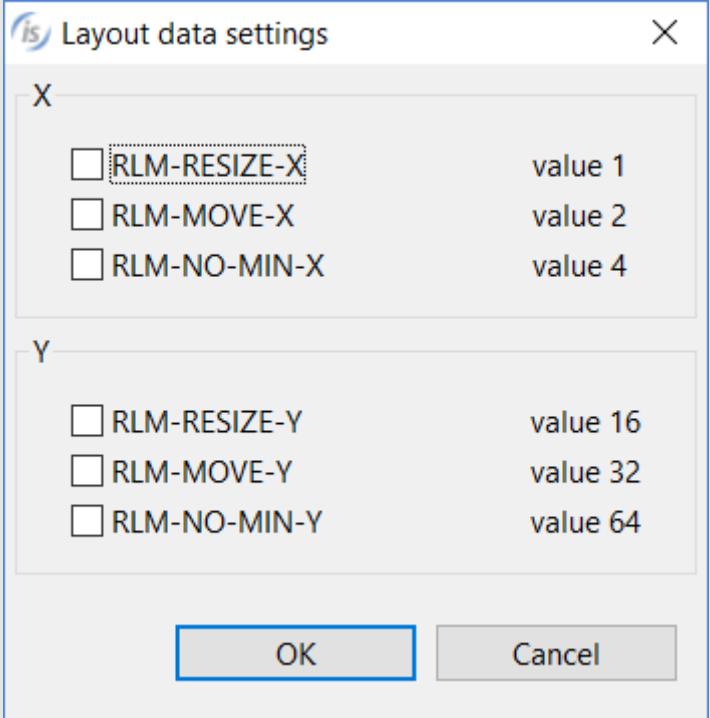
calendar font	Opens a dialog that allows the user to choose the calendar font.
 A screenshot of the 'Font settings' dialog box. At the top, it says 'Font' and shows a list of font names: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical size selector with values 8, 9, 10, 11, 12, 13, and 12 at the top. Below the font list are sections for 'Effects' (Bold, Italic, Underline, Strikethrough) and 'Preview' (a box labeled 'Preview Font'). At the bottom are 'OK' and 'Cancel' buttons. <p>The 'Font' section lists the following fonts: Calibri Calibri Light Cambria Cambria Math Candara Comic Sans MS Consolas Constantia</p> <p>The size selector shows the following options: 8 9 10 11 12 13</p>	
color	Opens a dialog that allows the user to choose the control color.
 A screenshot of the 'Color settings' dialog box. At the top, it says 'Color settings' and has 'Copy' and 'Paste' buttons. There is a checkbox 'Set RGB colors'. The 'Foreground' section shows a color swatch set to 'Black' (labeled '1 Black') with an 'RGB' button. The 'Background' section shows a color swatch set to 'Yellow' (labeled '15 Yellow') with an 'RGB' button. Both sections have 'Foreground intensity' and 'Background intensity' sub-sections with 'Low', 'None', and 'High' options, where 'None' is selected for both. Below these are 'Effects' options (Reverse, Underline, Blink, Protected) and a 'Preview' section with a yellow box labeled 'Sample'. At the bottom are 'OK' and 'Cancel' buttons. <p>The 'Foreground' section shows 'Black' (1 Black) and 'Background' shows 'Yellow' (15 Yellow).<p>The 'Effects' section includes: □ Reverse □ Underline □ Blink □ Protected</p></p>	

column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
decoration background	Opens a dialog to retrieve the value for the <i>Decoration-Background</i> property
	
decoration background visible	TRUE...The <i>Decoration-Background-Visible</i> style is generated FALSE...The <i>Decoration-Background-Visible</i> style is not generated
decoration border visible	TRUE...The <i>Decoration-Border-Visible</i> style is generated FALSE...The <i>Decoration-Border-Visible</i> style is not generated
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor Permanent styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
display format	Specifies the value for the <i>Display-Format</i> property
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. In the "Available Events" section, there is a scrollable list containing the following items: cmd/goto, cmd/help, msg-end-menu, msg-init-menu, msg-menu-input, msg-validate, and ntf-changed. Between the two sections are four buttons: a top-right button with a right-pointing arrow (>), a middle-right button with a double-right-pointing arrow (>>), a bottom-right button with a left-pointing arrow (<), and a bottom-left button with a double-left-pointing arrow (<<). At the bottom of the dialog are two buttons: "OK" (highlighted with a blue border) and "Cancel".</p>

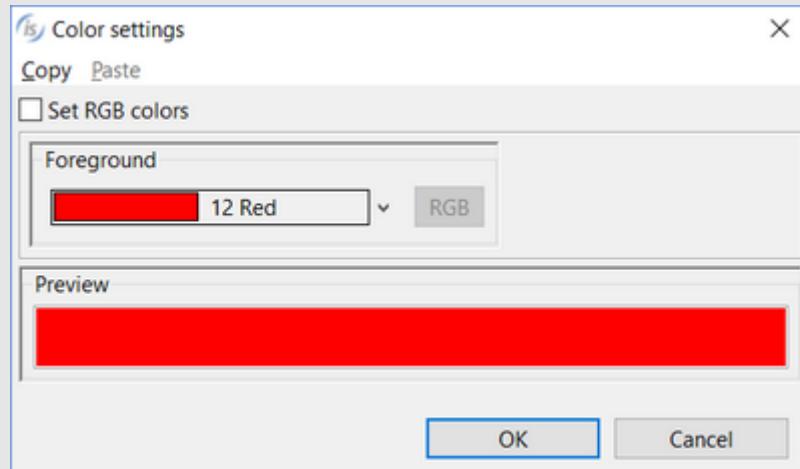
exclude event list NONE... The *Exclude-Event-List* property is not generated.
 0... *Exclude-Event-List=0* is generated.
 1... *Exclude-Event-List=1* is generated.

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a title bar 'Font settings' with a close button 'X'. Below it is a 'Font' section containing a list of font names: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical scroll bar and a size dropdown menu showing values 8, 9, 10, 11, 12 (highlighted), and 13. Below the font list is an 'Effects' section with checkboxes for Bold, Italic, Underline, and Strikethrough, all of which are unchecked. Underneath that is a 'Preview' section with a text area containing the text 'Preview Font'.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a title bar 'Color settings' with a close button 'X'. Below it is a 'Copy' and 'Paste' button, followed by a checkbox 'Set RGB colors' which is unchecked. The main area is titled 'Foreground' and contains a color swatch labeled '12 Red' and an 'RGB' button. Below this is a 'Preview' section with a large red rectangular area. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>help-id</i> .
hint	Specifies the value for the <i>Hint</i> property
id	Specifies the control id. This property is set automatically when the control is drawn.

key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules. 
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
maxday characters	Specifies the value for the <i>Maxday-Characters</i> property. You can choose a value between 0 and 4
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells

min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
no F4	TRUE... The <i>No-F4</i> style is generated FALSE... The <i>No-F4</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
notify change	TRUE...The <i>Notify-Change</i> style is generated FALSE...The <i>Notify-Change</i> style is not generated
no updown	TRUE...The <i>No-Updown</i> style is generated FALSE...The <i>No-Updown</i> style is not generated
numeric	TRUE...The <i>Numeric</i> style is generated FALSE...The <i>Numeric</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
read-only	TRUE...The <i>Read-Only</i> style is generated FALSE...The <i>Read-Only</i> style is not generated
right align	TRUE...The <i>Right-Align</i> style is generated FALSE...The <i>Right-Align</i> style is not generated
self-act	TRUE...The <i>Self-Act</i> style is generated FALSE...The <i>Self-Act</i> style is not generated
show none	TRUE...The <i>Show-None</i> style is generated FALSE...The <i>Show-None</i> style is not generated
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
spinner	TRUE...The <i>Spinner</i> style is generated FALSE...The <i>Spinner</i> style is not generated
style	CENTURY-DATE...The <i>Century-Date</i> style is generated LONG-DATE...The <i>Long-Date</i> style is generated SHORT-DATE...The <i>Short-Date</i> style is generated TIME...The <i>Time</i> style is generated

sunday foreground Opens a dialog to retrieve the value for the *Sunday-Foreground* property



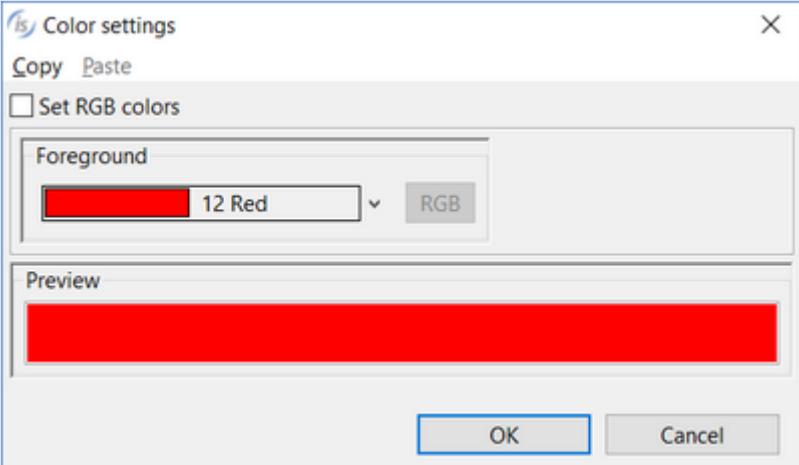
tab order Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn

value Specifies the value for the *Value* property

value format Allows the user to choose one of the following constant values:

DAVF-YYYYMMDD
DAVF-YYMMDD
DAVF-HHMMSShh
DAVF-HHMMSS
DAVF-YYYYMMDDHHMMSShh
DAVF-MMDDYYYY
DAVF-MMDDYY
DAVF-MMDDYYYYHHMMSShh
DAVF-DDMMYYYY
DAVF-DDMMYY
DAVF-DDMMYYYYHHMMSShh

visible NONE...*Visible* property is not generated
TRUE... *Visible=1* is generated
FALSE...*Visible=0* is generated

weekday foreground	Opens a dialog to retrieve the value for the <i>Weekday-Foreground</i> property
	
week of year visible	TRUE...The <i>Week-Of-Year-Visible</i> style is generated FALSE...The <i>Week-Of-Year-Visible</i> style is not generated
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
ntf-changed event	Allows the user to create a paragraph to handle the NTF-CHANGED event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-changed exception	Allows the user to create a paragraph to handle the NTF-CHANGED event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above

other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Variables	
bitmap-number variable	Numeric variable that hosts the value for the <i>Bitmap-Number</i> property
bitmap-width variable	Numeric variable that hosts the value for the <i>Bitmap-Width</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
decoration background variable	Numeric variable that hosts the value for the <i>Decoration-Background</i> property
display format variable	Numeric variable that hosts the value for the <i>Display-Format</i> property
enabled variable	Numeric variable that hosts the enabled state
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the hint value.
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
maxday characters variable	Numeric variable that hosts the value for the <i>Maxday-Characters</i> property
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width

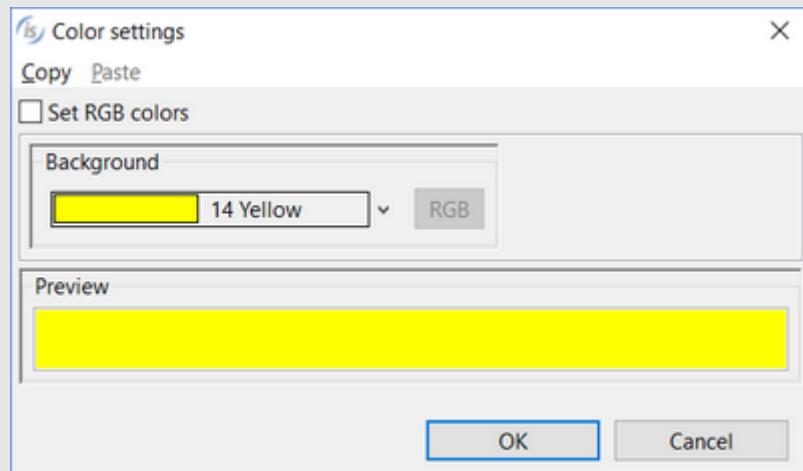
size variable	Numeric variable that hosts the size value
sunday foreground variable	Numeric variable that hosts the value for the <i>Sunday-Foreground</i> property
value variable	Numeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state
weekday foreground variable	Numeric variable that hosts the value for the <i>Weekday-Foreground</i> property

ENTRY FIELD

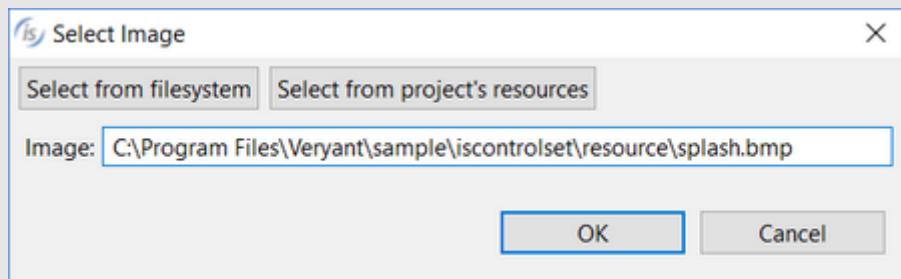
Refer to [ENTRY-FIELD](#) for details about properties, styles and events of this control.

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
action	Specifies the value for the <i>Action</i> property. You can choose between: None CUT COPY PASTE DELETE UNDO REDO SELECT ALL
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
alignment	NONE... no alignment style is generated LEFT... <i>Left</i> style is generated RIGHT... <i>Right</i> style is generated CENTER... <i>Center</i> style is generated HTML...no alignment style is generated
auto	TRUE... The <i>Auto</i> style is generated FALSE... The <i>Auto</i> style is not generated
autodecimal	Specifies the value for the <i>Auto-Decimal</i> property

background-color Opens a dialog that allows the user to choose the control background color.



bitmap Opens a dialog box that allows the user to select an image file to load into the control



bitmap-disabled Specifies the value for the *Bitmap-Disabled* property

bitmap-hint Specifies the value for the *Bitmap-Hint* property

bitmap-number Specifies the value for the *Bitmap-Number* property

bitmap-rollover Specifies the value for the *Bitmap-Rollover* property

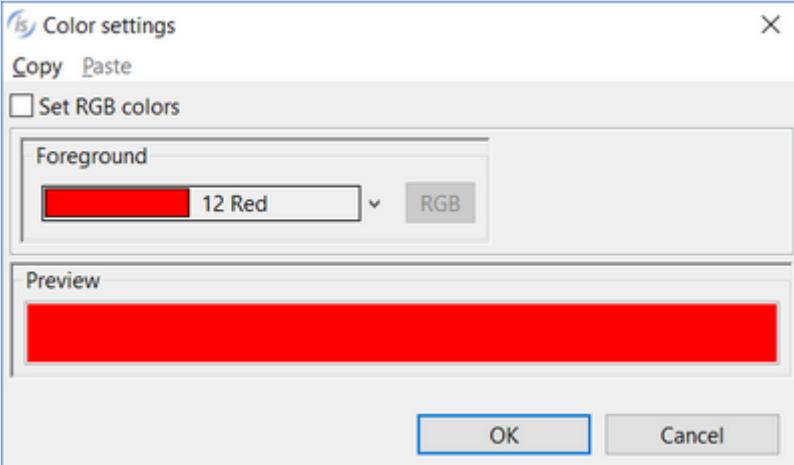
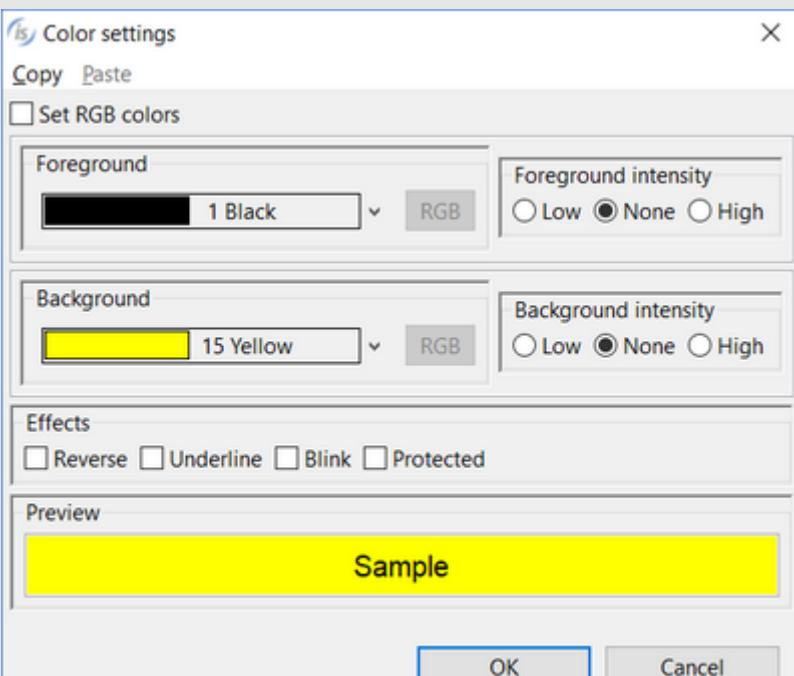
bitmap-trailing-disabled Specifies the value for the *Bitmap-Trailing-Disabled* property

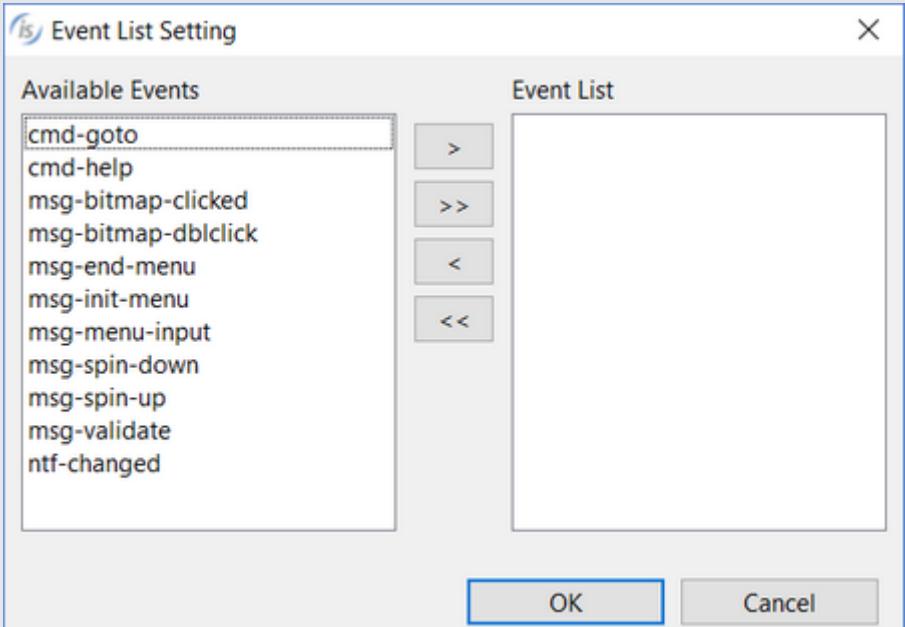
bitmap-trailing-hint Specifies the value for the *Bitmap-Trailing-Hint* property

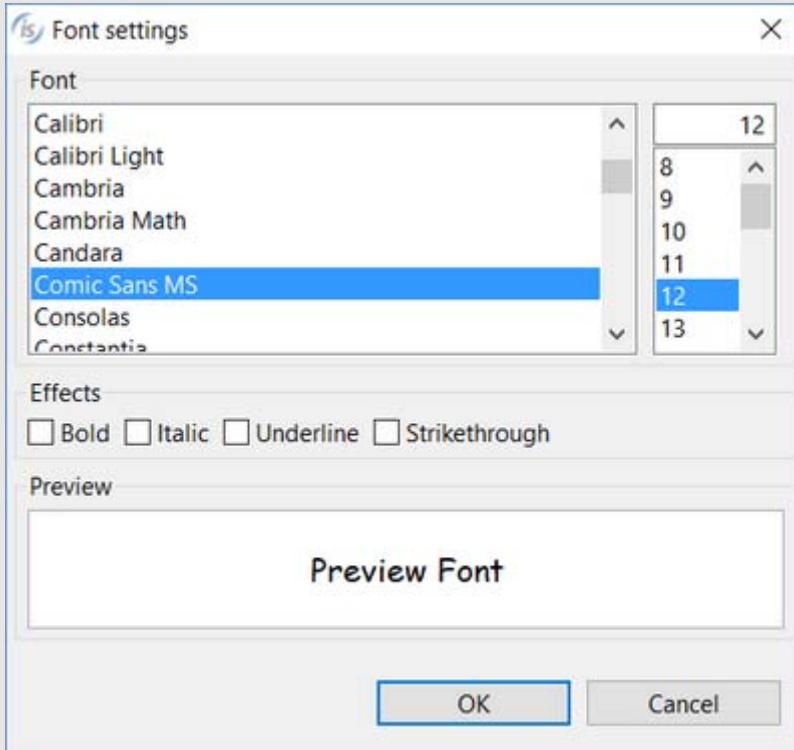
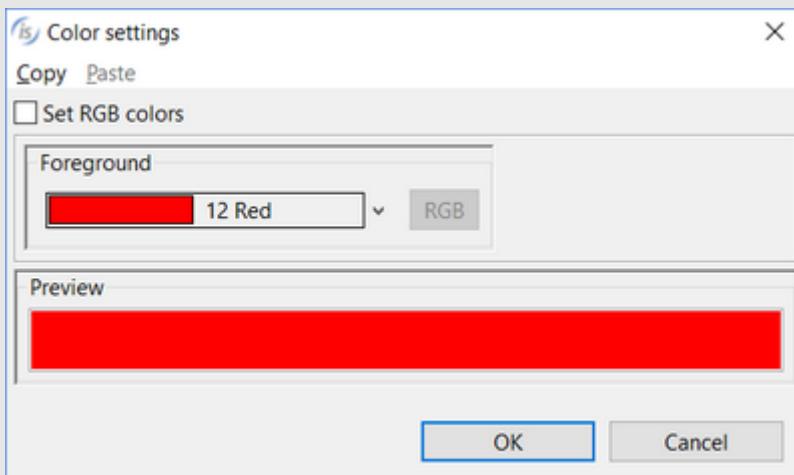
bitmap-trailing-number Specifies the value for the *Bitmap-Trailing-Number* property

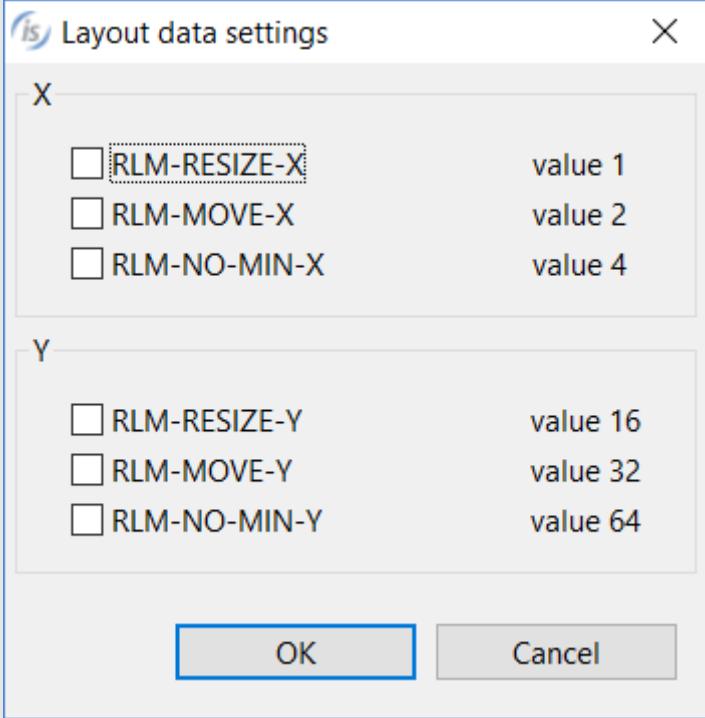
bitmap-trailing-rollover Specifies the value for the *Bitmap-Trailing-Rollover* property

border Allows the user to set one of the following three styles:
3-D
BOXED
NO-BOX

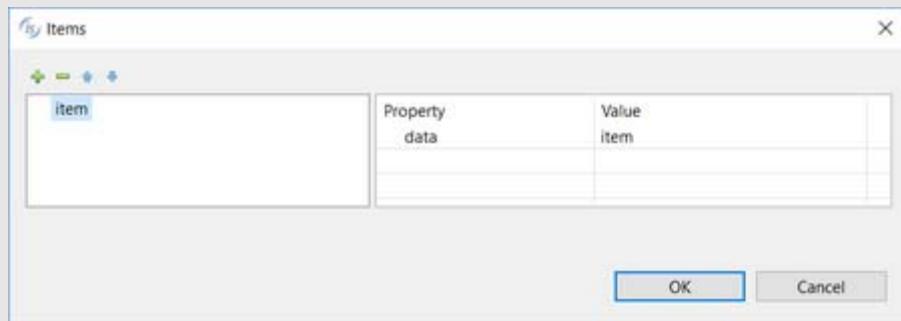
border-color	Opens a dialog that allows the user to choose the control border color.
	 A screenshot of the 'Color settings' dialog box. It has tabs for 'Copy' and 'Paste'. A checked checkbox 'Set RGB colors' is present. Under 'Foreground', there is a color swatch labeled '12 Red' with an 'RGB' button next to it. Below it is a 'Preview' section showing a red horizontal bar. At the bottom are 'OK' and 'Cancel' buttons.
case	NONE... Neither the <i>Upper</i> nor <i>Lower</i> styles are generated UPPER... <i>Upper</i> style is generated LOWER... <i>Lower</i> style is generated
color	Opens a dialog that allows the user to choose the control color.
	 A screenshot of the 'Color settings' dialog box. It has tabs for 'Copy' and 'Paste'. An unchecked checkbox 'Set RGB colors' is present. Under 'Foreground', there is a color swatch labeled '1 Black' with an 'RGB' button next to it, and a 'Foreground intensity' section with radio buttons for 'Low', 'None' (selected), and 'High'. Under 'Background', there is a color swatch labeled '15 Yellow' with an 'RGB' button next to it, and a 'Background intensity' section with radio buttons for 'Low', 'None' (selected), and 'High'. Under 'Effects', there are checkboxes for 'Reverse', 'Underline', 'Blink', and 'Protected'. Below these is a 'Preview' section showing a yellow horizontal bar with the word 'Sample' in black. At the bottom are 'OK' and 'Cancel' buttons.
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.

cursor	Specifies the value for the <i>Cursor</i> property
cursor-col	Specifies the value for the <i>Cursor-Col</i> property
cursor-row	Specifies the value for the <i>Cursor-Row</i> property
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor <i>Permanent</i> styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. The "Available Events" section contains a list of event names: cmd-goto, cmd-help, msg-bitmap-clicked, msg-bitmap-dblclick, msg-end-menu, msg-init-menu, msg-menu-input, msg-spin-down, msg-spin-up, msg-validate, and ntf-changed. To the right of this list are four buttons: >, >>, <, and <<. Below the "Event List" section are two buttons: "OK" and "Cancel".</p>	
exclude event list	NONE... The <i>Exclude-Event-List</i> property is not generated. 0... <i>Exclude-Event-List=0</i> is generated. 1... <i>Exclude-Event-List=1</i> is generated.
fill-char	Specifies the value for the <i>Fill-Char</i> property

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical size selector with values 8, 9, 10, 11, 12, and 13. '12' is selected. Below the font list are sections for 'Effects' (Bold, Italic, Underline, Strikethrough) and 'Preview' (a box labeled 'Preview Font'). At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It includes a 'Copy' and 'Paste' button, a checkbox for 'Set RGB colors', and a 'Foreground' color picker set to '12 Red'. A 'RGB' button is next to the picker. Below it is a 'Preview' section with a red rectangular preview area. At the bottom are 'OK' and 'Cancel' buttons.</p>
format-picture	Specifies the value for the <i>Pic</i> property
format-string	Specifies the value for the <i>Format-String</i> property
format-type	Specifies the value for the Format-Type property 0... MASK 1... NUMBER 2... DATE

height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE... The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
id	Specifies the control id. This property is set automatically when the control is drawn.
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value

lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
max-height	Specifies the control maximum height as expressed in cells
max-lines	Specifies the value for the <i>Max-Lines</i> property
max-text	Specifies the value for the <i>Max-Text</i> property
max-val	Specifies the value for the <i>Max-Val</i> property
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-val	Specifies the value for the <i>Min-Val</i> property
min-width	Specifies the control minimum width as expressed in cells
no-autosel	TRUE... The <i>No-Autosel</i> style is generated FALSE... The <i>No-Autosel</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
notify change	TRUE...The <i>Notify-Change</i> style is generated FALSE...The <i>Notify-Change</i> style is not generated
notify-change delay	Specifies the value for the <i>Notify-Change-Delay</i> property
no-wrap	TRUE...The <i>No-Wrap</i> style is generated FALSE...The <i>No-Wrap</i> style is not generated
numeric	TRUE...The <i>Numeric</i> style is generated FALSE...The <i>Numeric</i> style is not generated
placeholder	Specifies the value for the <i>Placeholder</i> property
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
proposal	Opens a dialog that allows the user to declare proposal items
proposal delay	Specifies the value for the <i>Proposal-Delay</i> property



proposal min text	Specifies the value for the <i>Proposal-Min-Text</i> property
proposals-unsorted	TRUE...The <i>Proposals-Unsorted</i> style is generated FALSE...The <i>Proposals-Unsorted</i> style is not generated
read-only	TRUE...The <i>Read-Only</i> style is generated FALSE...The <i>Read-Only</i> style is not generated
required	TRUE...The <i>Required</i> style is generated FALSE...The <i>Required</i> style is not generated
secure	TRUE...The <i>Secure</i> style is generated FALSE...The <i>Secure</i> style is not generated
selection-start	Specifies the value for the <i>Selection-Start</i> property
selection-start-col	Specifies the value for the <i>Selection-Start-Col</i> property
selection-start-row	Specifies the value for the <i>Selection-Start-Row</i> property
selection-text	Specifies the value for the <i>Selection-Text</i> property
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
style	Allows the user to choose the type of field between: SINGLELINE (default) MULTILINE VSCROLL VSCROLL-BAR SPINNER AUTO-SPIN
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
spell checking	Sets the value for the <i>Spell-Checking</i> property. Note that additional jar files must be added to the Project's Classpath in order to activate this feature at run time. Refer to Spell-Checking for details about the required additional items.
text orientation	0: NONE... <i>Text-Orientation</i> is not generated 1: LEFT-TO-RIGHT... <i>Text-Orientation=1</i> is generated 2: RIGHT-TO-LEFT... <i>Text-Orientation=2</i> is generated
use RETURN	TRUE... The <i>Use-Return</i> style is generated FALSE... The <i>Use-Return</i> style is not generated
use TAB	TRUE... The <i>Use-Tab</i> style is generated FALSE... The <i>Use-Tab</i> style is not generated

validation error message	Specifies the value for the <i>Validation-Errormsg</i> property
validation options	0: None 1: Case Insensitive 2: Left Trimmed 3: Case Insensitive + Left Trimmed 4: Right Trimmed 5: Case Insensitive + Right Trimmed 6: Left Trimmed + Right Trimmed 7: Case Insensitive + Left Trimmed+ Right Trimmed
validation regexp	Specifies the value for the <i>Validation-Regexp</i> property
value	Specifies the value for the <i>Value</i> property
value multiple	TRUE... The <i>Multiple</i> keyword is generated along with <i>Value</i> FALSE... The <i>Multiple</i> keyword is not generated along with <i>Value</i>
value picture	Specifies the value for the <i>PIC</i> property
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
visible proposals count	Specifies the value for the <i>Visible-Proposals-Count</i> property
width-in-cells	TRUE... The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-bitmap-clicked event	Allows the user to create a paragraph to handle the MSG-BITMAP-CLICKED event in the Procedure Division
msg-bitmap-dblclick event	Allows the user to create a paragraph to handle the MSG-BITMAP-DBLCLICK event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
ntf-changed event	Allows the user to create a paragraph to handle the NTF-CHANGED event in the Procedure Division
other event	Allows the user to create a custom paragraph

Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-changed exception	Allows the user to create a paragraph to handle the NTF-CHANGED event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Variables	
autodecimal variable	Numeric variable that hosts the value for the <i>Auto-Decimal</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
cursor-col variable	Numeric variable that hosts the cursor column value
cursor-row variable	Numeric variable that hosts the cursor row value
cursor variable	Numeric variable that hosts the cursor value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the hint value.
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value

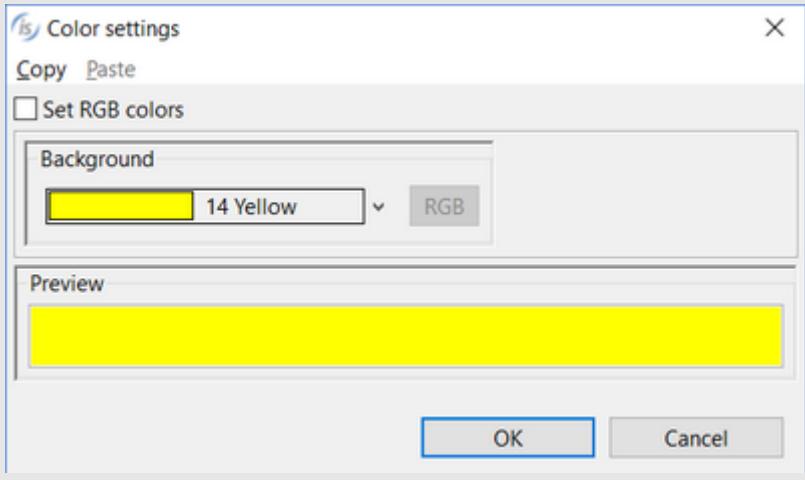
max-height variable	Numeric variable that hosts the maximum height
max-lines variable	Numeric variable that hosts the value for the <i>Max-Lines</i> property
max-text variable	Numeric variable that hosts the value for the <i>Max-Text</i> property
max-val variable	Numeric variable that hosts the value for the <i>Max-Val</i> property
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-val variable	Numeric variable that hosts the value for the <i>Min-Val</i> property
min-width variable	Numeric variable that hosts the minimum width
notify-change delay variable	Numeric variable that hosts the value for the <i>Notify-Change-Delay</i> property
placeholder variable	Alphanumeric variable that hosts the value of the <i>Placeholder</i> property
proposal delay variable	Numeric variable that hosts the value for the <i>Proposal-Delay</i> property
proposal min text variable	Numeric variable that hosts the value for the <i>Proposal-Min-Text</i> property
selection-start-col variable	Numeric variable that hosts the value for the <i>Selection-Start-Col</i> property
selection-start-row variable	Numeric variable that hosts the value for the <i>Selection-Start-Row</i> property
selection-start variable	Numeric variable that hosts the value for the <i>Selection-Start</i> property
size variable	Numeric variable that hosts the size value
validation error message variable	Alphanumeric variable that hosts the value of the <i>Validation-Errmsg</i> property
validation options variable	Numeric variable that hosts the value of the <i>Validation-Opts</i> property
validation regexp variable	Alphanumeric variable that hosts the value of the <i>Validation-Regexp</i> property
value variable	Alphanumeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

FRAME

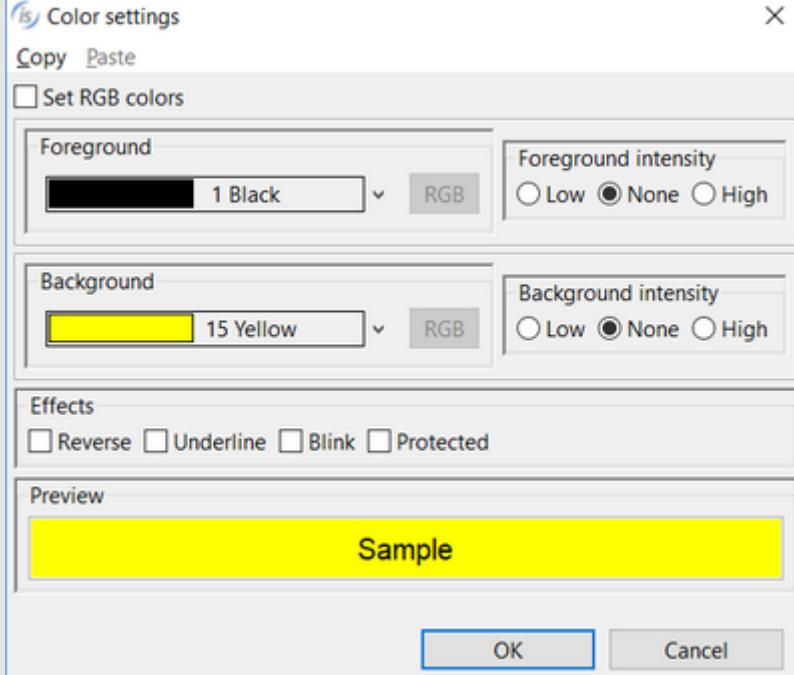
Refer to [FRAME](#) for details about properties, styles and events of this control.

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn

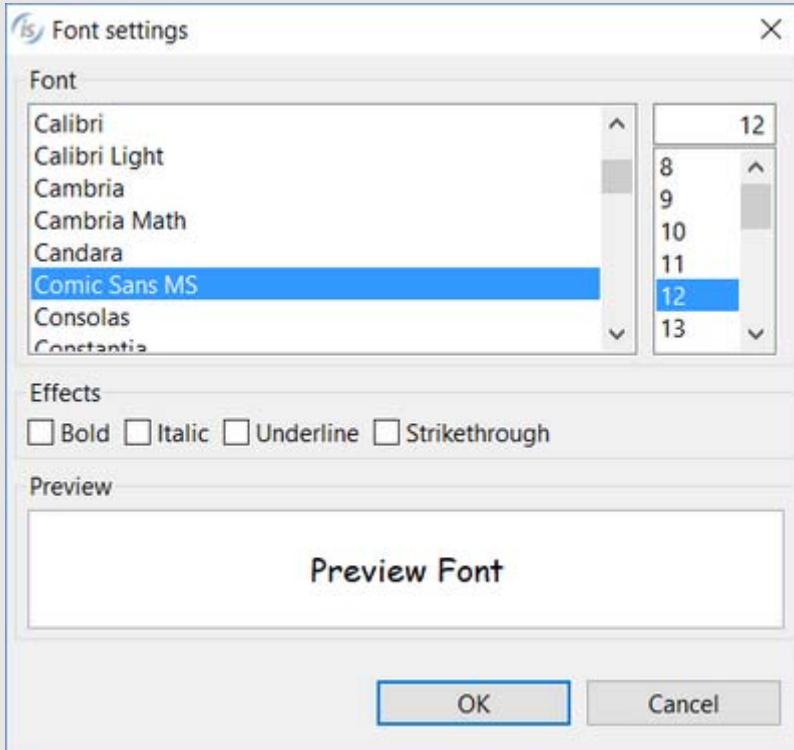
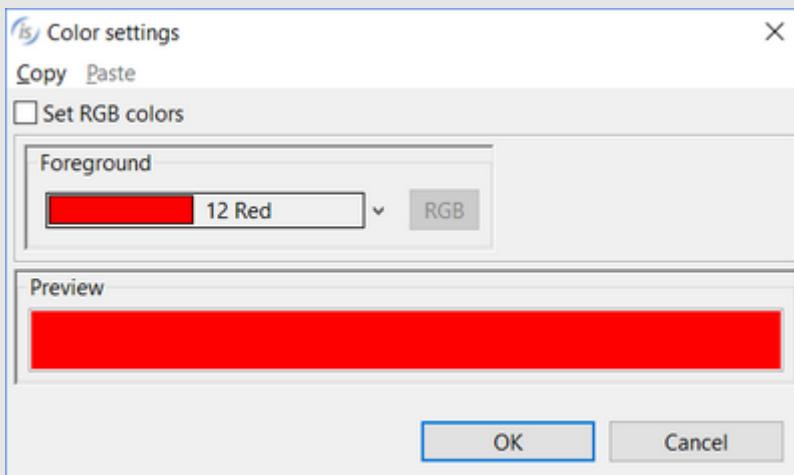
3D Style	Allows you to choose the frame type. The choices are as follows: <ul style="list-style-type: none"> • Engraved • Lowered • Normal • Raised • Rimmed
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
alternate	TRUE... The <i>Alternate</i> style is generated FALSE... The <i>Alternate</i> style is not generated
background-color	Opens a dialog that allows the user to choose the control background color.

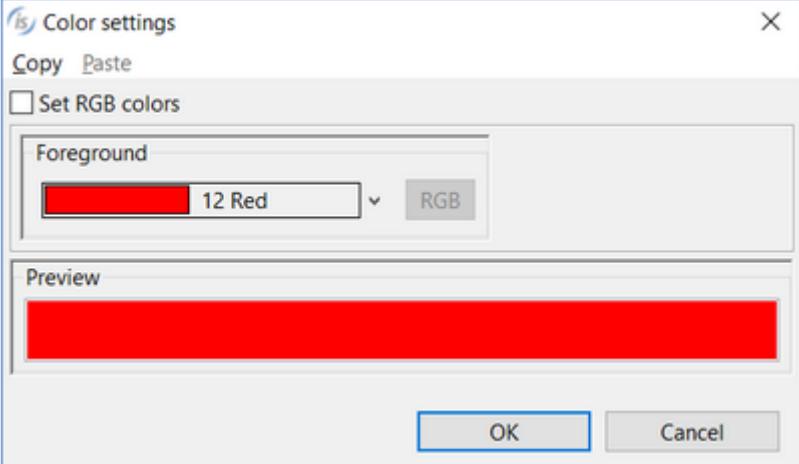
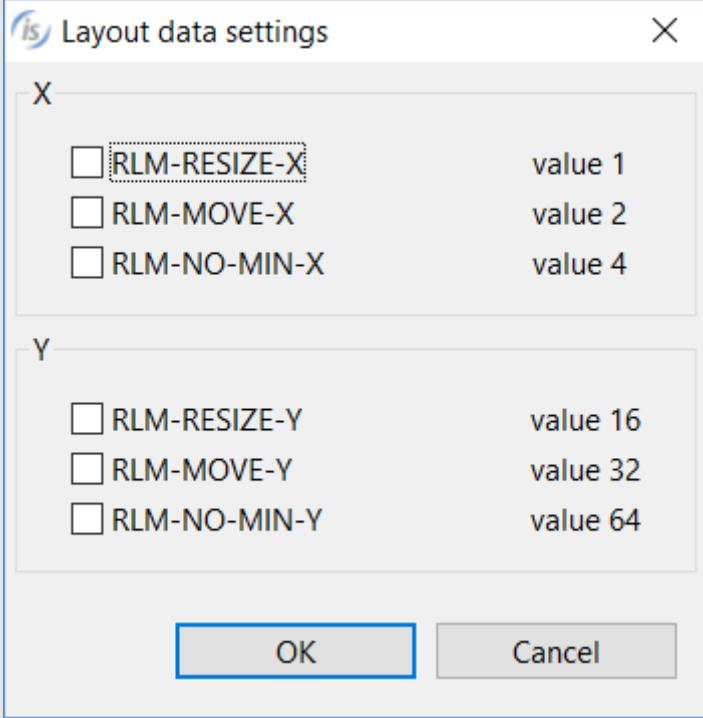


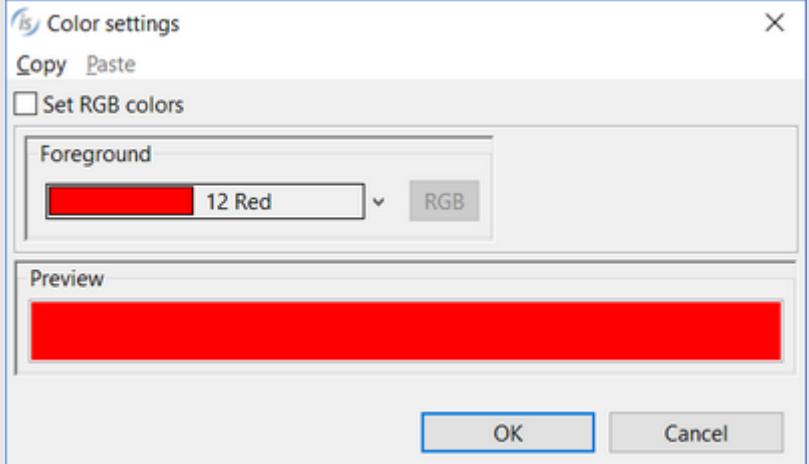
The screenshot shows the 'Color settings' dialog box. At the top, there are 'Copy' and 'Paste' buttons. Below them is a checkbox labeled 'Set RGB colors'. Underneath is a 'Background' section containing a color swatch labeled '14 Yellow' and an 'RGB' button. A 'Preview' section shows a larger yellow rectangle. At the bottom right are 'OK' and 'Cancel' buttons.

color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specifies the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor <i>Permanent</i> styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

fill color	Opens a dialog to retrieve the value for the <i>Fill-Color</i> property
	<p>The screenshot shows the 'Color settings' dialog box. At the top, there are 'Copy' and 'Paste' buttons and a checkbox for 'Set RGB colors'. Below this is a 'Background' section containing a color swatch labeled '14 Yellow' and an 'RGB' button. A preview window below shows a yellow bar. At the bottom are 'OK' and 'Cancel' buttons.</p>
fill color2	Opens a dialog to retrieve the value for the <i>Fill-Color2</i> property
	<p>The screenshot shows the 'Color settings' dialog box. At the top, there are 'Copy' and 'Paste' buttons and a checkbox for 'Set RGB colors'. Below this is a 'Foreground' section containing a color swatch labeled '12 Red' and an 'RGB' button. A preview window below shows a red bar. At the bottom are 'OK' and 'Cancel' buttons.</p>

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right is a font size dropdown menu with options 8, 9, 10, 11, 12, and 13, where '12' is selected. Below the font list are checkboxes for Bold, Italic, Underline, and Strikethrough. A preview window shows the text 'Preview Font' in the selected font and size. At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a checkbox 'Set RGB colors' which is unchecked. Below it is a 'Foreground' color picker set to '12 Red'. A preview window shows a red rectangle. At the bottom are 'OK' and 'Cancel' buttons.</p>
full height	TRUE...The <i>Full-Height</i> style is generated FALSE...The <i>Full-Height</i> style is not generated
heavy	TRUE...The <i>Heavy</i> style is generated FALSE...The <i>Heavy</i> style is not generated
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE... The <i>Height-In-Cells</i> style is not generated

help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
high color	Opens a dialog to retrieve the value for the <i>High-Color</i> property
	 <p>The screenshot shows the 'Color settings' dialog box. At the top, there are 'Copy' and 'Paste' buttons and a checkbox for 'Set RGB colors'. Below this is a 'Foreground' section with a color swatch labeled '12 Red' and an 'RGB' button. A 'Preview' section shows a red rectangle. At the bottom are 'OK' and 'Cancel' buttons.</p>
id	Specifies the control id. This property is set automatically when the control is drawn.
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	 <p>The screenshot shows the 'Layout data settings' dialog box. It has two main sections: 'X' and 'Y'. Each section contains three checkboxes: 'RLM-RESIZE-X/Y', 'RLM-MOVE-X/Y', and 'RLM-NO-MIN-X/Y', each associated with a numerical value: 'value 1', 'value 2', 'value 4' for the X section, and 'value 16', 'value 32', 'value 64' for the Y section. At the bottom are 'OK' and 'Cancel' buttons.</p>
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn

line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	<p>DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting</p> <p>None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value</p> <p><i>CELLS</i>... <i>CELLS</i> is generated after the <i>Lines</i> value</p> <p><i>PIXELS</i>... <i>PIXELS</i> is generated after the <i>Lines</i> value</p>
lock	<p>TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse.</p> <p>FALSE...You can move the control on the Screen Designer by dragging it with the mouse</p>
low-color	Opens a dialog to retrieve the value for the <i>Low-Color</i> property
	
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
no-tab	<p>TRUE...The <i>No-Tab</i> style is generated</p> <p>FALSE...The <i>No-Tab</i> style is not generated</p>
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn

size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
title	Specifies the value for the <i>Title</i> property
title-position	0: UNALIGNED 1: TOP LEFT 2: TOP CENTER 3: TOP RIGHT 4: BOTTOM LEFT 5: BOTTOM CENTER 6: BOTTOM RIGHT 7: CENTERED VERTICALLY AND HORIZONTALLY
transparent	TRUE...The <i>Transparent</i> style is generated FALSE...The <i>Transparent</i> style is not generated
very heavy	TRUE...The <i>Very-Heavy</i> style is generated FALSE...The <i>Very-Heavy</i> style is generated
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
No Events available.	
Exceptions	
No Exceptions available.	
Procedures	
No Procedures available.	
Variables	
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
fill color variable	Numeric variable that hosts the value for the <i>Fill-Color</i> property
fill color2 variable	Numeric variable that hosts the value for the <i>Fill-Color2</i> property

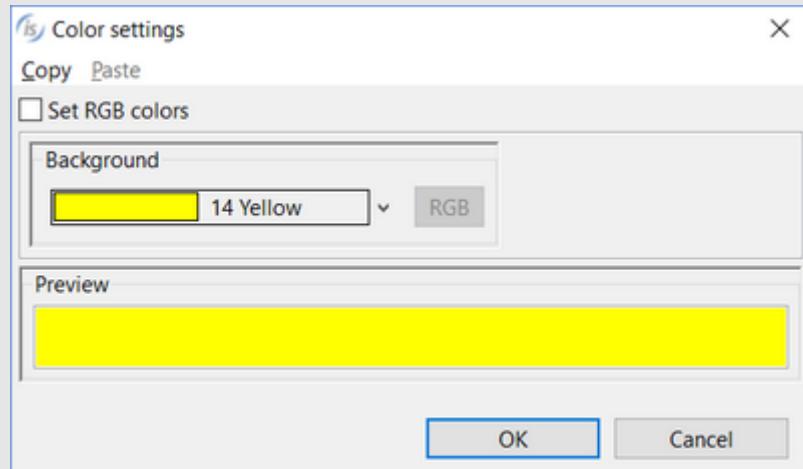
fill percent	Numeric variable that hosts the value for the <i>Fill-Percent</i> property
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the hint value.
high color variable	Numeric variable that hosts the value for the <i>High-Color</i> property
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
low color variable	Numeric variable that hosts the value for the <i>Low-Color</i> property
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
title-position variable	Alphanumeric variable that hosts the value for the <i>Title-Position</i> property
title variable	Alphanumeric variable that hosts the value for the <i>Title</i> property
visible variable	Numeric variable that hosts the visible state

GRID

Refer to [GRID](#) for details about properties, styles and events of this control.

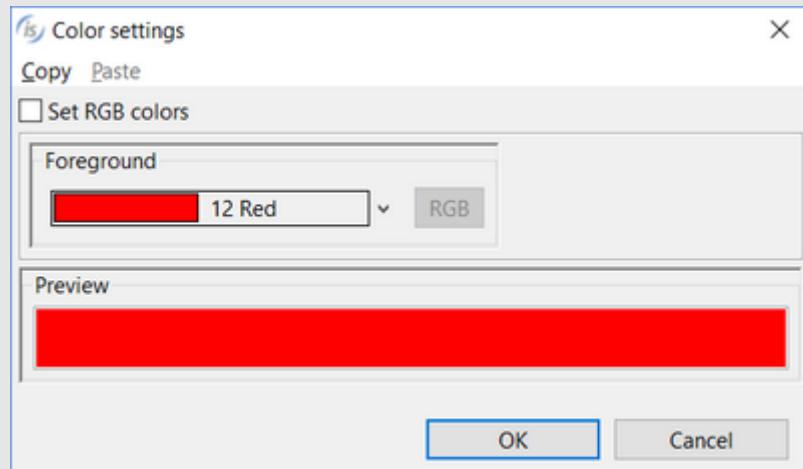
Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
action	Specifies the value for the <i>Action</i> property. You can choose between: None FIRST-PAGE LAST-PAGE CURRENT-PAGE
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
adjustable-columns	TRUE... The <i>Adjustable-Columns</i> style is generated FALSE... The <i>Adjustable-Columns</i> style is not generated
adjustable-rows	TRUE... The <i>Adjustable-Rows</i> style is generated FALSE... The <i>Adjustable-Rows</i> style is not generated

`background-color` Opens a dialog that allows the user to choose the control background color.

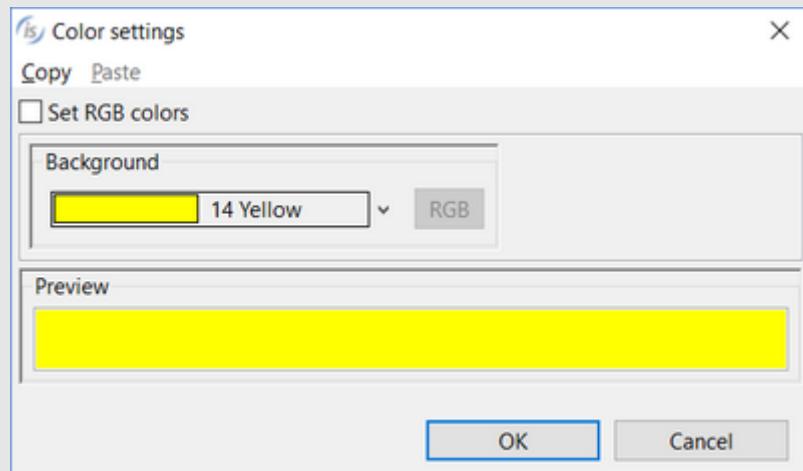


`border` Allows the user to set one of the following three styles:
3-D
BOXED
NO-BOX

`border-color` Opens a dialog that allows the user to choose the control border color.

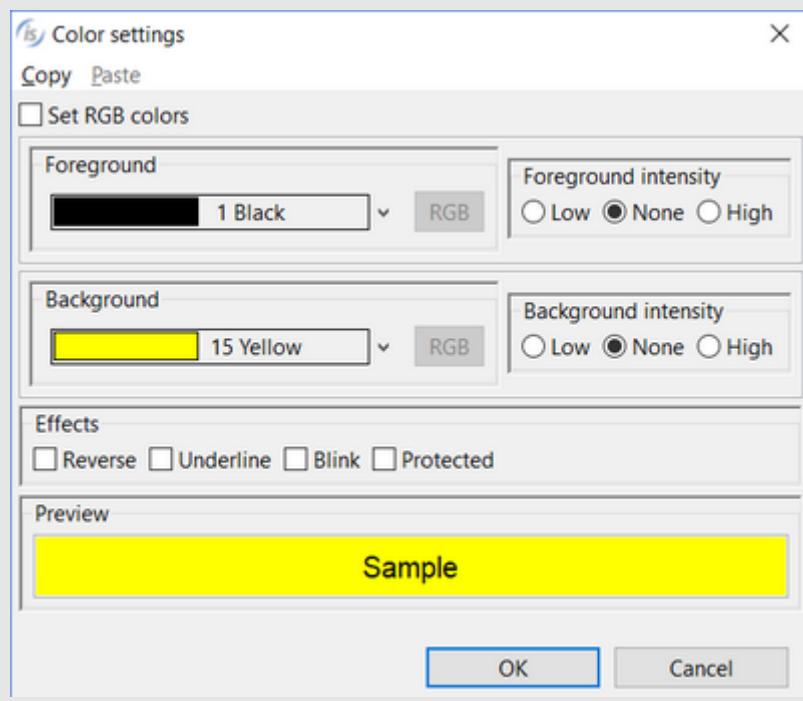


cell entry background color Opens a dialog that allows to set the *Cell-Entry-Background-Color* property.

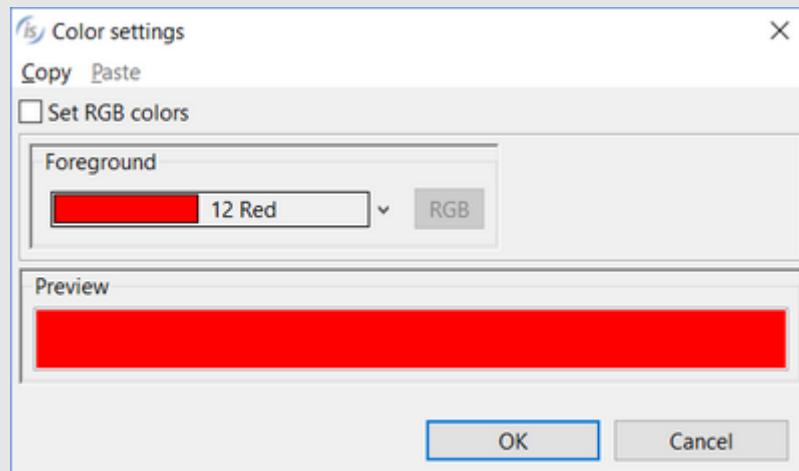


cell entry color

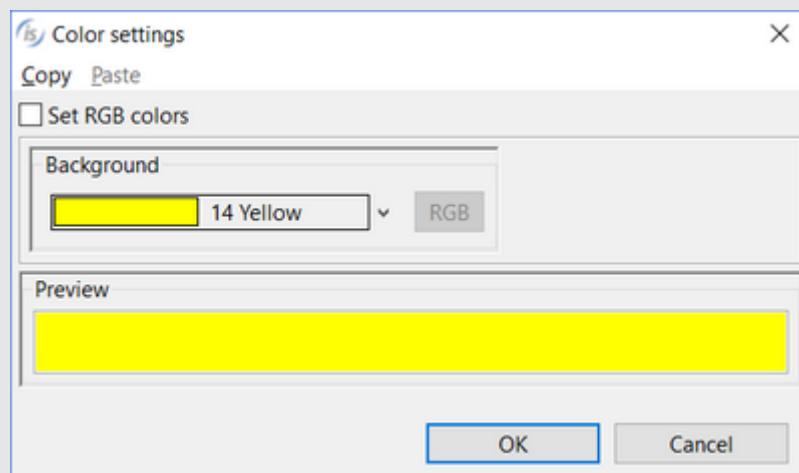
Opens a dialog that allows to set the *Cell-Entry-Color* property.

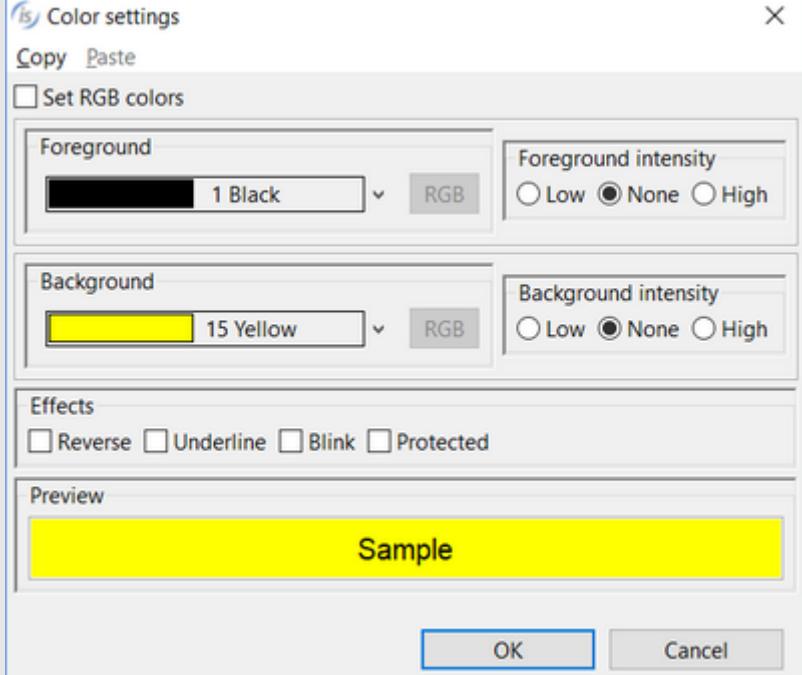
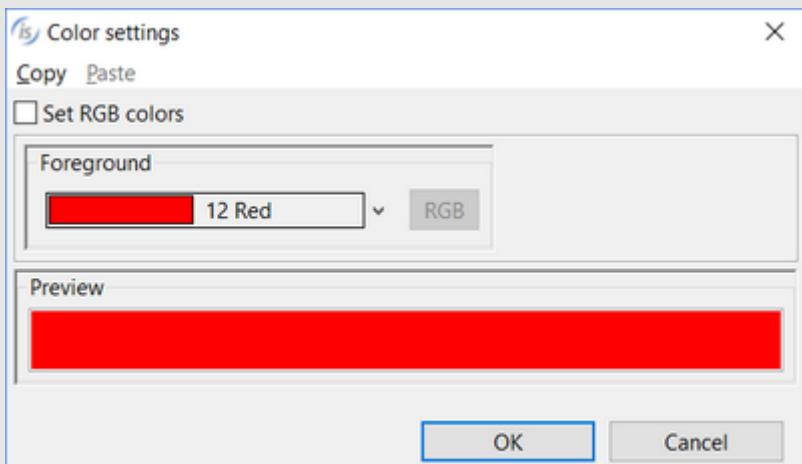


cell entry foreground color Opens a dialog that allows to set the *Cell-Entry-Foreground-Color* property.



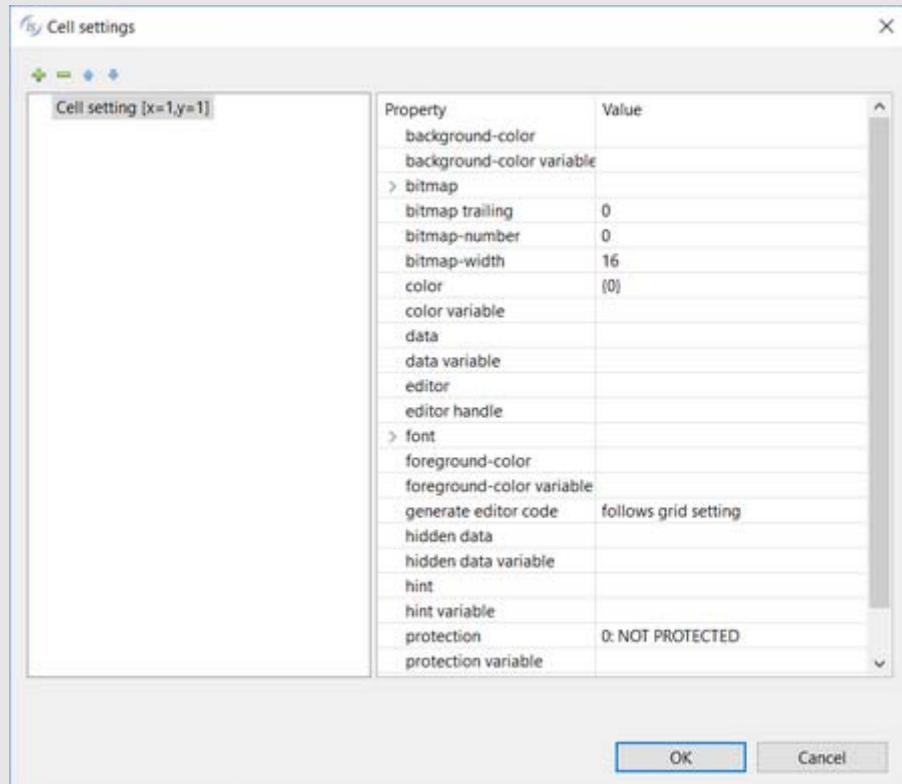
cell selected background color Opens a dialog that allows to set the *Cell-Selected-Background-Color* property.



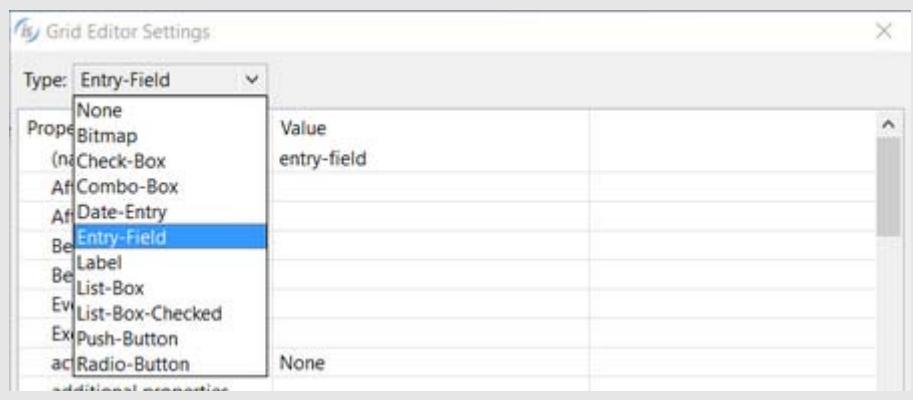
cell selected color	Opens a dialog that allows to set the <i>Cell-Selected-Color</i> property.
 The dialog is titled "Color settings". It has sections for "Foreground" (color swatch 1 Black, RGB button) and "Background" (color swatch 15 Yellow, RGB button). It also includes "Effects" (checkboxes for Reverse, Underline, Blink, Protected) and a "Preview" section showing a yellow bar labeled "Sample". There are "OK" and "Cancel" buttons at the bottom. <p>Color settings</p> <p>Copy Paste</p> <p><input type="checkbox"/> Set RGB colors</p> <p>Foreground</p> <p>1 Black <input type="button" value="RGB"/></p> <p>Background</p> <p>15 Yellow <input type="button" value="RGB"/></p> <p>Effects</p> <p><input type="checkbox"/> Reverse <input type="checkbox"/> Underline <input type="checkbox"/> Blink <input type="checkbox"/> Protected</p> <p>Preview</p> <p>Sample</p> <p>OK Cancel</p>	
cell selected foreground color	Opens a dialog that allows to set the <i>Cell-Selected-Foreground-Color</i> property.
 The dialog is titled "Color settings". It has a "Foreground" section (color swatch 12 Red, RGB button) and a "Preview" section showing a red bar. There are "OK" and "Cancel" buttons at the bottom. <p>Color settings</p> <p>Copy Paste</p> <p><input type="checkbox"/> Set RGB colors</p> <p>Foreground</p> <p>12 Red <input type="button" value="RGB"/></p> <p>Preview</p> <p>OK Cancel</p>	

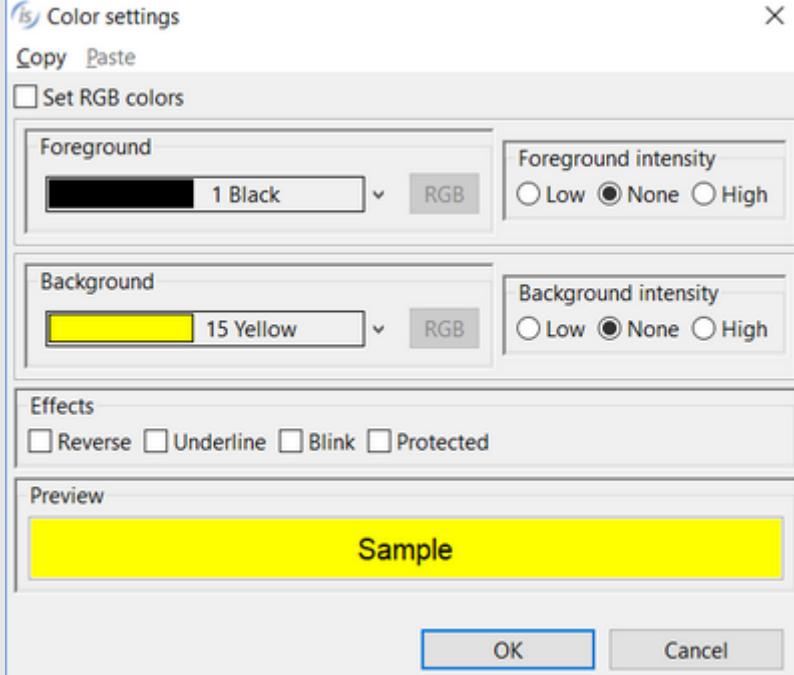
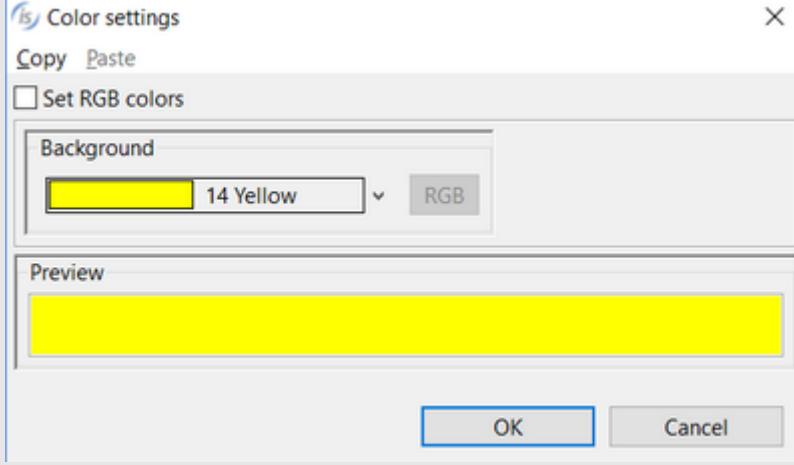
cell settings

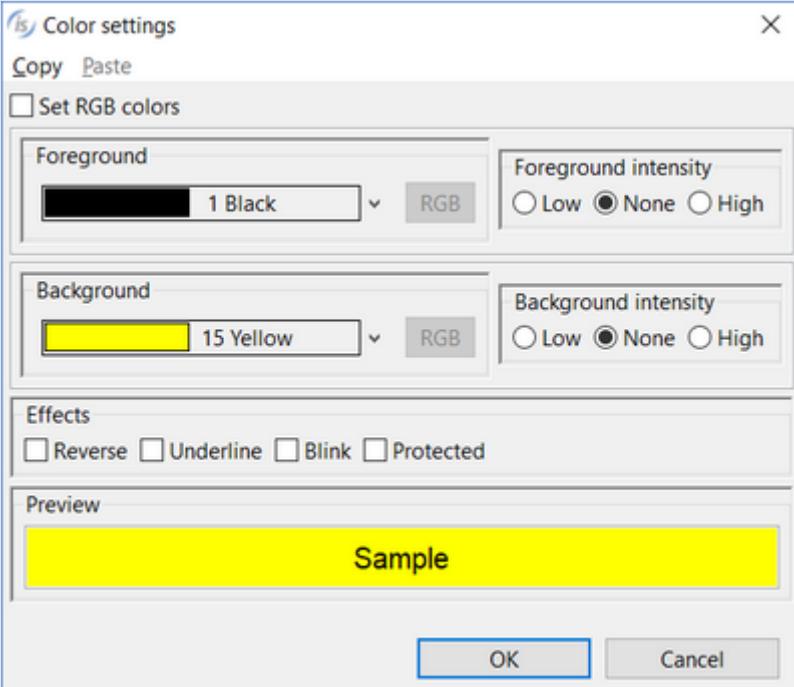
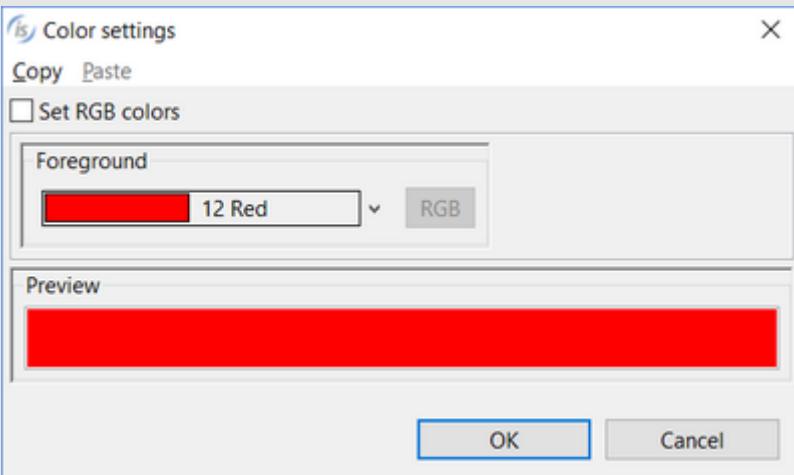
Opens a dialog that allows the user to set cells content



The "editor" property pops up a new dialog in which you can choose the graphical control that must appear in the cell during the editing.

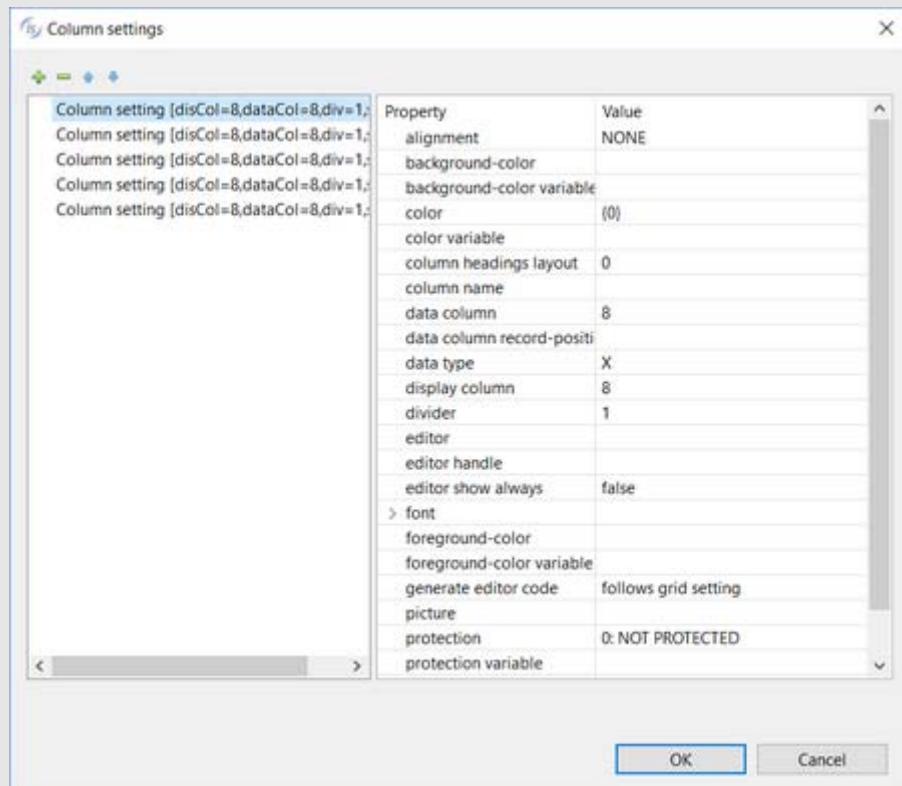


color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column headings	TRUE... The <i>Column-Headings</i> style is generated FALSE... The <i>Column-Headings</i> style is not generated
column headings height	Specifies the value for the <i>Column-Headings-Height</i> property.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
column selected background color	Opens a dialog that allows to set the <i>Column-Selected-Background-Color</i> property.
	

column selected color	Opens a dialog that allows to set the <i>Column-Selected-Color</i> property.
 The dialog is titled "Color settings". It has sections for "Foreground" (color swatch 1 Black, RGB button) and "Background" (color swatch 15 Yellow, RGB button). Under "Effects", there are checkboxes for Reverse, Underline, Blink, and Protected. A "Preview" section shows a yellow bar labeled "Sample". Buttons at the bottom are "OK" and "Cancel".	
column selected foreground color	Opens a dialog that allows to set the <i>Column-Selected-Foreground-Color</i> property.
 The dialog is titled "Color settings". It has a "Foreground" section (color swatch 12 Red, RGB button) and a "Preview" section showing a red bar. Buttons at the bottom are "OK" and "Cancel".	

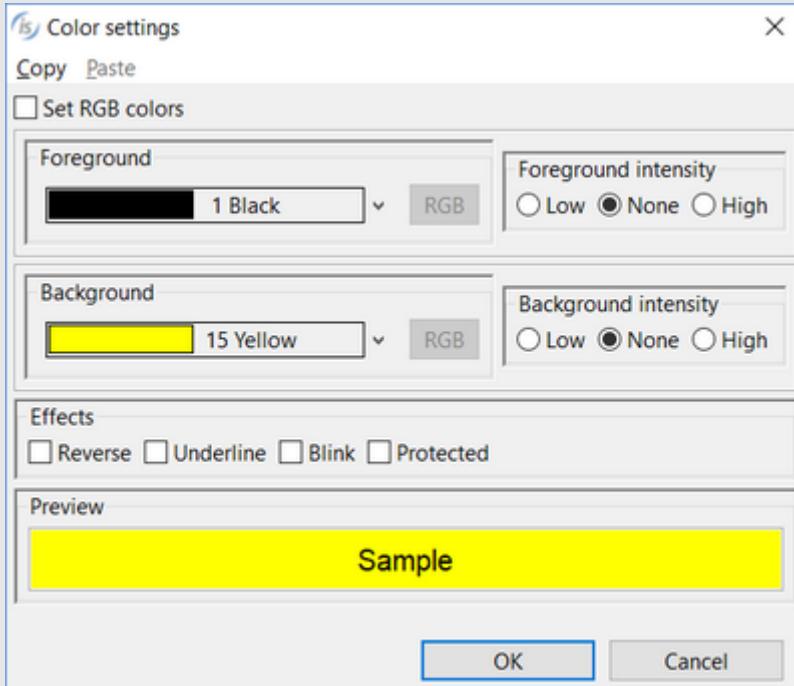
column settings

Opens a dialog that allows the user to define columns



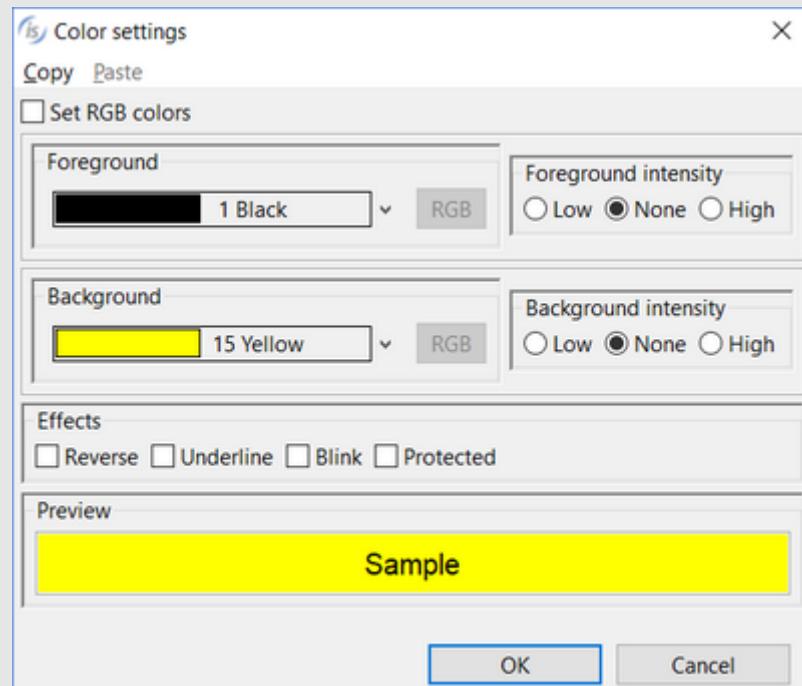
css-base-style-name
css-style-name

Specify the CSS style associated with the control. It works only in a Web Direct 2 environment.
See [Customize the EIS WD2 Layout using CSS](#) for more information.

cursor color	Opens a dialog that allows the user to choose the cursor color.
	
cursor frame width	Specifies the value for the <i>Cursor-Frame-Width</i> property
cursor X	Specifies the value for the <i>Cursor-X</i> property
cursor Y	Specifies the value for the <i>Cursor-Y</i> property
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor Permanent styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated

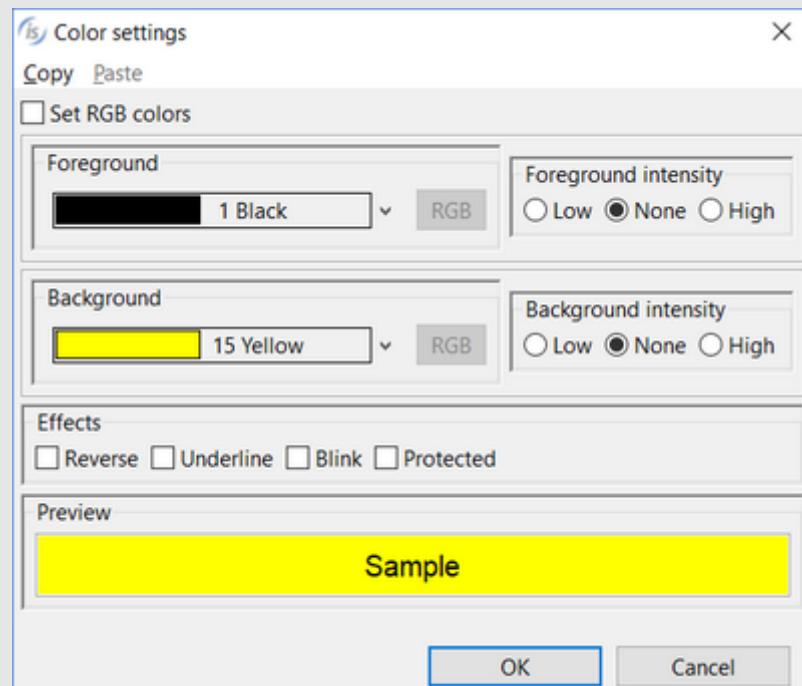
divider color

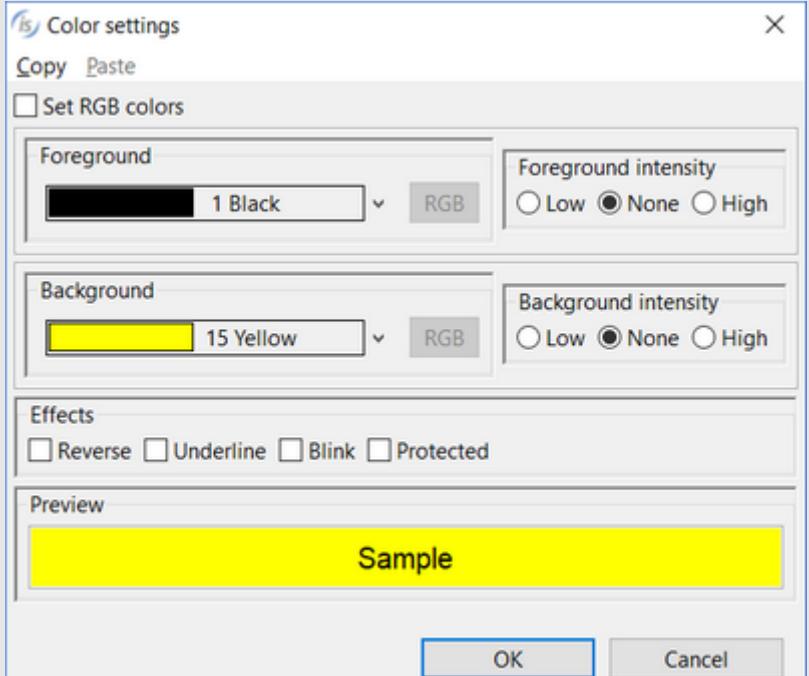
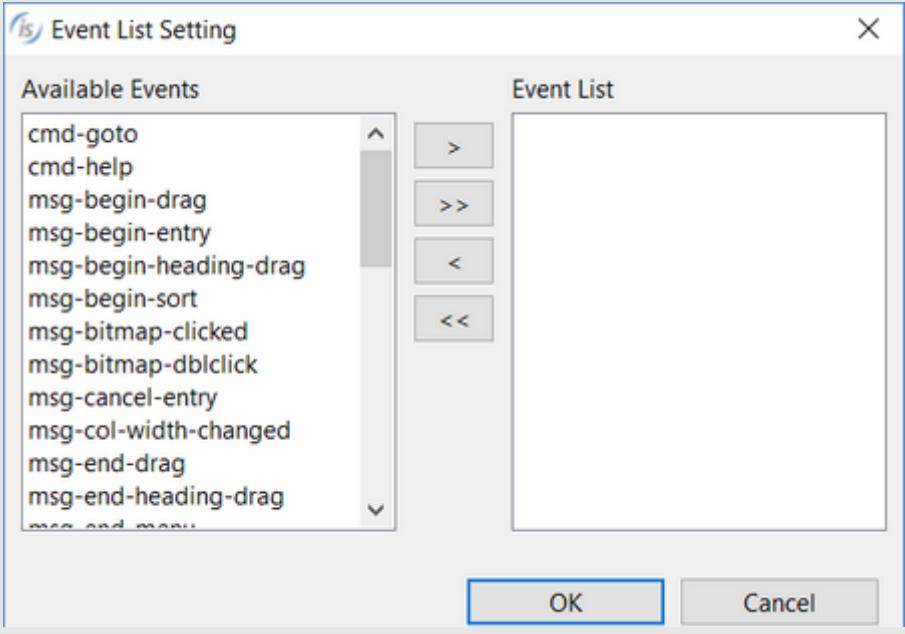
Opens a dialog to retrieve the value for the *Divider-Color* property



drag color

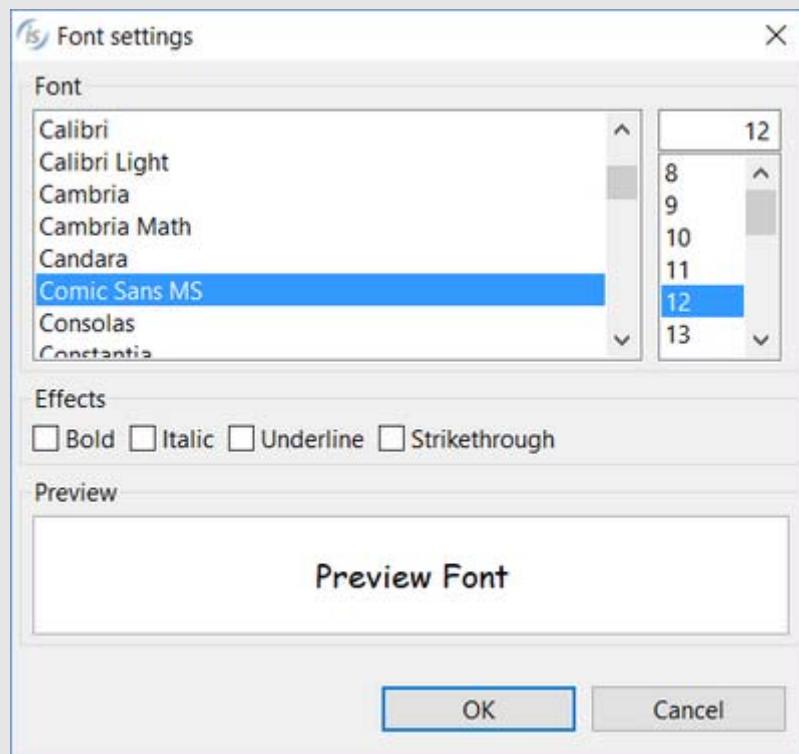
Opens a dialog to retrieve the value for the *Drag-Color* property

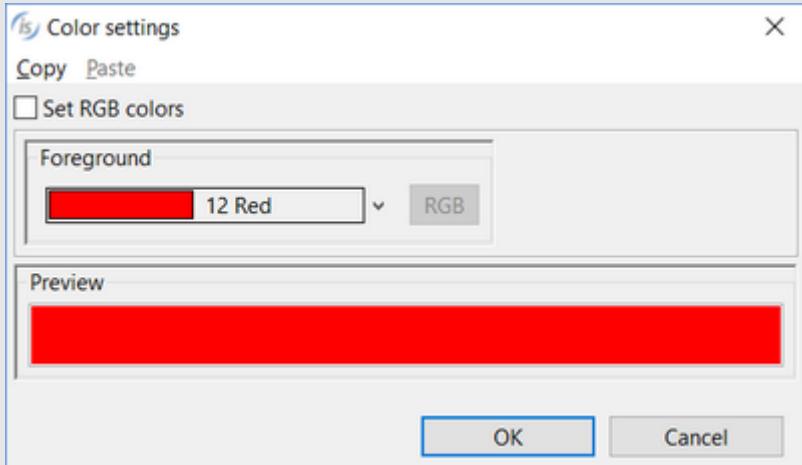


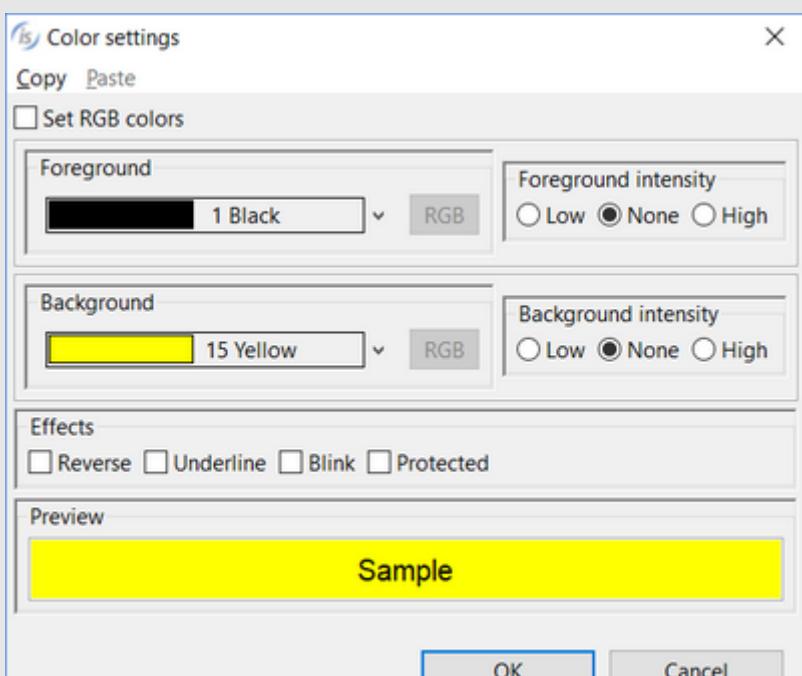
enabled	NONE...The <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
end color	Opens a dialog to retrieve the value for the <i>End-Color</i> property
	
event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	

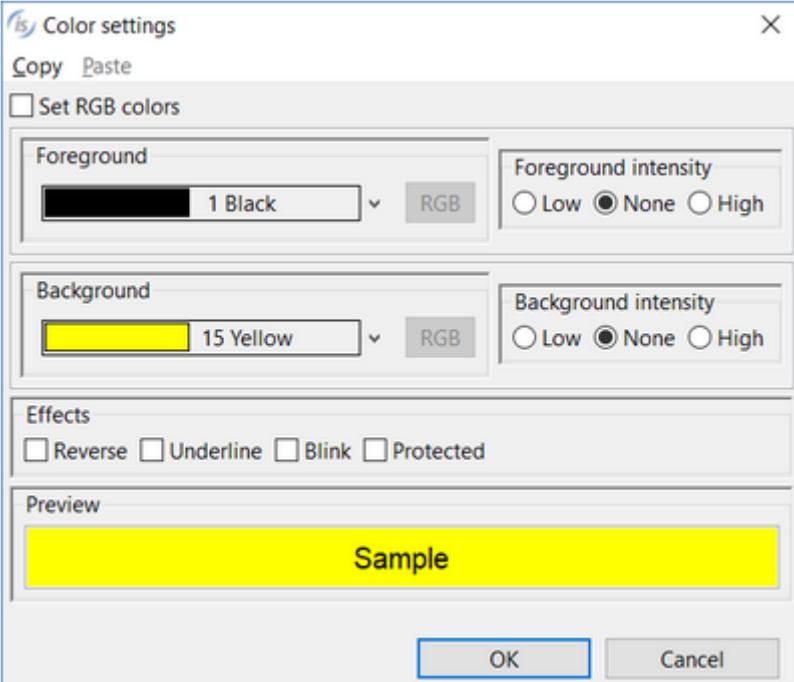
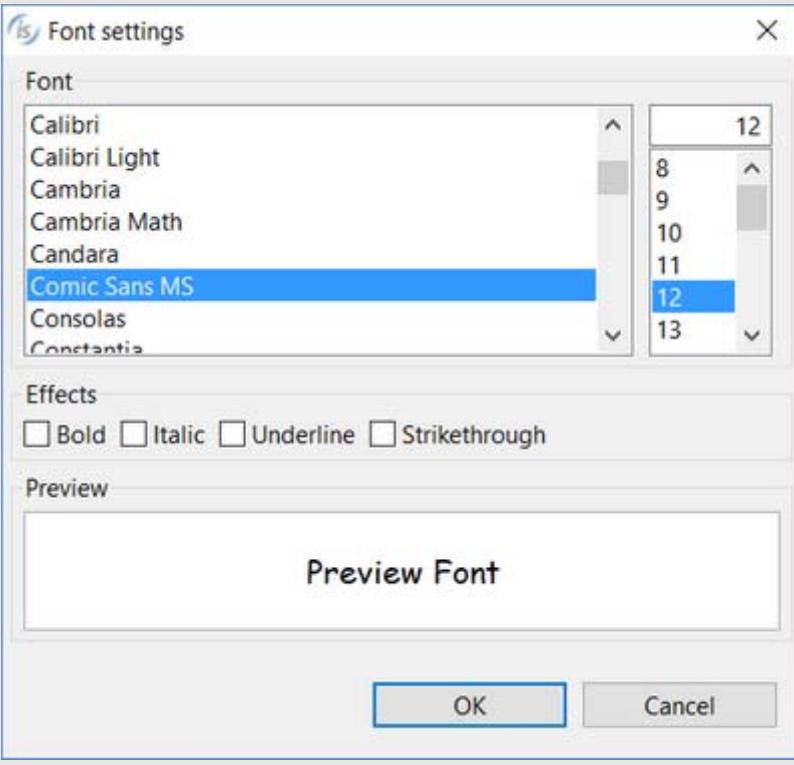
exclude event list	NONE... The <i>Exclude-Event-List</i> property is not generated. 0... <i>Exclude-Event-List=0</i> is generated. 1... <i>Exclude-Event-List=1</i> is generated.
export file format	Specifies the value for the <i>Export-File-Format</i> property.
export file name	Specifies the value for the <i>Export-File-Name</i> property.
filterable columns	TRUE...The <i>Filterable-Columns</i> style is generated FALSE... The <i>Filterable-Columns</i> style is not generated

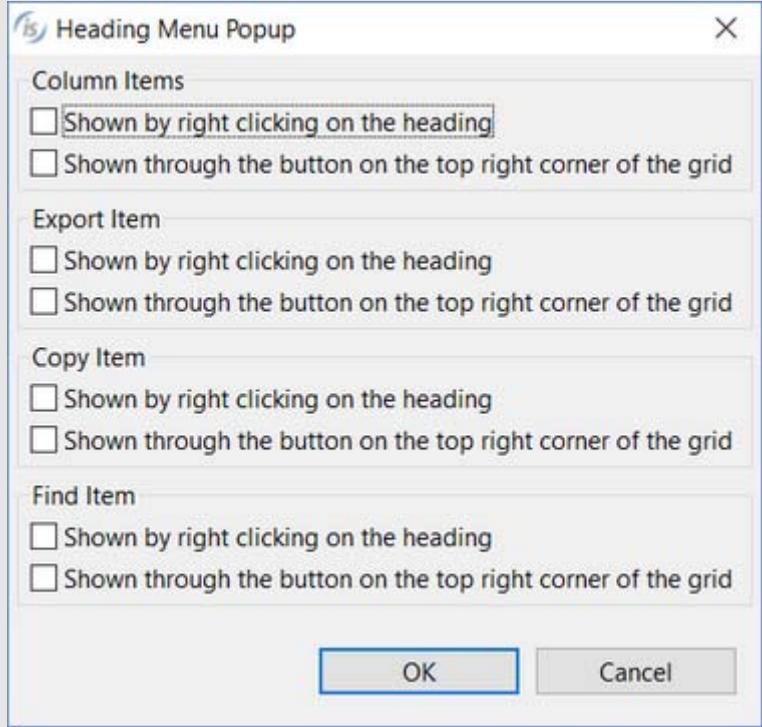
font Opens a dialog that allows the user to choose the control font.

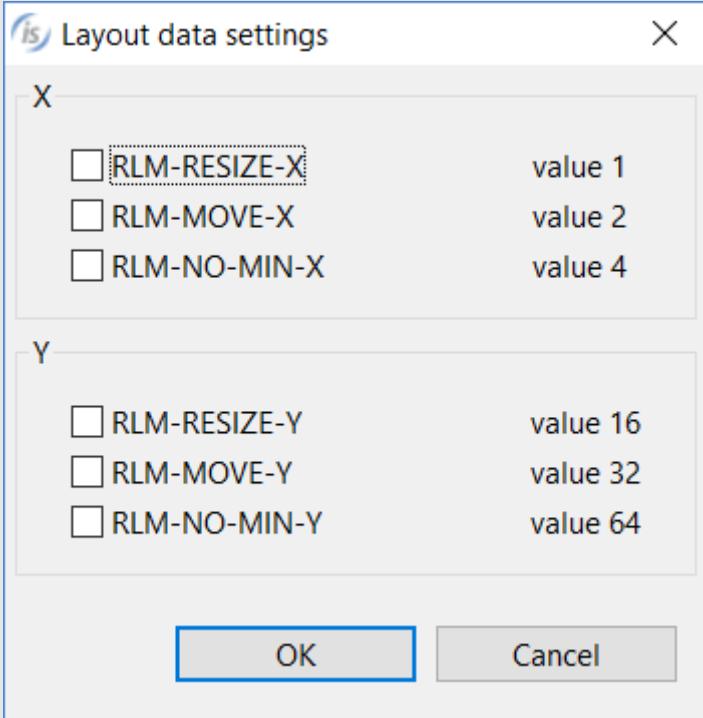


foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a 'Foreground' section where a red color swatch is selected, labeled '12 Red'. There is also an 'RGB' button. Below this is a 'Preview' section showing a horizontal bar colored red. At the bottom are 'OK' and 'Cancel' buttons.</p>
generate editor code	on Msg-Begin-Entry...The cell editor control will be displayed only when the cell is edited on Display... The cell editor control will be displayed on each cell as the grid appears

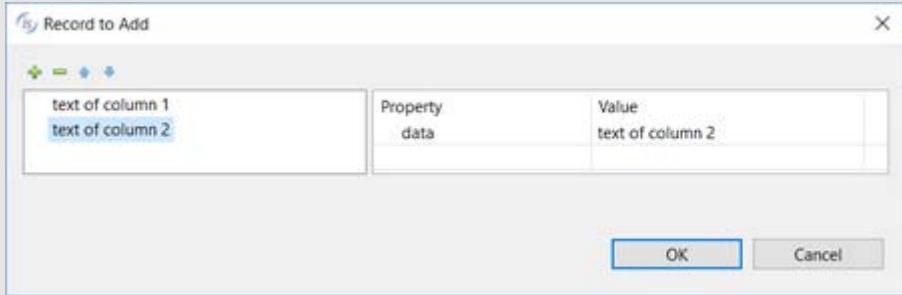
heading color	Opens a dialog to retrieve the value for the <i>Heading-Color</i> property
	 <p>The screenshot shows the 'Color settings' dialog box. It includes sections for 'Foreground' (black color swatch, labeled '1 Black') and 'Background' (yellow color swatch, labeled '15 Yellow'). Each color section has an 'RGB' button and a 'Foreground intensity' or 'Background intensity' group with radio buttons for 'Low', 'None' (selected), and 'High'. Below the colors are 'Effects' options for Reverse, Underline, Blink, and Protected. A 'Preview' section shows a yellow bar with the word 'Sample'. At the bottom are 'OK' and 'Cancel' buttons.</p>

heading divider color	Opens a dialog to retrieve the value for the <i>Heading-Divider-Color</i> property
	
heading font	Opens a dialog that allows the user to choose the heading font.
	

heading menu popup	Opens a dialog that allows to configure the Grid heading menu.
	
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE... The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
hscroll	TRUE... The <i>Hscroll</i> style is generated FALSE... The <i>Hscroll</i> style is not generated
id	Specifies the control id. This property is set automatically when the control is drawn.
key	Specifies the value for the <i>Key</i> property.
last-row	Specifies the value for the <i>Last-Row</i> property

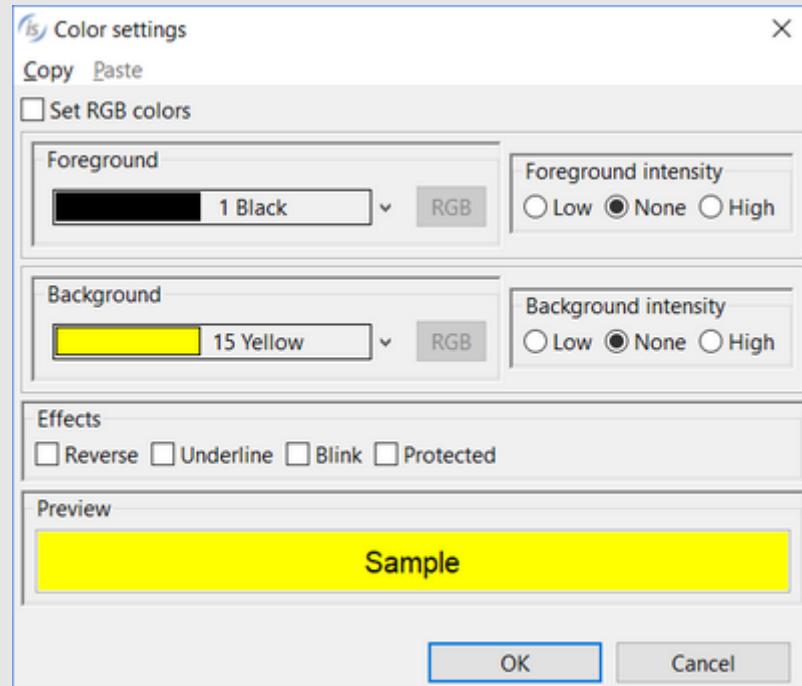
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	<p>DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting</p> <p>None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value</p> <p><i>CELLS</i>... <i>CELLS</i> is generated after the <i>Lines</i> value</p> <p><i>PIXELS</i>... <i>PIXELS</i> is generated after the <i>Lines</i> value</p>
lock	<p>TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse.</p> <p>FALSE...You can move the control on the Screen Designer by dragging it with the mouse</p>
lm-on-columns	<p>NONE... <i>Lm-On-Columns</i> is not generated</p> <p>TRUE... <i>Lm-On-Columns=1</i> is generated</p> <p>FALSE... <i>Lm-On-Columns=0</i> is generated</p>
mass-update	<p>TRUE... <i>Mass-Update=1</i> is generated</p> <p>FALSE... <i>Mass-Update</i> property is not generated</p>
max-height	Specifies the control maximum height as expressed in cells

max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
mouse-wheel-scroll	Specifies the value for the <i>Mouse-Wheel-Scroll</i> property
no-autosel	TRUE...The <i>No-Autoselection</i> style is generated FALSE...The <i>No-Autoselection</i> style is not generated
no-cell-drag	TRUE...The <i>No-Cell-Drag</i> style is generated FALSE...The <i>No-Cell-Drag</i> style is not generated
no search	TRUE...The <i>No-Search</i> style is generated FALSE... The <i>No-Search</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
num col headings	Specifies the value for the <i>Num-Col-Headings</i> property
num row headings	Specifies the value for the <i>Num-Row-Headings</i> property
num columns	Specifies the number of columns in the grid. Use column settings to configure the single columns.
num rows	Specifies the value for the <i>Num-Rows</i> property
paged	TRUE...The <i>Paged</i> style is generated FALSE... The <i>Paged</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
protection	Specifies the value for the <i>Protection</i> property
record-to-add	Opens a dialog that allows the user to describe the record



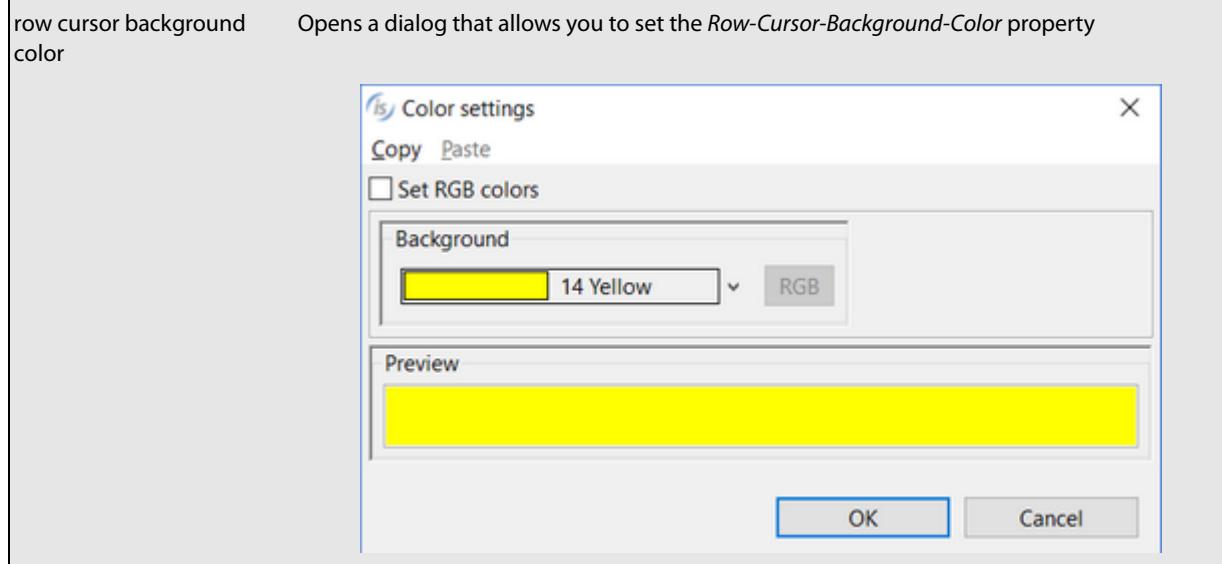
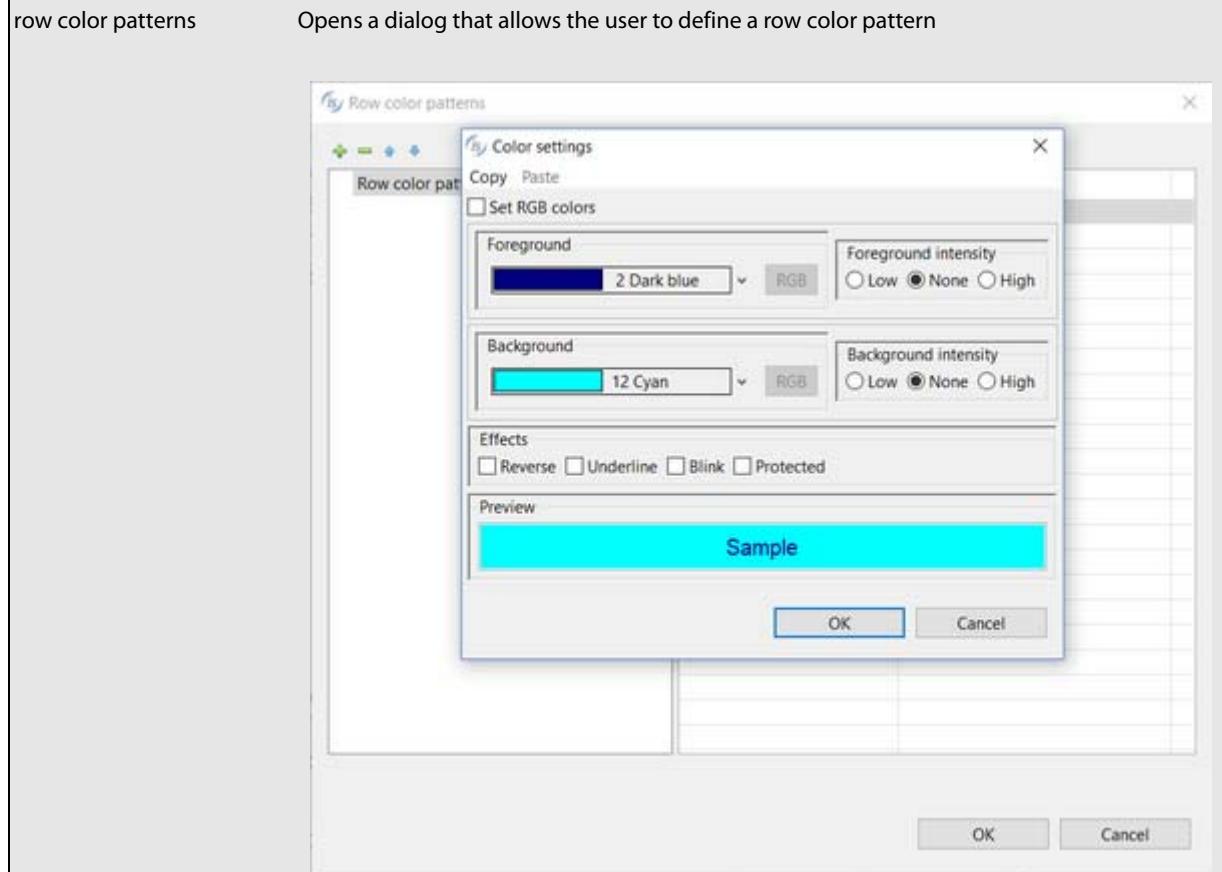
region color

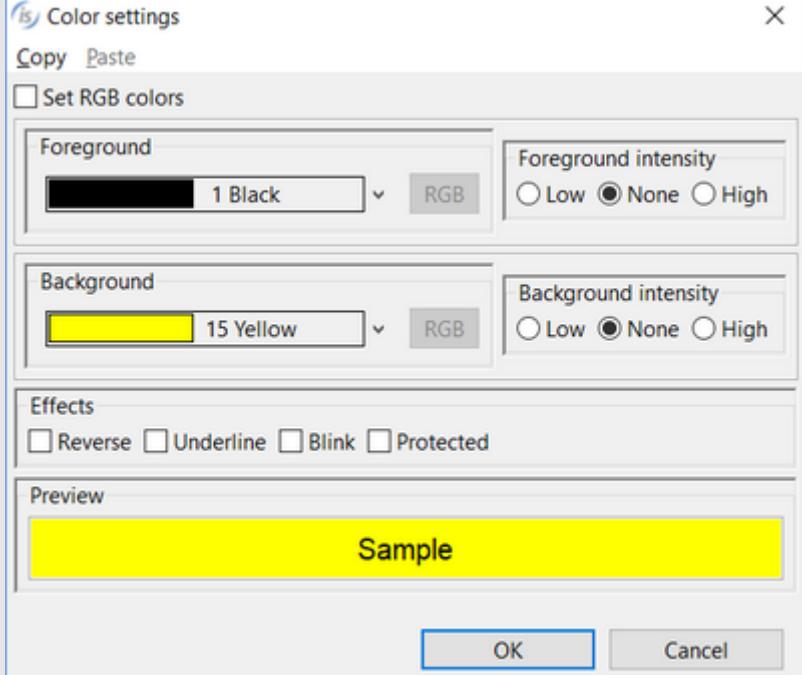
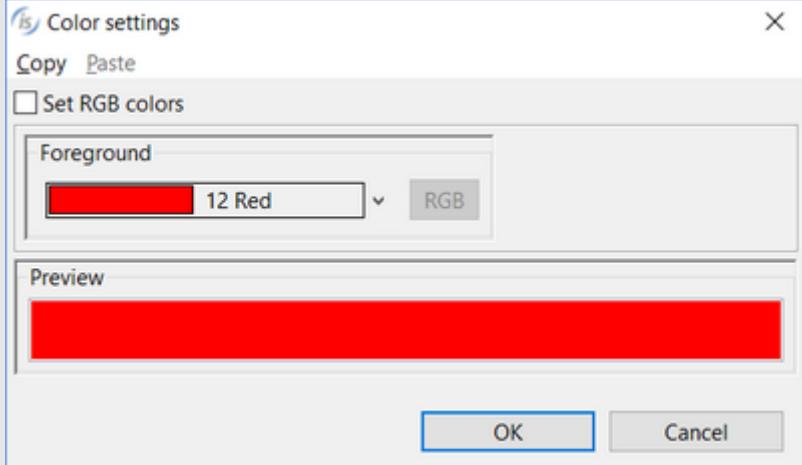
Opens a dialog to retrieve the value for the *Region-Color* property



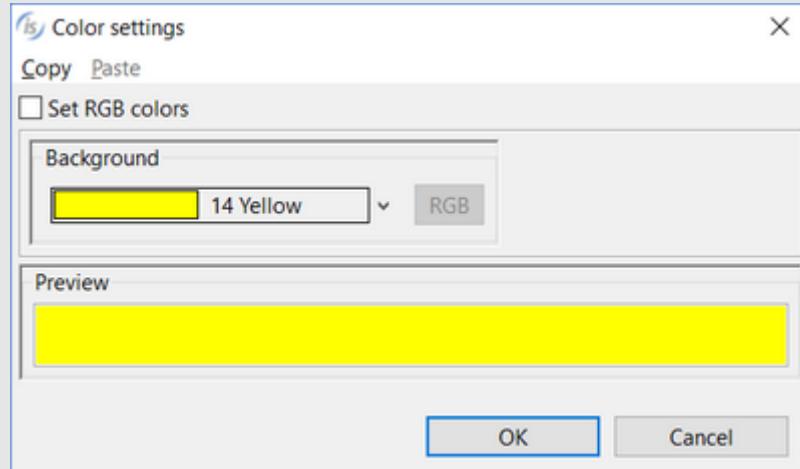
reordering columns

TRUE...The *Reordering-Columns* style is generated
FALSE...The *Reordering-Columns* style is not generated

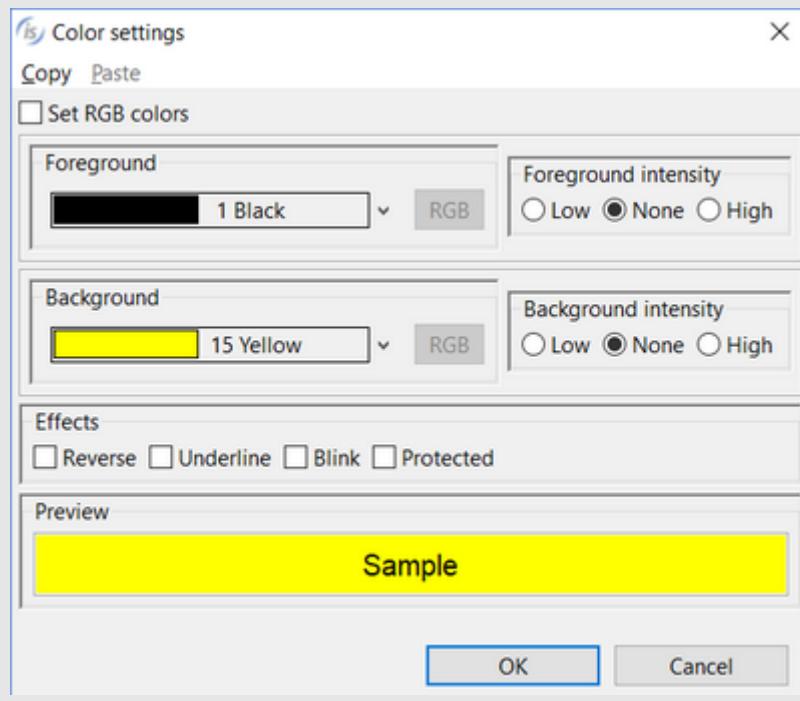


row cursor color	Opens a dialog that allows you to set the <i>Row-Cursor-Color</i> property
	 <p>The screenshot shows the 'Color settings' dialog box. It has sections for 'Foreground' (set to 'Black') and 'Background' (set to 'Yellow'). There are also sections for 'Effects' (checkboxes for Reverse, Underline, Blink, Protected) and a 'Preview' section showing a yellow bar labeled 'Sample'. Buttons for 'OK' and 'Cancel' are at the bottom.</p>
row cursor foreground color	Opens a dialog that allows you to set the <i>Row-Cursor-Foreground-Color</i> property
	 <p>The screenshot shows the 'Color settings' dialog box. It has a 'Foreground' section (set to 'Red') and a 'Preview' section showing a red bar. Buttons for 'OK' and 'Cancel' are at the bottom.</p>
row dividers	Specifies the value for the <i>Row-Dividers</i> property
row headings	TRUE...The <i>Row-Headings</i> style is generated FALSE...The <i>Row-Headings</i> style is not generated

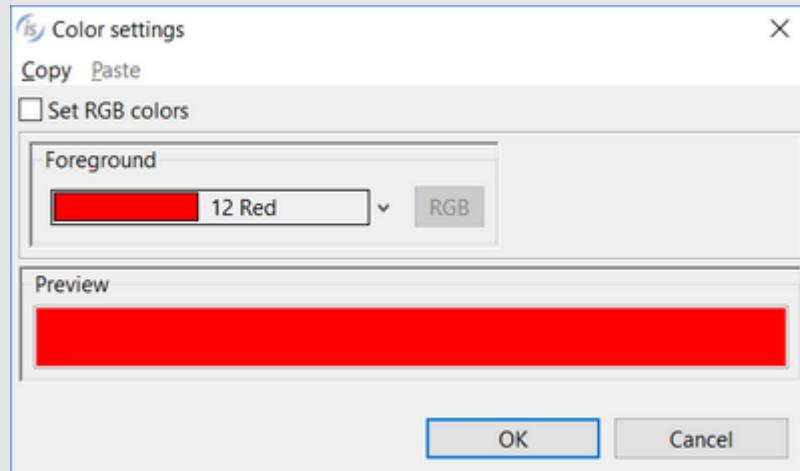
row selected background color Opens a dialog that allows to set the *Row-Selected-Background-Color* property.



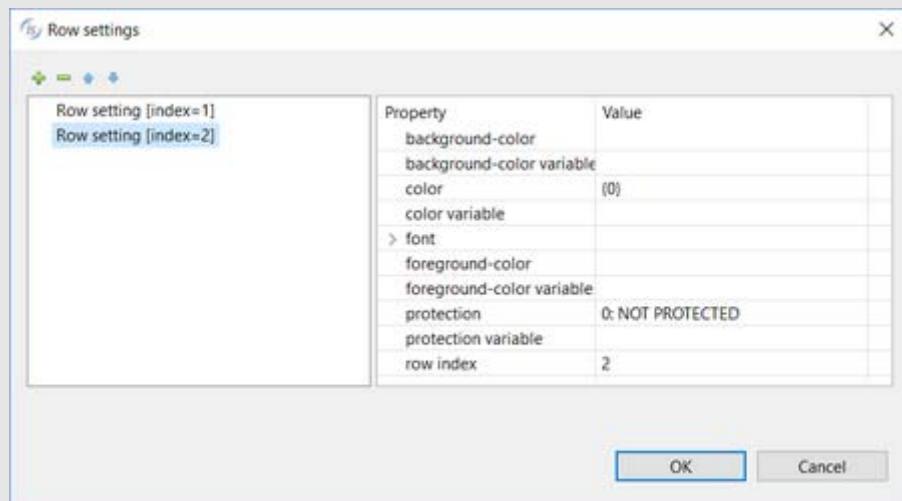
row selected color Opens a dialog that allows to set the *Row-Selected-Color* property.



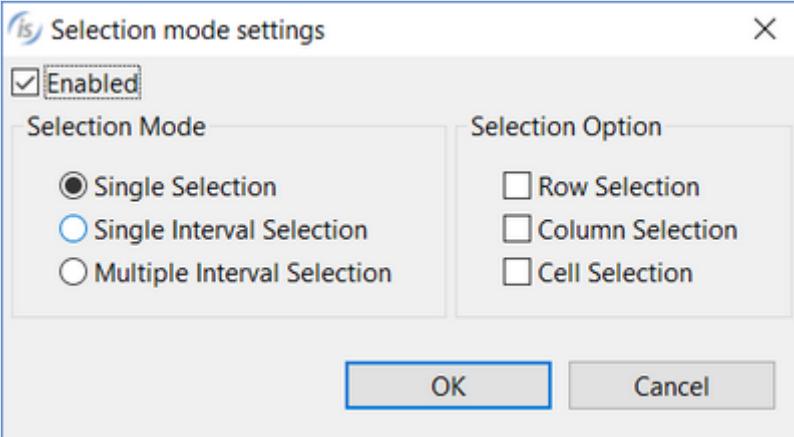
row selected foreground color Opens a dialog that allows to set the *Row-Selected-Foreground-Color* property.



row settings Opens a dialog that allows the user to configure rows characteristics



rows-per-page Specifies the value for the *Rows-Per-Page* property

selection mode	Opens a dialog that allows to set the <i>Selection-Mode</i> property.
	
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	<p>DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting</p> <p>None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value</p> <p><i>CELLS</i>... <i>CELLS</i> is generated after the <i>Size</i> value</p> <p><i>PIXELS</i>... <i>PIXELS</i> is generated after the <i>Size</i> value</p>
sortable columns	<p>TRUE...The <i>Sortable-Columns</i> style is generated</p> <p>FALSE...The <i>Sortable-Columns</i> style is not generated</p>
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
tiled heading	<p>TRUE... The <i>Tiled-Headings</i> style is generated</p> <p>FALSE... The <i>Tiled-Headings</i> style is not generated</p>
use TAB	<p>TRUE... The <i>Use-Tab</i> style is generated</p> <p>False... The <i>Use-Tab</i> style is not generated</p>
virtual width	Specifies the value for the <i>Virtual-Width</i> property
visible	<p>NONE...<i>Visible</i> property is not generated</p> <p>TRUE... <i>Visible=1</i> is generated</p> <p>FALSE...<i>Visible=0</i> is generated</p>
vpadding	Specifies the value for the <i>Vpadding</i> property
vscroll	<p>TRUE...The <i>Vscroll</i> style is generated</p> <p>FALSE...The <i>Vscroll</i> style is not generated</p>
width-in-cells	<p>TRUE...The <i>Width-In-Cells</i> style is generated</p> <p>FALSE... The <i>Width-In-Cells</i> style is not generated</p>
Events	

cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-begin-drag event	Allows the user to create a paragraph to handle the MSG-BEGIN-DRAG event in the Procedure Division
msg-begin-entry event	Allows the user to create a paragraph to handle the MSG-BEGIN-ENTRY event in the Procedure Division
msg-begin-heading-drag event	Allows the user to create a paragraph to handle the MSG-BEGIN-HEADING-DRAG event in the Procedure Division
msg-begin-sort event	Allows the user to create a paragraph to handle the MSG-BEGIN-SORT event in the Procedure Division
msg-bitmap-clicked	Allows the user to create a paragraph to handle the MSG-BITMAP-CLICKED event in the Procedure Division
msg-bitmap-dblclick	Allows the user to create a paragraph to handle the MSG-BITMAP-DBLCLICK event in the Procedure Division
msg-cancel-entry	Allows the user to create a paragraph to handle the MSG-CANCEL-ENTRY event in the Procedure Division
msg-col-width-changed	Allows the user to create a paragraph to handle the MSG-COL-WIDTH-CHANGED event in the Procedure Division
msg-end-drag event	Allows the user to create a paragraph to handle the MSG-END-DRAG event in the Procedure Division
msg-end-heading-drag event	Allows the user to create a paragraph to handle the MSG-END-HEADING-DRAG event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-finish-entry event	Allows the user to create a paragraph to handle the MSG-FINISH-ENTRY event in the Procedure Division
msg-finish-sort event	Allows the user to create a paragraph to handle the MSG-FINISH-SORT event in the Procedure Division
msg-gd-dblclick event	Allows the user to create a paragraph to handle the MSG-GD-DBLCLICK event in the Procedure Division
msg-goto-cell-drag event	Allows the user to create a paragraph to handle the MSG-GOTO-CELL-DRAG event in the Procedure Division
msg-goto-cell event	Allows the user to create a paragraph to handle the MSG-GOTO-CELL event in the Procedure Division
msg-goto-cell-mouse event	Allows the user to create a paragraph to handle the MSG-GOTO-CELL-.MOUSE event in the Procedure Division
msg-goto-cell-out-next event	Allows the user to create a paragraph to handle the MSG-GOTO-CELL-OUT-NEXT event in the Procedure Division

msg-goto-cell-out-prev event	Allows the user to create a paragraph to handle the MSG-GOTO-CELL-OUT-PREV event in the Procedure Division
msg-grid-rbutton-down event	Allows the user to create a paragraph to handle the MSG-GRID-RBUTTON-DOWN event in the Procedure Division
msg-grid-rbutton-up event	Allows the user to create a paragraph to handle the MSG-GRID-RBUTTON-UP event in the Procedure Division
msg-heading-clicked event	Allows the user to create a paragraph to handle the MSG-HEADING-CLICKED event in the Procedure Division
msg-heading-dblclick event	Allows the user to create a paragraph to handle the MSG-HEADING-DBLCLICK event in the Procedure Division
msg-heading-dragged event	Allows the user to create a paragraph to handle the MSG-HEADING-DRAGGED event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-paged-first event	Allows the user to create a paragraph to handle the MSG-PAGED-FIRST event in the Procedure Division
msg-paged-last event	Allows the user to create a paragraph to handle the MSG-PAGED-LAST event in the Procedure Division
msg-paged-next event	Allows the user to create a paragraph to handle the MSG-PAGED-NEXT event in the Procedure Division
msg-paged-nextpage event	Allows the user to create a paragraph to handle the MSG-PAGED-NEXTPAGE event in the Procedure Division
msg-paged-prev event	Allows the user to create a paragraph to handle the MSG-PAGED-PREV event in the Procedure Division
msg-paged-prevpage event	Allows the user to create a paragraph to handle the MSG-PAGED-PREVPAGE event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	

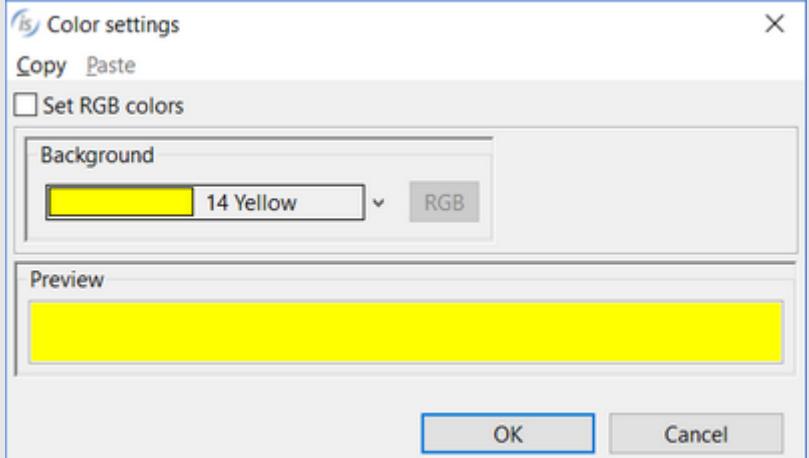
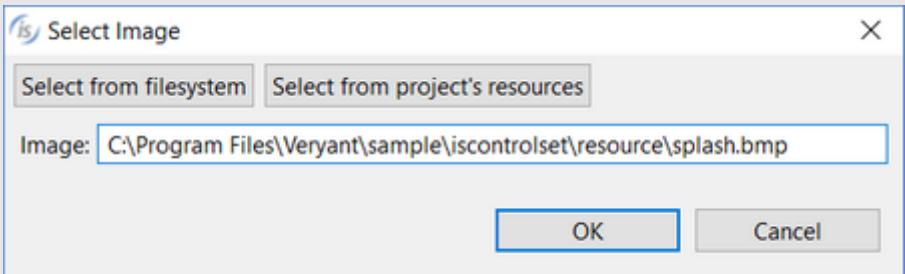
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Variables	
background-color variable	Numeric variable that hosts the value for the <i>Background-Color</i> property.
border color variable	Numeric variable that hosts the value for the <i>Border-Color</i> property.
cell-entry background color variable	Numeric variable that hosts the value for the <i>Cell-Entry-Background-Color</i> property.
cell-entry color variable	Numeric variable that hosts the value for the <i>Cell-Entry-Color</i> property.
cell-entry foreground color variable	Numeric variable that hosts the value for the <i>Cell-Entry-Foreground-Color</i> property.
cell selected background color variable	Numeric variable that hosts the value for the <i>Cell-Selected-Background-Color</i> property.
cell selected color variable	Numeric variable that hosts the value for the <i>Cell-Selected-Color</i> property.
cell selected foreground color variable	Numeric variable that hosts the value for the <i>Cell-Selected-Foreground-Color</i> property.
color variable	Numeric variable that hosts the color value
column headings height variable	Numeric variable that hosts the value for the <i>Column-Headings-Height</i> property.
column variable	Numeric variable that hosts the column value
cursor color variable	Numeric variable that hosts the value for the <i>Cursor-Color</i> property
cursor frame width variable	Numeric variable that hosts the value for the <i>Cursor-Frame-Width</i> property
cursor X variable	Numeric variable that hosts the value for the <i>Cursor-X</i> property
cursor Y variable	Numeric variable that hosts the value for the <i>Cursor-Y</i> property
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
divider color variable	Numeric variable that hosts the value for the <i>Divider-Color</i> property
drag color variable	Numeric variable that hosts the value for the <i>Drag-Color</i> property
enabled variable	Numeric variable that hosts the enabled state
end color variable	Numeric variable that hosts the value for the <i>End-Color</i> property

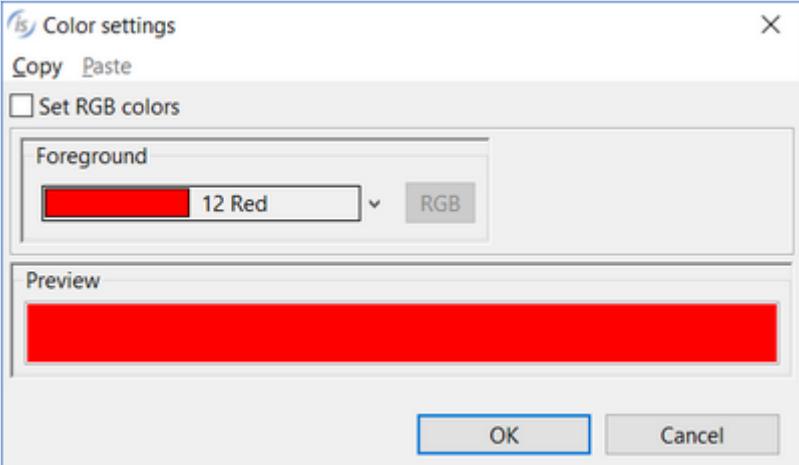
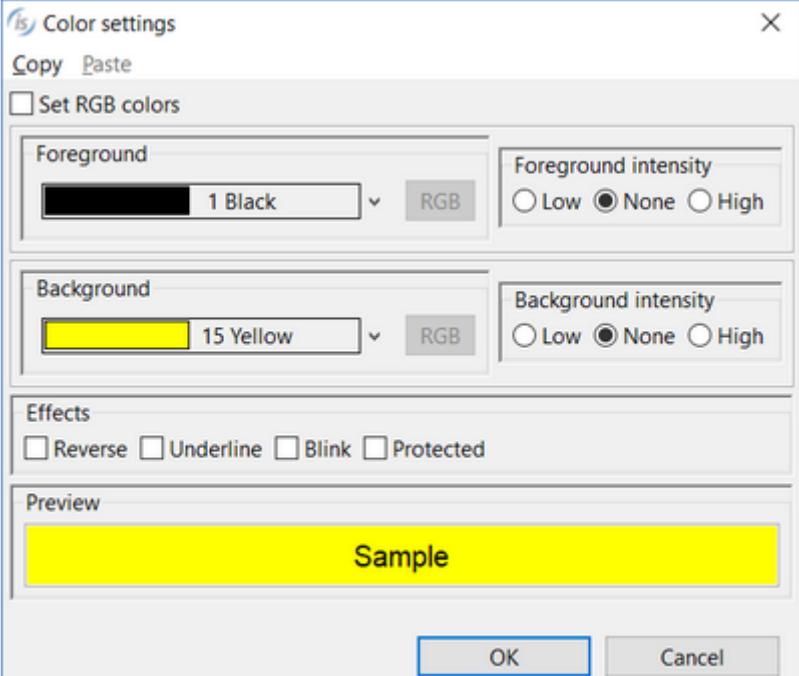
heading color variable	Numeric variable that hosts the value for the <i>Heading-Color</i> property
heading divider color variable	Numeric variable that hosts the value for the <i>Heading-Divider-Color</i> property
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the hint value.
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
last-row variable	Numeric variable that hosts the value for the <i>Last-Row</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
mass-update variable	Numeric variable that hosts the value for the <i>Mass-Update</i> property
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
num col headings variable	Numeric variable that hosts the value for the <i>Num-Col-Headings</i> property
num rows variable	Numeric variable that hosts the value for the <i>Num-Rows</i> property
protection variable	Numeric variable that hosts the value for the <i>Protection</i> property
record data	Numeric variable that hosts the value for the <i>Record-Data</i> property
record-to-add variable	Numeric variable that hosts the value for the <i>Record-To-Add</i> property
region color variable	Numeric variable that hosts the value for the <i>Region-Color</i> property
size variable	Numeric variable that hosts the size value
value container	occurs item that hosts control items
virtual width variable	Numeric variable that hosts the value for the <i>Virtual-Width</i> property
visible variable	Numeric variable that hosts the visible state
vpadding variable	Numeric variable that hosts the value for the <i>Vpadding</i> property

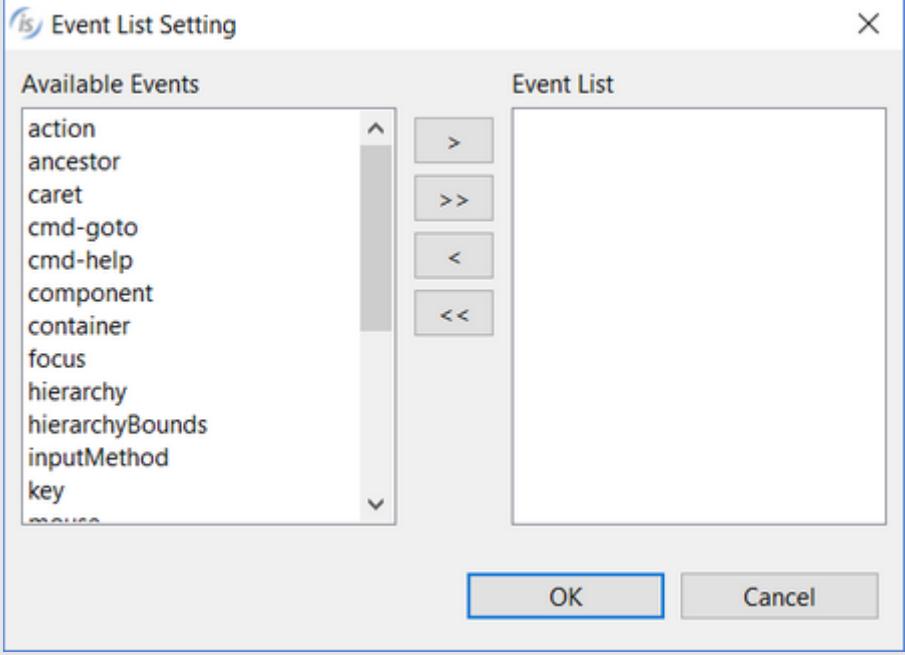
JAVA BEAN

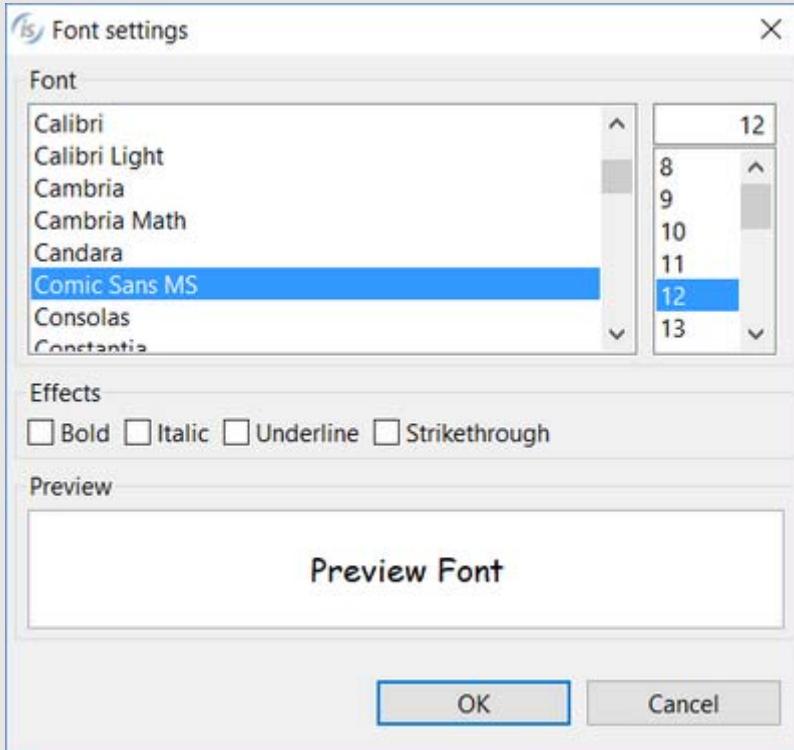
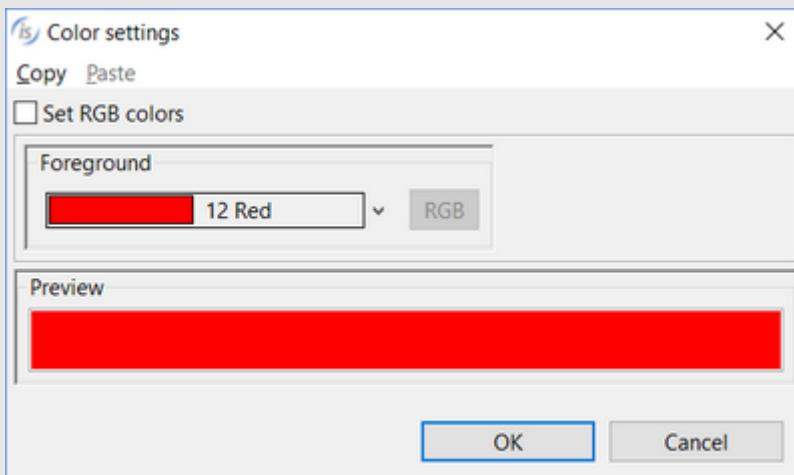
Refer to [JAVA-BEAN](#) for details about properties, styles and events of this control.

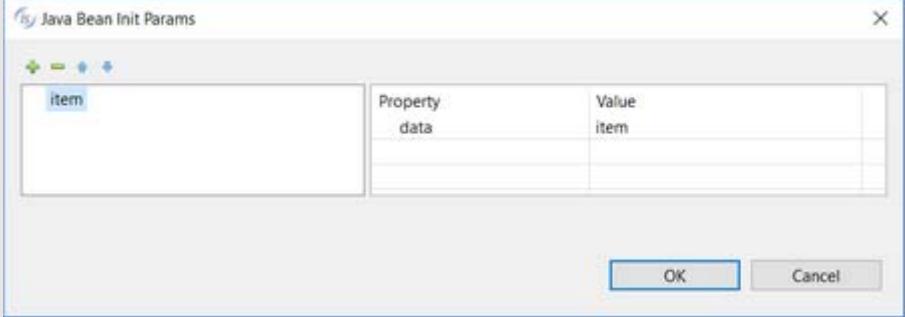
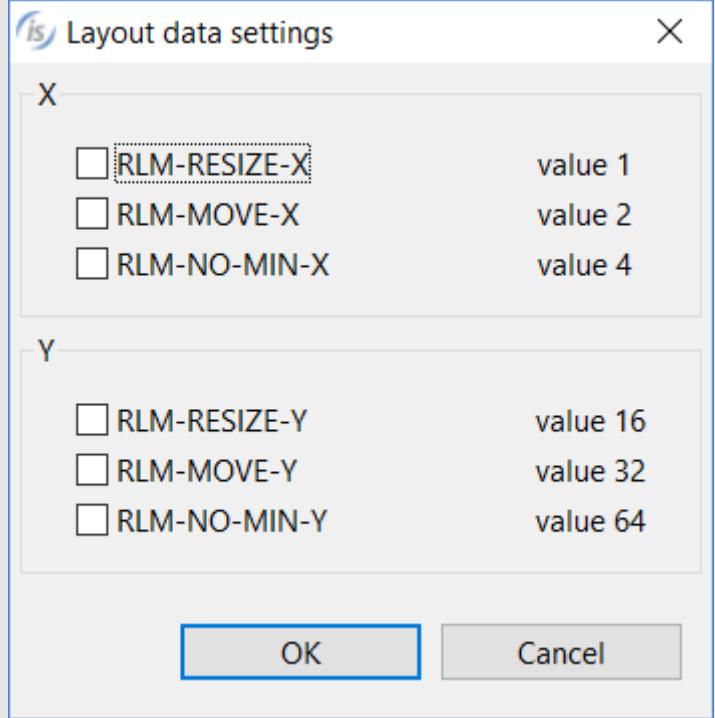
Properties

(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
background-color	Opens a dialog that allows the user to choose the control background color. 
bitmap	Opens a dialog box that allows the user to select an image file to load into the control 
bitmap-width	Specifies the value for the <i>Bitmap-Width</i> property
border	Allows the user to choose between one of the following styles: 3-D BOXED NO-BOX

border-color	Opens a dialog that allows the user to choose the control border color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has sections for 'Foreground' and 'Background'. Under 'Foreground', a color swatch is red, labeled '12 Red', with an 'RGB' button next to it. A preview bar below shows a red rectangle. At the bottom are 'OK' and 'Cancel' buttons.</p>
CLSID	Specifies the value for the <i>CLSID</i> property
color	Opens a dialog that allows the user to choose the control color.
	 <p>The screenshot shows the 'Color settings' dialog box with more extensive options. It includes sections for 'Foreground' (black, labeled '1 Black') and 'Background' (yellow, labeled '15 Yellow'), each with an 'RGB' button. To the right of the foreground section are 'Foreground intensity' options: 'Low', 'None' (selected), and 'High'. Below these are 'Effects' checkboxes for 'Reverse', 'Underline', 'Blink', and 'Protected'. A 'Preview' section shows a yellow rectangle with the word 'Sample'. At the bottom are 'OK' and 'Cancel' buttons.</p>
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.

destroy type	AUTOMATIC...neither the <i>Temporary</i> nor Permanent styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
event list	Opens a dialog that allows to choose which events must be added to the event list of this control. The content depends on the CLSID property.
	 <p>The dialog box is titled "Event List Setting". It has two main panes: "Available Events" on the left and "Event List" on the right. The "Available Events" pane contains a list of event names: action, ancestor, caret, cmd-goto, cmd-help, component, container, focus, hierarchy, hierarchyBounds, inputMethod, key, and menu. To the right of this list are four buttons: a top-right button with a right-pointing arrow (>), a middle-right button with a double-right-pointing arrow (>>), a bottom-right button with a left-pointing arrow (<), and a bottom-left button with a double-left-pointing arrow (<<). Below the "Event List" pane are two buttons: "OK" and "Cancel".</p>
exclude event list	NONE... The <i>Exclude-Event-List</i> property is not generated. 0... <i>Exclude-Event-List=0</i> is generated. 1... <i>Exclude-Event-List=1</i> is generated.

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical scroll bar and a size selector with values 8, 9, 10, 11, 12 (highlighted), and 13. Below the font list are sections for 'Effects' (Bold, Italic, Underline, Strikethrough) and 'Preview' (a text area containing 'Preview Font'). At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It includes a 'Copy' and 'Paste' button, a checkbox for 'Set RGB colors', and a 'Foreground' color picker set to '12 Red'. A 'RGB' button is next to the picker. Below the picker is a 'Preview' section with a red rectangular preview area. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
hscroll	TRUE...The <i>Hscroll</i> style is generated FALSE...The <i>Hscroll</i> style is not generated

id	Specifies the control id. This property is set automatically when the control is drawn.
init parameters	Opens a dialog that allows the user to set the init parameters 
init signature	Specifies the value for the <i>Init-Signature</i> property
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules. 
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn

lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
use ALT	TRUE...The <i>Use-Alt</i> style is generated FALSE...The <i>Use-Alt</i> style is not generated
use RETURN	TRUE...The <i>Use-Return</i> style is generated FALSE...The <i>Use-Return</i> style is not generated
use TAB	TRUE...The <i>Use-Tab</i> style is generated FALSE...The <i>Use-Tab</i> style is not generated
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
vscroll	TRUE...The <i>Vscroll</i> style is generated FALSE...The <i>Vscroll</i> style is not generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division

cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-jb-event event	Allows the user to create a paragraph to handle the MSG-JB-EVENT event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPTION PROCEDURE
Initialization routine	Allows the user to create a paragraph to handle the control initialization
Variables	
bitmap-width variable	Numeric variable that hosts the value for the <i>Bitmap-Width</i> property
CLSID variable	Alphanumeric variable that hosts the value for the <i>CLSID</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state

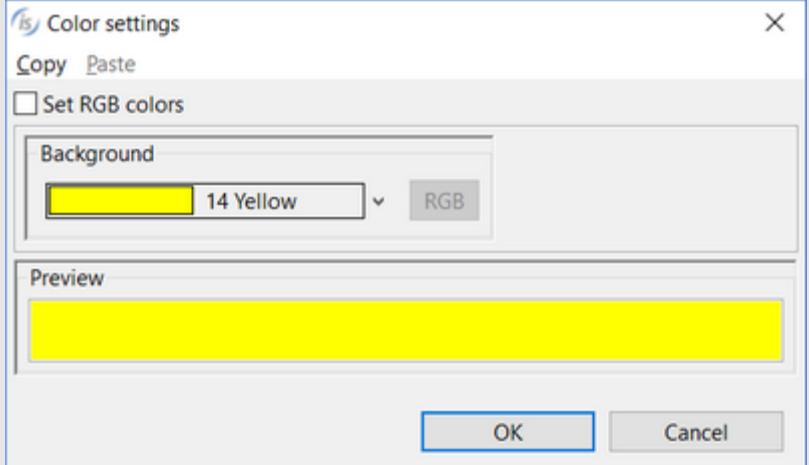
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the hint value.
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the Key property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
object in	Alphanumeric variable that hosts the object logical name
size variable	Numeric variable that hosts the size value
visible variable	Numeric variable that hosts the visible state

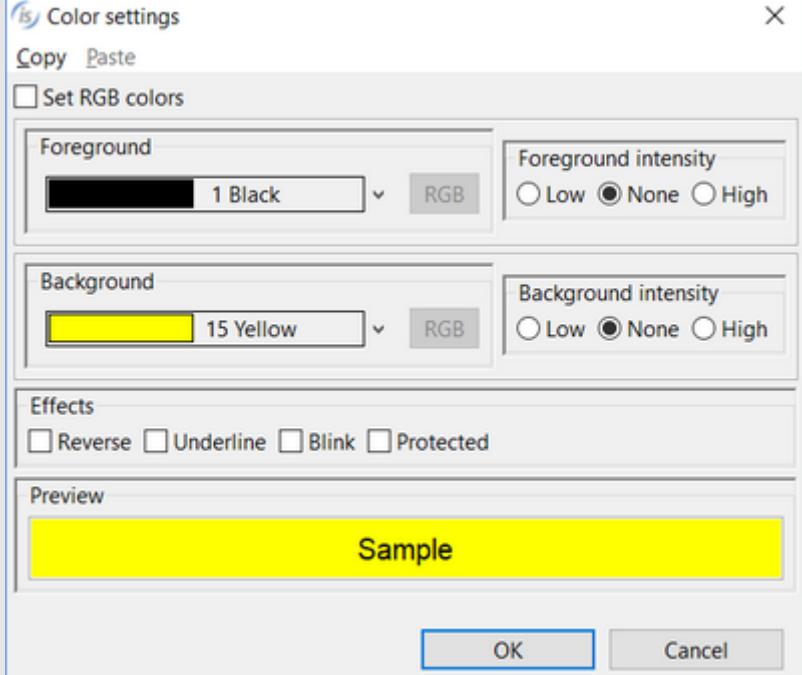
LABEL

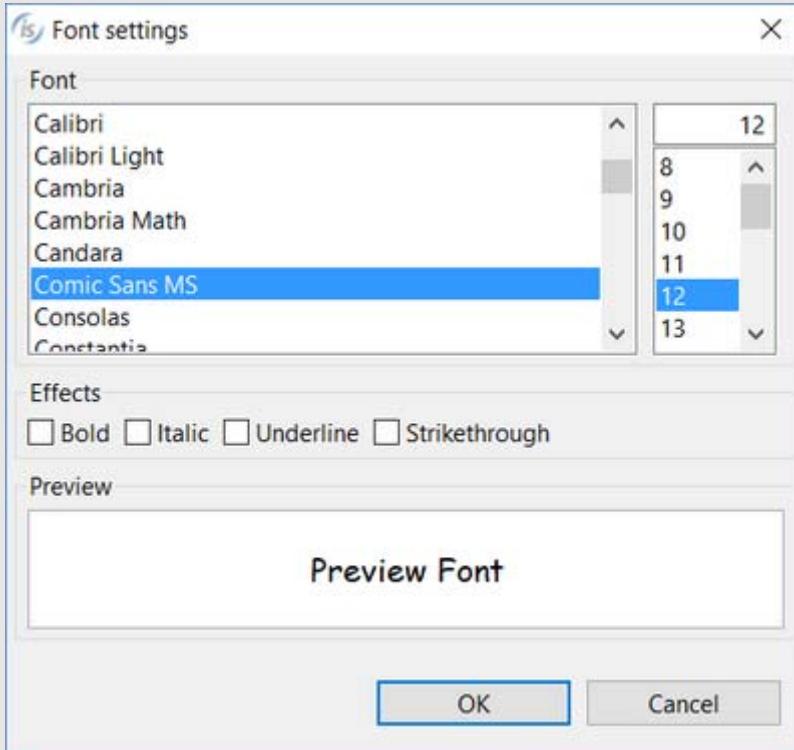
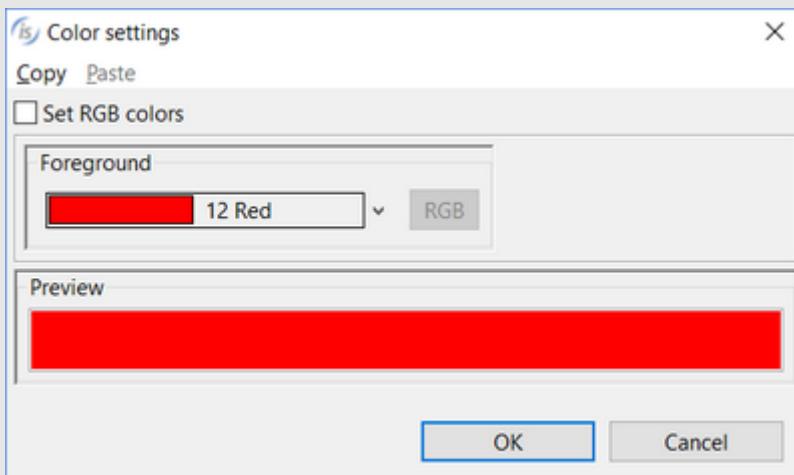
Refer to [LABEL](#) for details about properties, styles and events of this control.

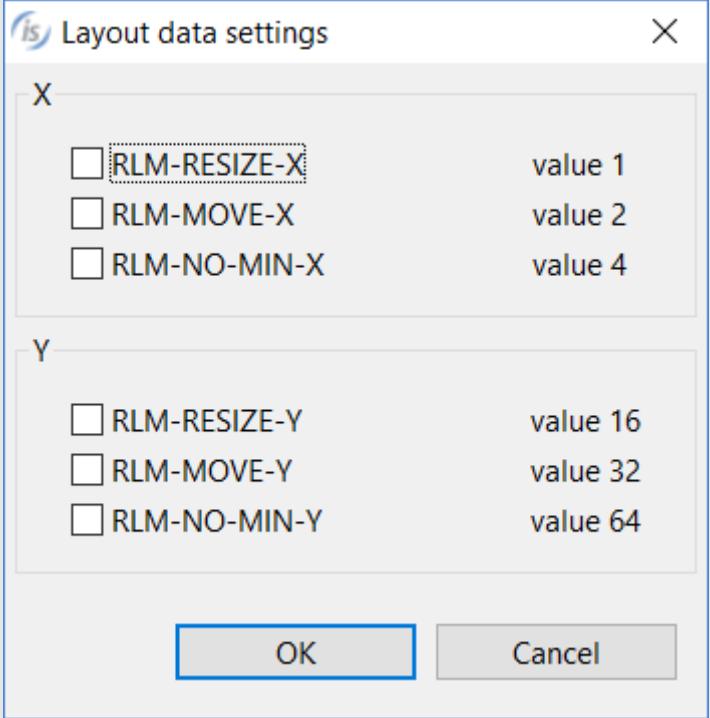
Properties

(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
background-color	Opens a dialog that allows the user to choose the control background color.



color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor <i>Permanent</i> styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a title bar 'Font settings' with a close button 'X'. Below it is a 'Font' section containing a list of font names: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical scroll bar and a size dropdown menu showing values 8, 9, 10, 11, 12 (selected), and 13. Below the font list is an 'Effects' section with checkboxes for Bold, Italic, Underline, and Strikethrough, all of which are unchecked. Underneath that is a 'Preview' section with a text area containing the placeholder 'Preview Font'. At the bottom are 'OK' and 'Cancel' buttons, with 'OK' being the primary button.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a title bar 'Color settings' with a close button 'X'. Below it is a 'Copy' and 'Paste' button, followed by a checkbox 'Set RGB colors' which is unchecked. A 'Foreground' section contains a color swatch set to '12 Red' and a 'RGB' button. Below that is a 'Preview' section with a red rectangular preview area. At the bottom are 'OK' and 'Cancel' buttons, with 'OK' being the primary button.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property

orientation	Horizontal Left... No orientation style is generated, so <i>Left</i> is assumed Horizontal Center... The <i>Center</i> style is generated Horizontal Right...The <i>Right</i> style is generated Vertical Top-Left... The <i>Vertical</i> style is generated Vertical Top-Right... The <i>Vertical</i> and <i>Right</i> styles are generated Vertical Center... The <i>Vertical</i> and <i>Center</i> styles are generated Vertical Bottom-Left... The <i>Vertical</i> and <i>Bottom</i> styles are generated Vertical Bottom-Right... The <i>Vertical</i> , <i>Bottom</i> and <i>Right</i> styles are generated
id	Specifies the control id. This property is set automatically when the control is drawn.
key	Specifies the value for the <i>Key</i> property.
label offset	Specifies the value for the <i>Label-Offset</i> property
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn

lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
no-key-letter	TRUE...The <i>No-Key-Letter</i> style is generated FALSE...The <i>No-Key-Letter</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
title	Specifies the value for the <i>Title</i> property
title picture	Specifies the picture for the title variable
top	TRUE...The <i>Top</i> style is generated FALSE...The <i>Top</i> style is not generated
transparent	TRUE...The <i>Transparent</i> style is generated FALSE...The <i>Transparent</i> style is not generated
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
Events	No Events available.
Exceptions	

No Exceptions available.

Procedures

No Procedures available.

Variables

color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the value for the <i>Hint</i> property
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
label-offset variable	Numeric variable that hosts the value for the <i>Label-Offset</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
title variable	Alphanumeric variable that hosts the value for the <i>Title</i> property
visible variable	Numeric variable that hosts the visible state

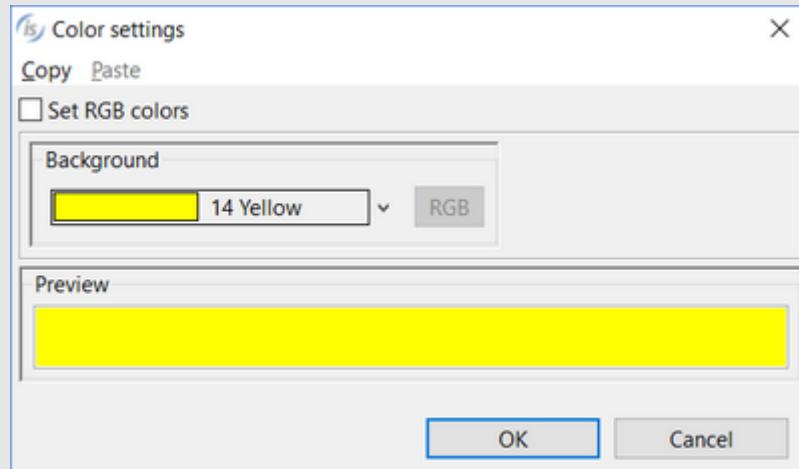
LIST BOX

Refer to [LIST-BOX](#) for details about properties, styles and events of this control.

Properties

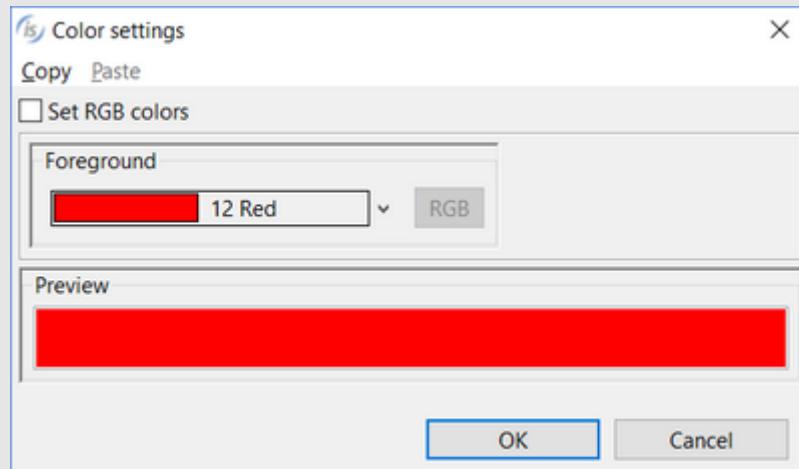
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.

background-color Opens a dialog that allows the user to choose the control background color.

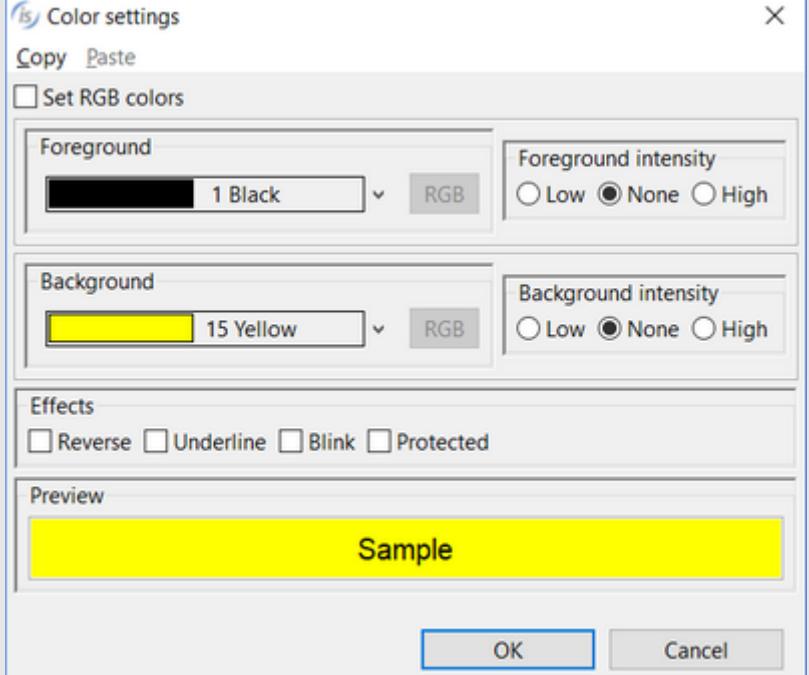
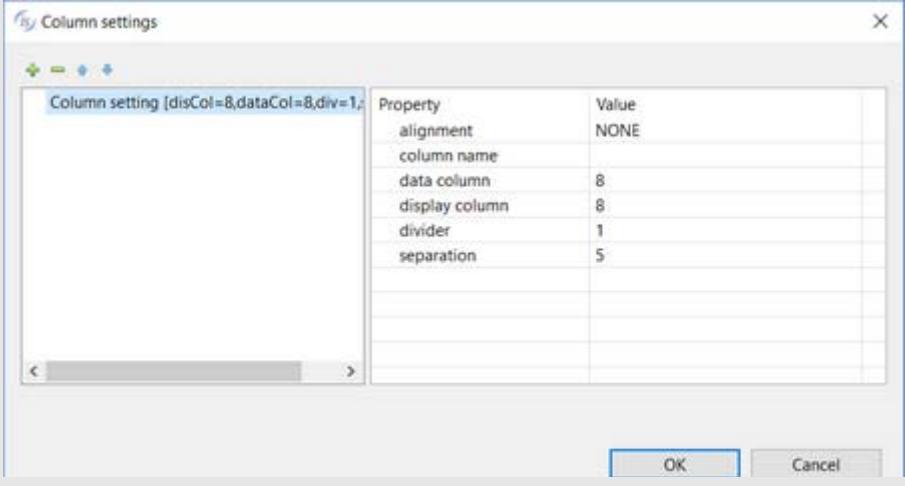


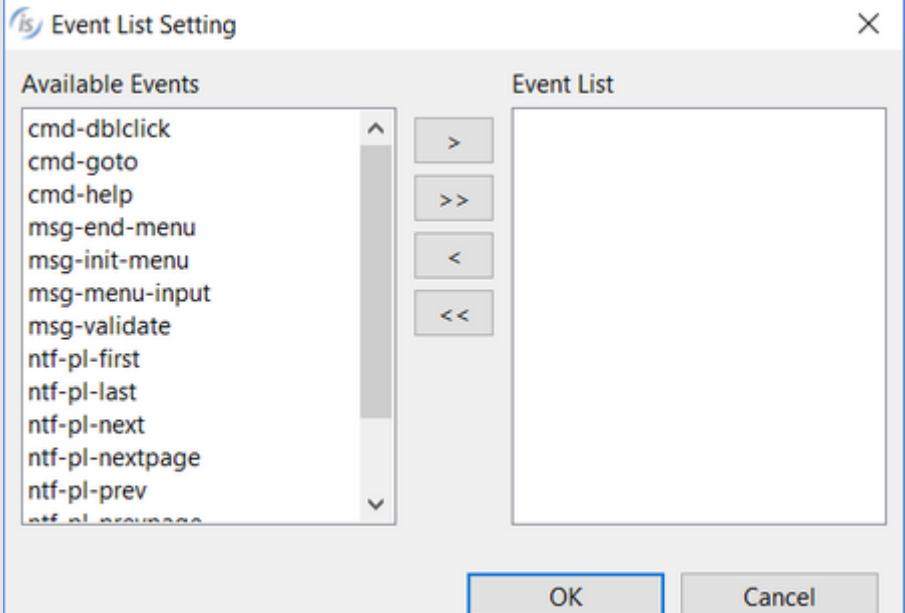
border Allows the user to set one of the following three styles:
3-D
BOXED
NO-BOX

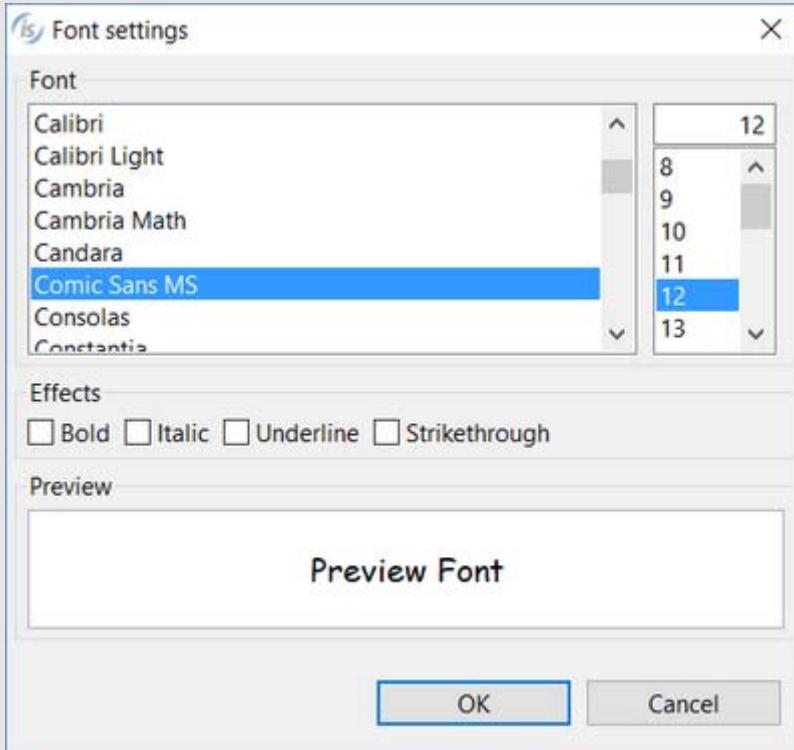
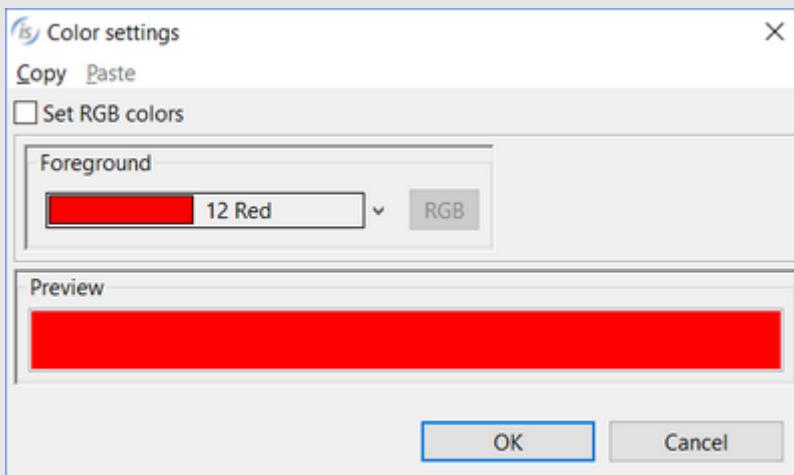
border-color Opens a dialog that allows the user to choose the control border color.

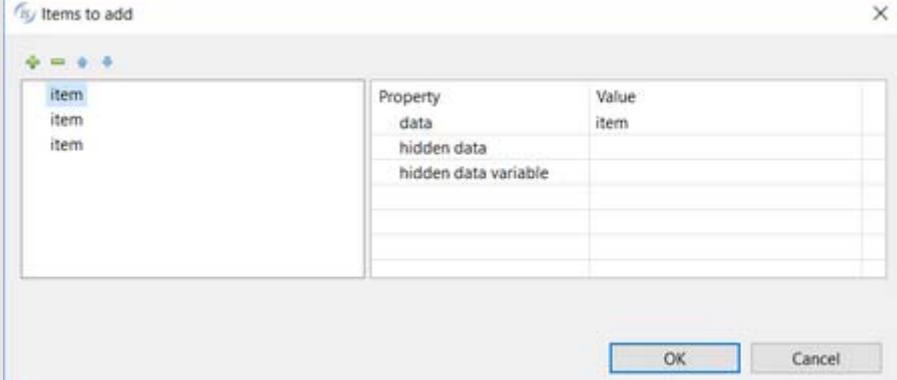
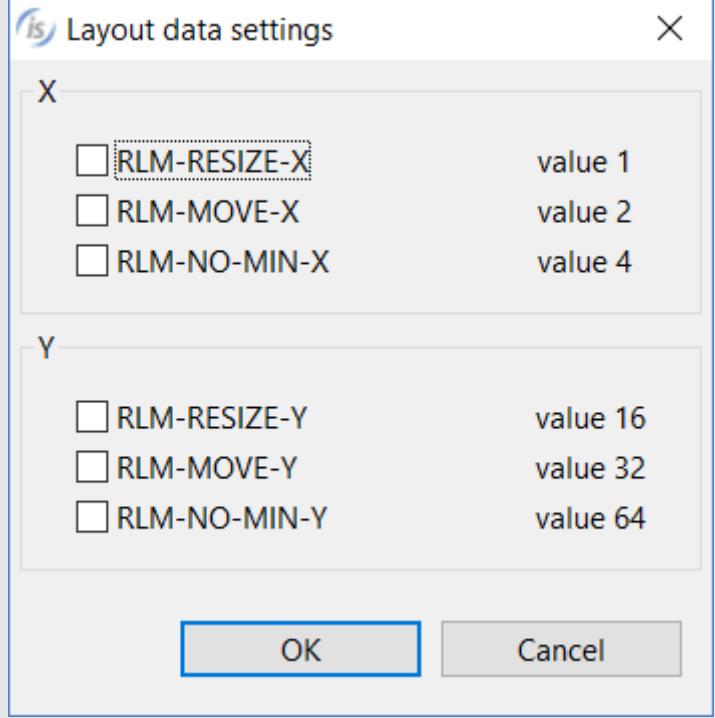


case None....Neither the *Upper* nor *Lower* styles are generated
UPPER....*Upper* style is generated
LOWER...*Lower* style is generated

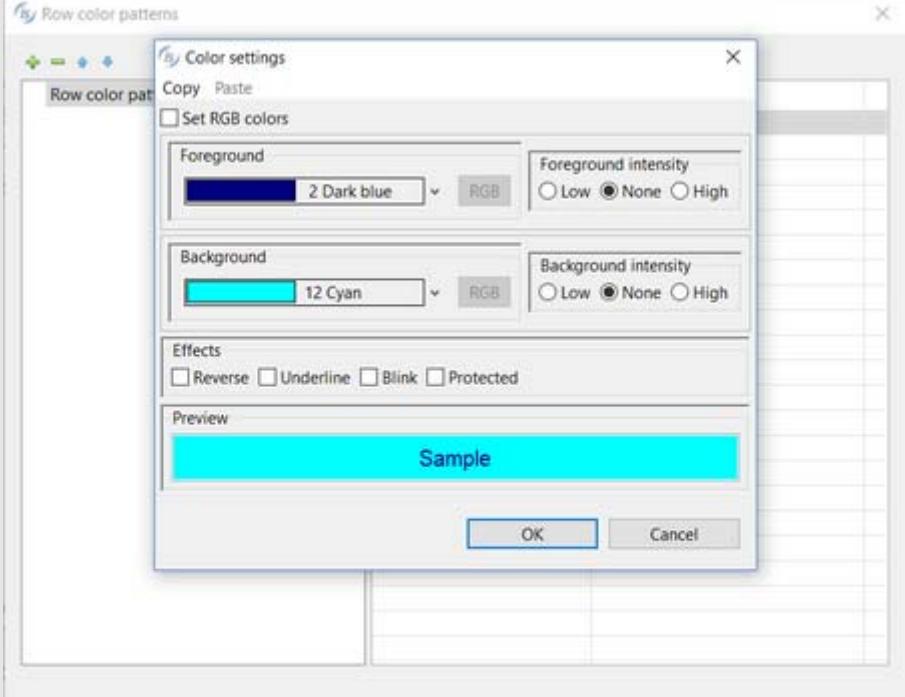
color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
column settings	Opens a dialog that allows the user to define columns
	
css-base-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment.
css-style-name	See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.

destroy type	AUTOMATIC...neither the <i>Temporary</i> nor Permanent styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main panes: "Available Events" on the left and "Event List" on the right. The "Available Events" pane contains a list of event names: cmd-dblclick, cmd-goto, cmd-help, msg-end-menu, msg-init-menu, msg-menu-input, msg-validate, ntf-pl-first, ntf-pl-last, ntf-pl-next, ntf-pl-nextpage, and ntf-pl-prev. Between the two panes are four buttons: '>', '>>', '<', and '<<'. At the bottom are "OK" and "Cancel" buttons.</p>
exception-value	Specifies the value for the <i>Exception-Value</i> property
exclude event list	NONE... The <i>Exclude-Event-List</i> property is not generated. 0... <i>Exclude-Event-List=0</i> is generated. 1... <i>Exclude-Event-List=1</i> is generated.
export file format	Specifies the value for the <i>Export-File-Format</i> property.
export file name	Specifies the value for the <i>Export-File-Name</i> property.

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. On the right, there's a vertical font size selector with values 8, 9, 10, 11, 12, and 13, where '12' is selected. Below the font list are sections for 'Effects' (Bold, Italic, Underline, Strikethrough) and 'Preview' (a text area containing 'Preview Font'). At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It includes a 'Set RGB colors' checkbox, which is unchecked. Below it is a 'Foreground' color picker set to '12 Red'. A preview window shows a red rectangle. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property
id	Specifies the control id. This property is set automatically when the control is drawn.

item-to-add	Opens a dialog that allows the user to set text and icons for each single item
	
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn

lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lm-on-columns	NONE... <i>Lm-On-Columns</i> is not generated TRUE... <i>Lm-On-Columns=1</i> is generated FALSE... <i>Lm-On-Columns=0</i> is generated
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
mass-update	TRUE... <i>Mass-Update=1</i> is generated FALSE... <i>Mass-Update</i> property is not generated
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
mouse-wheel-scroll	Specifies the value for the <i>Mouse-Wheel-Scroll</i> property
no search	TRUE...The <i>No-Search</i> style is generated FALSE...The <i>No-Search</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
notify dblclick	TRUE...The <i>Notify-Dblclick</i> style is generated FALSE...The <i>Notify-Dblclick</i> style is not generated
notify selchange	TRUE...The <i>Notify-Selchange</i> style is generated FALSE...The <i>Notify-Selchange</i> style is not generated
paged	TRUE...The <i>Paged</i> style is generated FALSE... The <i>Paged</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.

row color patterns	Opens a dialog that allows the user to define a row color pattern
	
selection-index	Specifies the value for the <i>Selection-Index</i> property
selection-mode	SINGLE-SELECTION... <i>Selection-Mode=1</i> is generated. SINGLE-INTERVAL-SELECTION... <i>Selection-Mode=2</i> is generated. MULTIPLE-INTERVAL-SELECTION... <i>Selection-Mode=4</i> is generated.
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
sort order	Specifies the value for the <i>Sort-Order</i> property. You can choose between: 0 PL-SORT-DEFAULT 1 PL-SORT-NONE 2 PL-SORT-NATIVE 3PL-SORT-NATIVE-IGNORE-CASE
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
termination-value	Specifies the value for the <i>Termination-Value</i> property
unsorted	TRUE... The <i>Unsorted</i> style is generated FALSE... The <i>Unsorted</i> style is not generated
value	Specifies the value for the <i>Value</i> property

visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-dblclick event	Allows the user to create a paragraph to handle the CMD-DBLCLICK event in the Procedure Division
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
ntf-pl-first event	Allows the user to create a paragraph to handle the MSG-PL-FIRST event in the Procedure Division
ntf-pl-last event	Allows the user to create a paragraph to handle the NTF-PL-LAST event in the Procedure Division
ntf-pl-next event	Allows the user to create a paragraph to handle the NTF-PL-NEXT event in the Procedure Division
ntf-pl-nextpage event	Allows the user to create a paragraph to handle the NTF-PL-NEXTPAGE event in the Procedure Division
ntf-pl-prev event	Allows the user to create a paragraph to handle the NTF-PL-PREV event in the Procedure Division
ntf-pl-prevpage event	Allows the user to create a paragraph to handle the NTF-PL-PREVPAGE event in the Procedure Division
ntf-pl-search event	Allows the user to create a paragraph to handle the NTF-PL-SEARCH event in the Procedure Division
ntf-selchange-event	Allows the user to create a paragraph to handle the NTF-SELCHANGE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	

cmd-dblclick exception	Allows the user to create a paragraph to handle the CMD-DBLCLICK event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-pl-first exception	Allows the user to create a paragraph to handle the NTF-PL-FIRST event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-pl-last exception	Allows the user to create a paragraph to handle the NTF-PL-LAST event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-pl-next exception	Allows the user to create a paragraph to handle the NTF-PL-NEXT event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-pl-nextpage exception	Allows the user to create a paragraph to handle the NTF-PL-NEXTPAGE event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-pl-prev exception	Allows the user to create a paragraph to handle the NTF-PL-PREV event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-pl-prevpage exception	Allows the user to create a paragraph to handle the NTF-PL-PREVPAGE event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-pl-search exception	Allows the user to create a paragraph to handle the NTF-PL-SEARCH event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-selchange exception	Allows the user to create a paragraph to handle the NTF-SELCHANGE event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Link To	Associates a paragraph with the control that will be executed when the control is double clicked
Variables	
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value

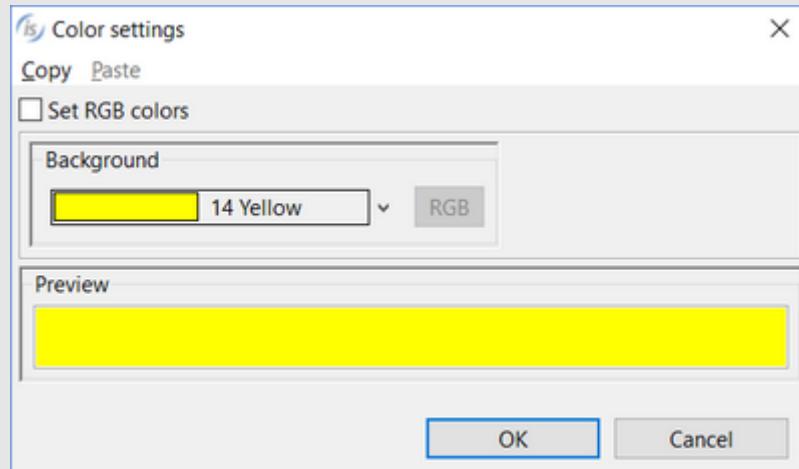
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
exception-value variable	Numeric variable that hosts the value for the <i>Exception-Value</i> property
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the value for the <i>Hint</i> property
id variable	Numeric variable that hosts the control id
item-to-add variable	Alphanumeric variable that hosts the value for the <i>Item-To-Add</i> property
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
value container	occurs item that hosts control items
value variable	Alphanumeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

LIST BOX CHECKED

Refer to [LIST-BOX](#) for details about properties, styles and events of this control.

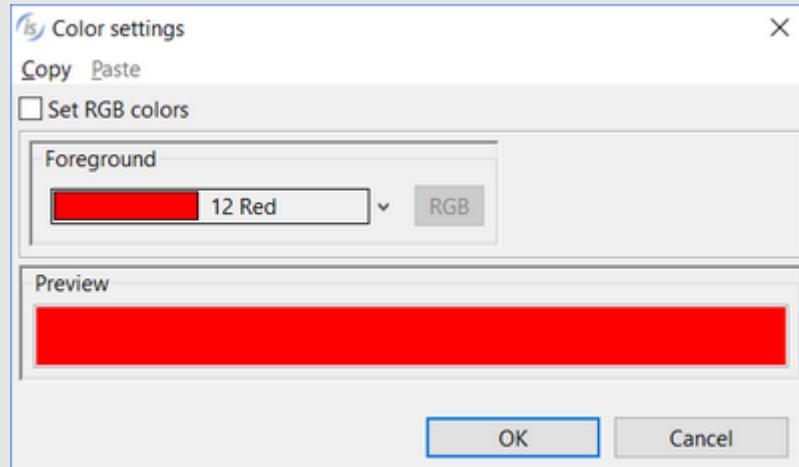
Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.

background-color Opens a dialog that allows the user to choose the control background color.

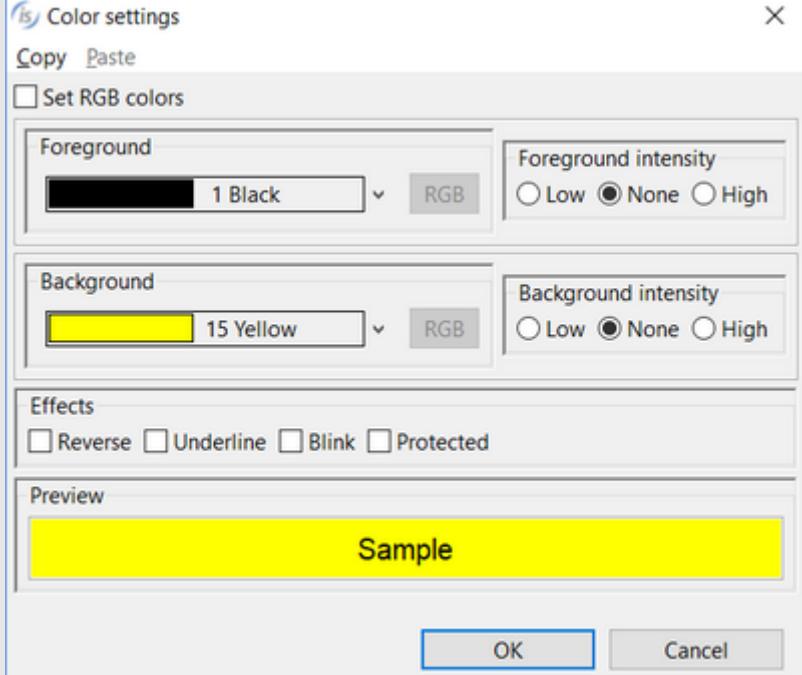


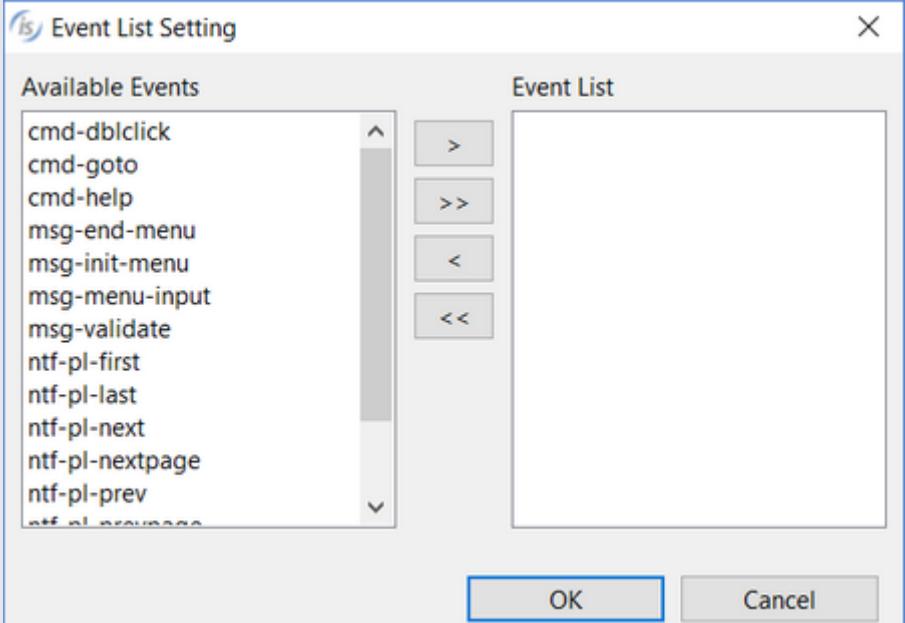
border Allows the user to set one of the following three styles:
3-D
BOXED
NO-BOX

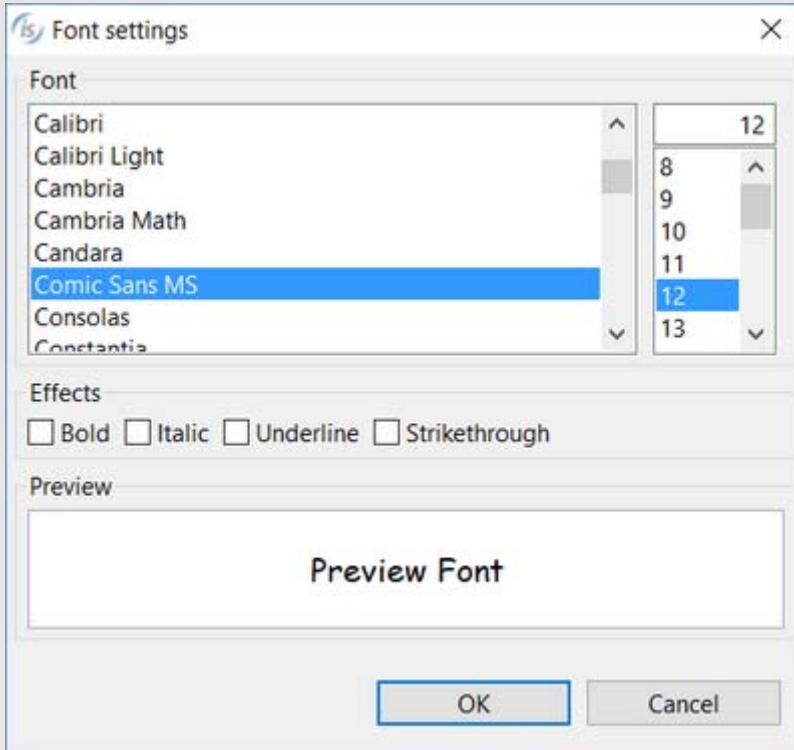
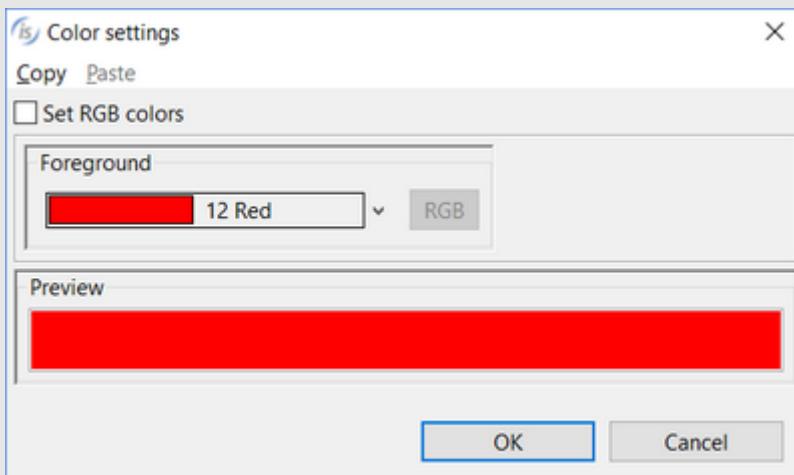
border-color Opens a dialog that allows the user to choose the control border color.

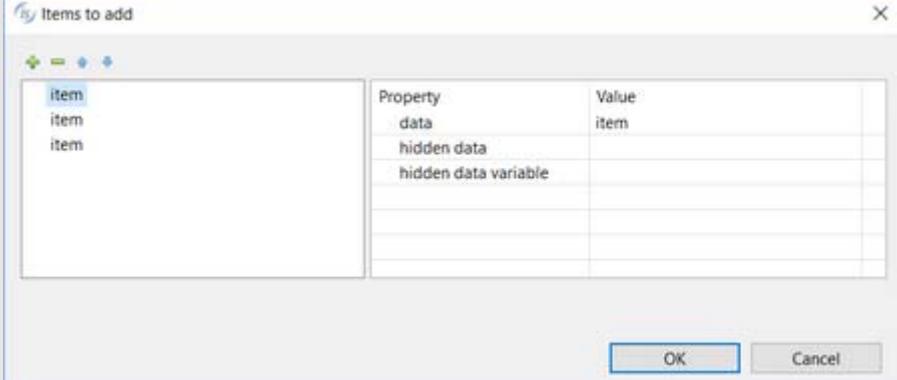
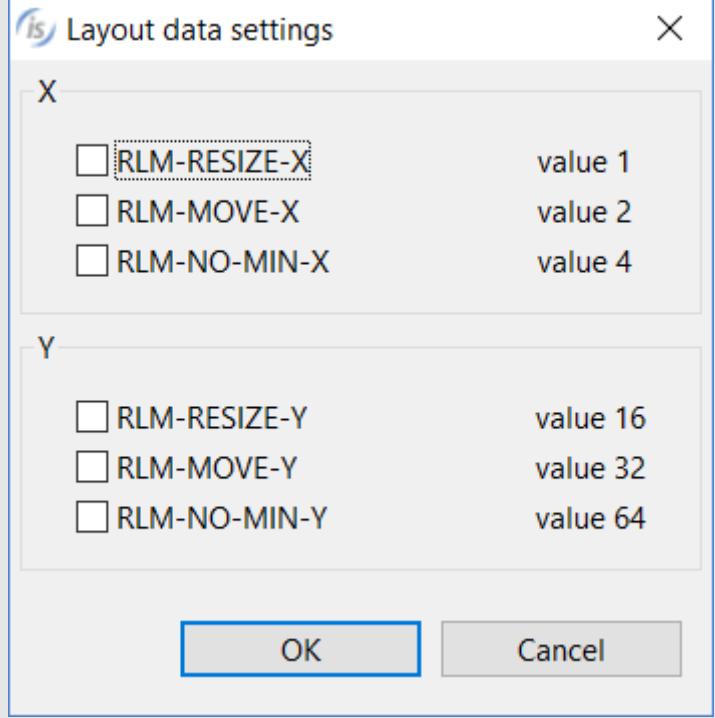


case None....Neither the *Upper* nor *Lower* styles are generated
UPPER....*Upper* style is generated
LOWER....*Lower* style is generated

color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor <i>Permanent</i> styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main panes: "Available Events" on the left and "Event List" on the right. The "Available Events" pane contains a list of event names: cmd-dblclick, cmd-goto, cmd-help, msg-end-menu, msg-init-menu, msg-menu-input, msg-validate, ntf-pl-first, ntf-pl-last, ntf-pl-next, ntf-pl-nextpage, ntf-pl-prev, and ntf-pl-recursion. Between the two panes are four buttons: '>', '>>', '<', and '<<'. At the bottom are "OK" and "Cancel" buttons.</p>
exception-value	Specifies the value for the <i>Exception-Value</i> property
exclude event list	<p>NONE... The <i>Exclude-Event-List</i> property is not generated.</p> <p>0... <i>Exclude-Event-List=0</i> is generated.</p> <p>1... <i>Exclude-Event-List=1</i> is generated.</p>
export file format	Specifies the value for the <i>Export-File-Format</i> property.
export file name	Specifies the value for the <i>Export-File-Name</i> property.

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. On the right, there's a vertical font size selector with values 8, 9, 10, 11, 12, and 13, where '12' is selected. Below the font list are sections for 'Effects' (Bold, Italic, Underline, Strikethrough) and 'Preview' (a text area containing 'Preview Font'). At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It includes a 'Set RGB colors' checkbox, which is unchecked. Below it is a 'Foreground' color picker set to '12 Red'. A preview window shows a red rectangle. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property
id	Specifies the control id. This property is set automatically when the control is drawn.

item-to-add	Opens a dialog that allows the user to set text and icons for each single item
	
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn

lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
mass-update	TRUE... <i>Mass-Update=1</i> is generated FALSE... <i>Mass-Update</i> property is not generated
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
notify dblclick	TRUE...The <i>Notify-Dblclick</i> style is generated FALSE...The <i>Notify-Dblclick</i> style is not generated
notify selchange	TRUE...The <i>Notify-Selchange</i> style is generated FALSE...The <i>Notify-Selchange</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
selection index	Specifies the value for the <i>Selection-Index</i> property
selection mode	SINGLE-SELECTION...Shows radio buttons before items. Only one item can be selected. MULTIPLE-SELECTION...Shows check boxes before items. Multiple items can be selected.
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
termination-value	Specifies the value for the <i>Termination-Value</i> property
unsorted	TRUE... The <i>Unsorted</i> style is generated FALSE... The <i>Unsorted</i> style is not generated
value	Specifies the value for the <i>Value</i> property
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated

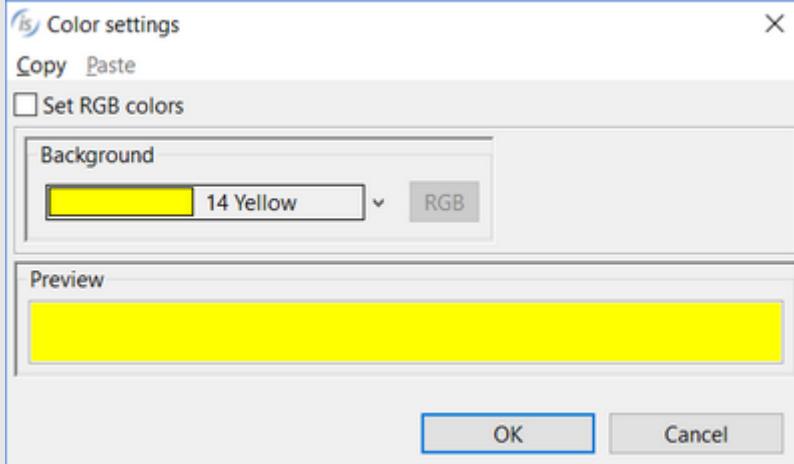
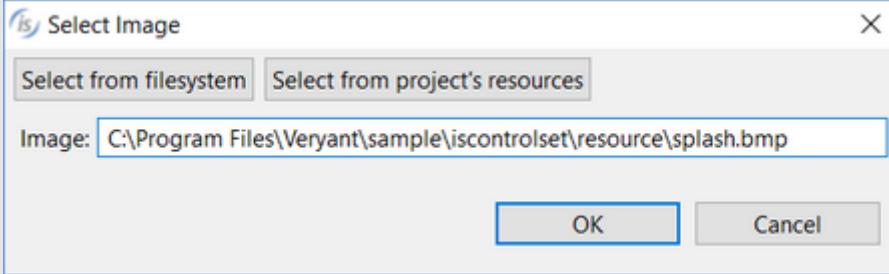
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
ntf-selchange-event	Allows the user to create a paragraph to handle the NTF-SELCHANGE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-selchange exception	Allows the user to create a paragraph to handle the NTF-SELCHANGE event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Link To	Associates a paragraph with the control that will be executed when the control is double clicked
Variables	

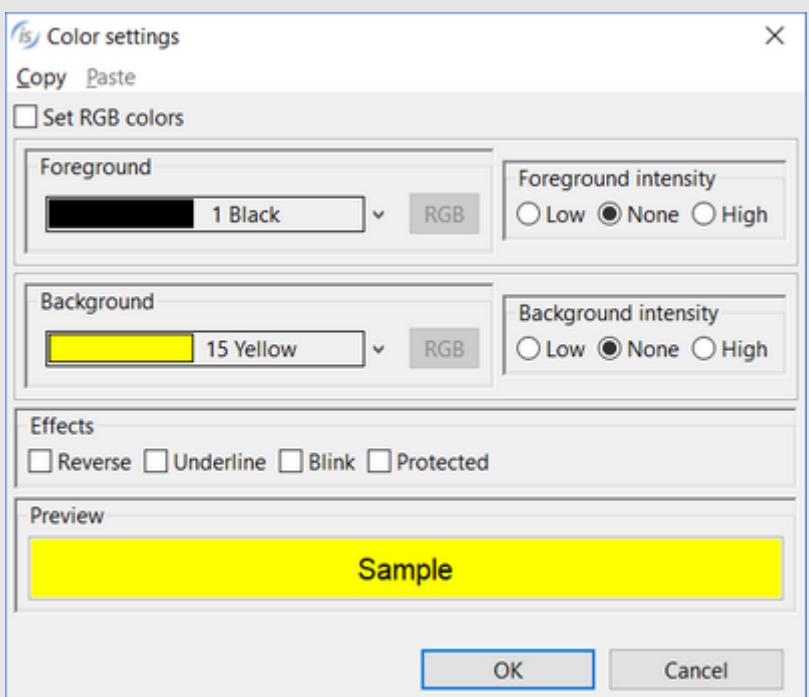
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
exception-value variable	Numeric variable that hosts the value for the <i>Exception-Value</i> property
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the value for the <i>Hint</i> property
id variable	Numeric variable that hosts the control id
item-to-add variable	Alphanumeric variable that hosts the value for the <i>Item-To-Add</i> property
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
value container	occurs item that hosts control items
value variable	Alphanumeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

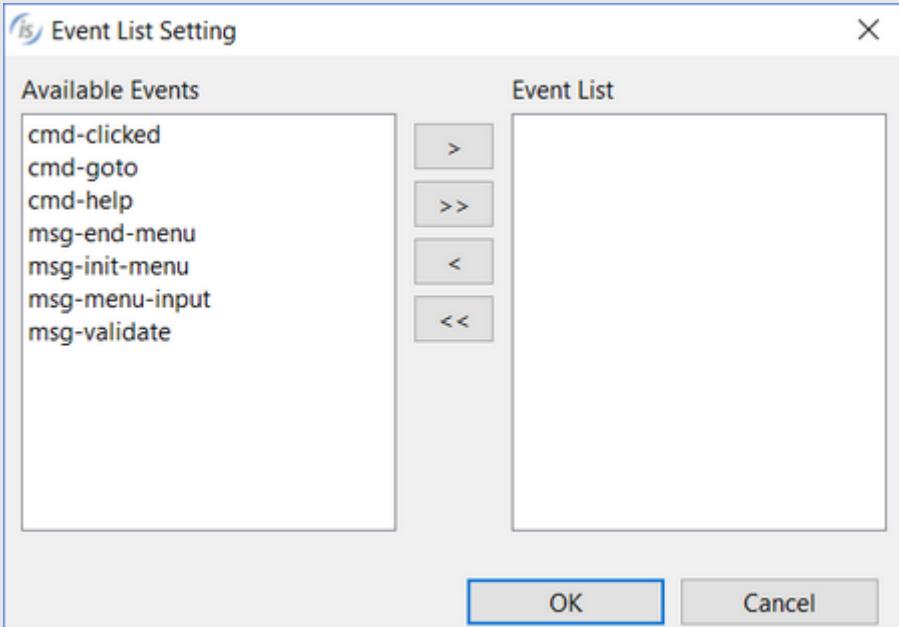
PUSH BUTTON

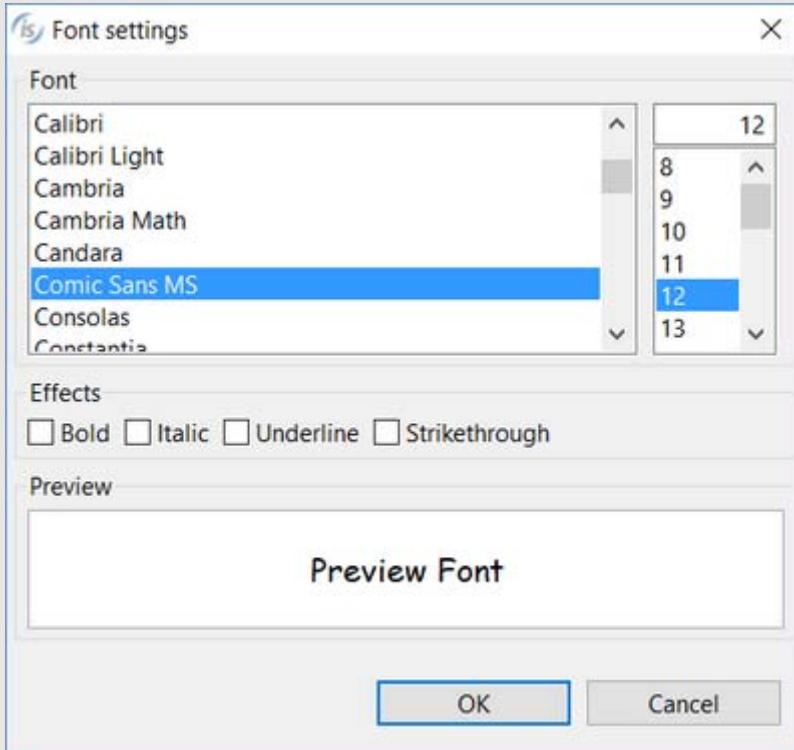
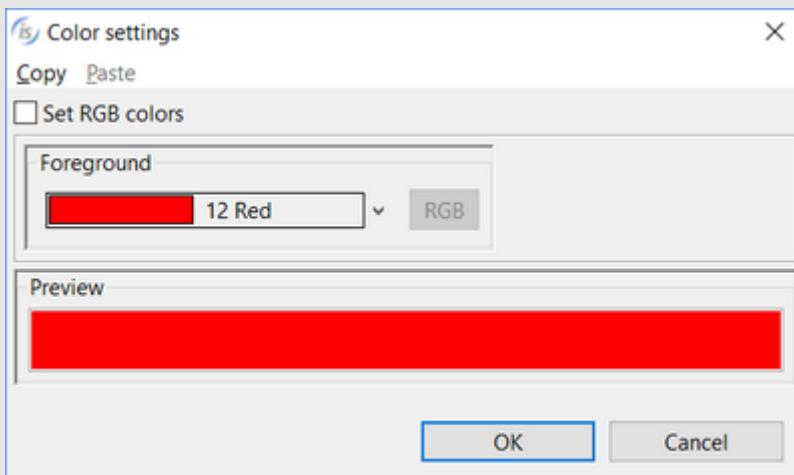
Refer to [PUSH-BUTTON](#) for details about properties, styles and events of this control.

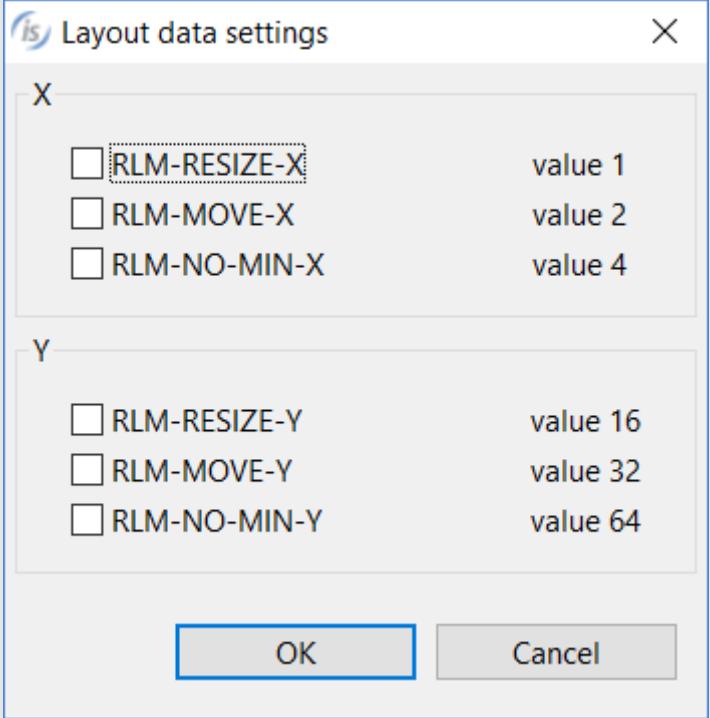
Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.

alignment	Allows to specify the alignment of the title within the button area. It works only if the button is not multiline. Possible values are: <ul style="list-style-type: none">• Top-Left• Top-Center• Top-Right• Center-Left• Center• Center-Right• Bottom-Left• Bottom-Center• Bottom-Right
auto-fit	TRUE... The image is scaled in order to fit the control boundaries. This is achieved through additional code generated in the Procedure Division. FALSE...The image is shown as is, if it's too large for the control boundaries, it will be truncated.
background-color	Opens a dialog that allows the user to choose the control background color. 
bitmap	Opens a dialog box that allows the user to select an image file to load into the control 
bitmap-disabled	Specifies the value for the <i>Bitmap-Disabled</i> property
bitmap-frame	NONE... Neither <i>Framed</i> nor <i>Unframed</i> are generated FRAMED...The <i>Framed</i> style is generated UNFRAMED...The <i>Unframed</i> style is generated

bitmap-number	Specifies the value for the <i>Bitmap-Number</i> property
bitmap-pressed	Specifies the value for the <i>Bitmap-Pressed</i> property
bitmap-rollover	Specifies the value for the <i>Bitmap-Rollover</i> property
bitmap-rollover-selected	Specifies the value for the <i>Bitmap-Rollover-Selected</i> property
bitmap-selected	Specifies the value for the <i>Bitmap-Selected</i> property
bitmap-square	TRUE...The <i>Square</i> style is generated FALSE...The <i>Square</i> style is not generated
bitmap-width	Specifies the value for the <i>Bitmap-Width</i> property
color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment.
css-icon	See Customize the EIS WD2 Layout using CSS for more information.
css-style-name	
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor Permanent styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated

enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. The "Available Events" section contains a list of event names: cmd-clicked, cmd-goto, cmd-help, msg-end-menu, msg-init-menu, msg-menu-input, and msg-validate. Between the two sections are four buttons: > (top), >> (middle), < (bottom), and << (bottom). At the bottom of the dialog are "OK" and "Cancel" buttons.</p>
exception-value	Specifies the value for the <i>Exception-Value</i> property
exclude event list	NONE... The <i>Exclude-Event-List</i> property is not generated. 0... <i>Exclude-Event-List=0</i> is generated. 1... <i>Exclude-Event-List=1</i> is generated.
flat	TRUE...The <i>Flat</i> style is generated FALSE...The <i>Flat</i> style is not generated

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS (which is highlighted with a blue selection bar), Consolas, and Constantia. To the right of the font list is a vertical scroll bar and a numeric font size selector with values 8, 9, 10, 11, 12 (which is also highlighted with a blue selection bar), and 13. Below the font list are sections for 'Effects' (checkboxes for Bold, Italic, Underline, and Strikethrough) and 'Preview' (a preview window showing the text 'Preview Font' in the selected font and size). At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It includes a 'Copy' and 'Paste' button, a checkbox for 'Set RGB colors', and a 'Foreground' color picker set to '12 Red'. A preview window below shows a red rectangle. At the bottom are 'OK' and 'Cancel' buttons.</p>
format-picture	Specifies the content for <i>Pic</i> property. Provide a valid COBOL picture or there will be errors at compile time
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE... The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property

id	Specifies the control id. This property is set automatically when the control is drawn.
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules. 
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells

min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
multiline	TRUE...The <i>Multiline</i> style is generated FALSE...The <i>Multiline</i> style is not generated
no-auto-default	TRUE...The <i>No-Auto-Default</i> style is generated FALSE...The <i>No-Auto-Default</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
self-act	TRUE...The <i>Self-Act</i> style is generated FALSE...The <i>Self-Act</i> style is not generated
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
style	DEFAULT-BUTTON CANCEL-BUTTON ESCAPE-BUTTON OK-BUTTON
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
termination-value	Specifies the value for the <i>Termination-Value</i> property
title	Specifies the value for the <i>Title</i> property
title-position	0...NONE 1...LEFT 2...RIGHT 3...TOP 5...CENTER 6...TOP-LEFT 7...TOP-CENTER 8...TOP-RIGHT 9...CENTER-LEFT 10...CENTER-RIGHT 11...BOTTOM-LEFT 12...BOTTOM-CENTER 13...BOTTOM-RIGHT

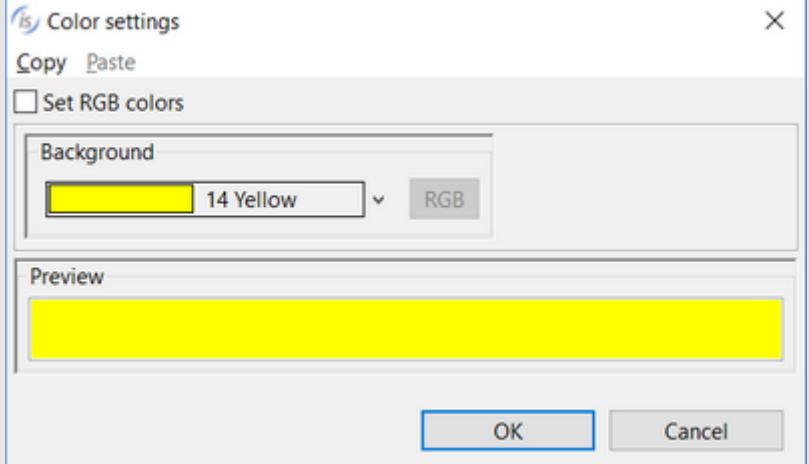
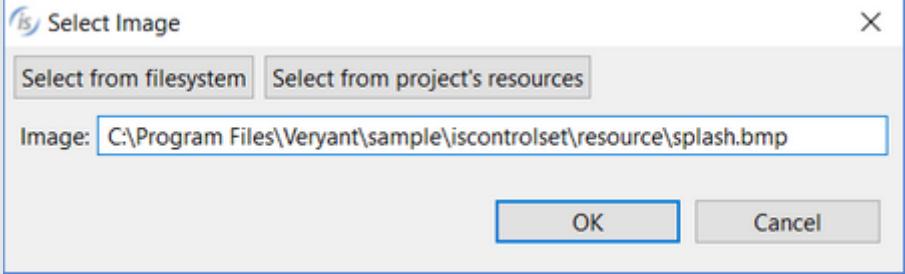
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-clicked event	Allows the user to create a paragraph to handle the CMD-CLICKED event in the Procedure Division
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-clicked exception	Allows the user to create a paragraph to handle the CMD-CLICKED event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE

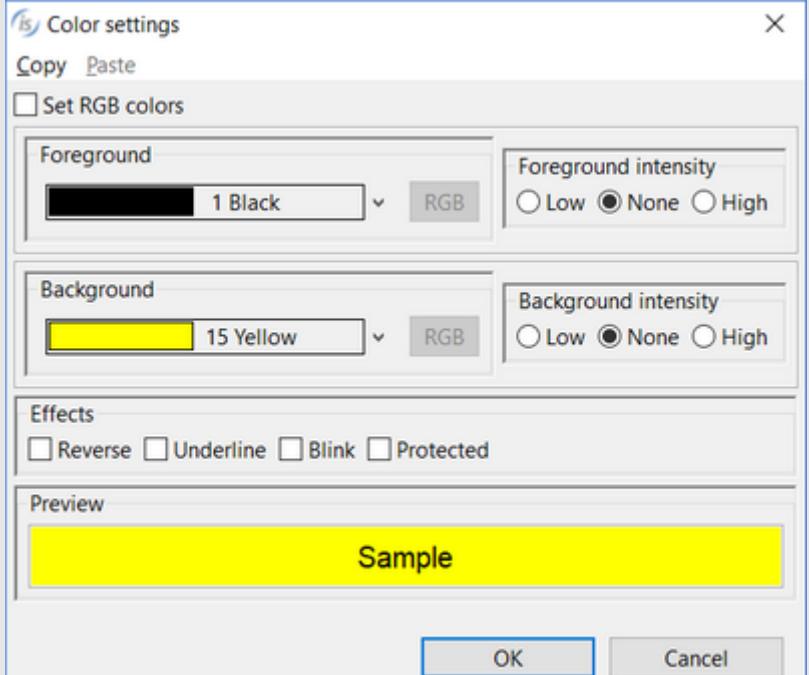
Link To	Associates a paragraph with the control that will be executed when the button is clicked
Variables	
bitmap-disabled variable	Numeric variable that hosts the value for the <i>Bitmap-Disabled</i> property
bitmap-number variable	Numeric variable that hosts the value for the <i>Bitmap-Number</i> property
bitmap-pressed value	Numeric variable that hosts the value for the <i>Bitmap-Pressed</i> property
bitmap-rollover-selected variable	Numeric variable that hosts the value for the <i>Bitmap-Rollover-Selected</i> property
bitmap-rollover variable	Numeric variable that hosts the value for the <i>Bitmap-Rollover</i> property
bitmap-selected variable	Numeric variable that hosts the value for the <i>Bitmap-Selected</i> property
bitmap-width variable	Numeric variable that hosts the value for the <i>Bitmap-Width</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
exception-value variable	Numeric variable that hosts the value for the <i>Exception-Value</i> property
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the value for the <i>Hint</i> property
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
title variable	Numeric variable that hosts the value for the <i>Title</i> property
title-position variable	Numeric variable that hosts the value for the <i>Title-Position</i> property
value variable	Numeric variable that hosts the value for the <i>Value</i> property

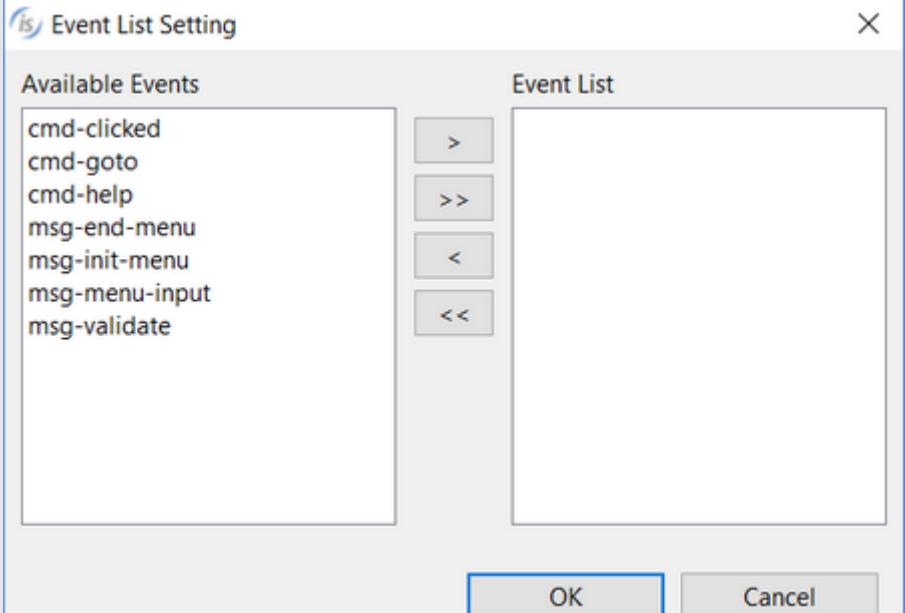
visible variable	Numeric variable that hosts the visible state
------------------	---

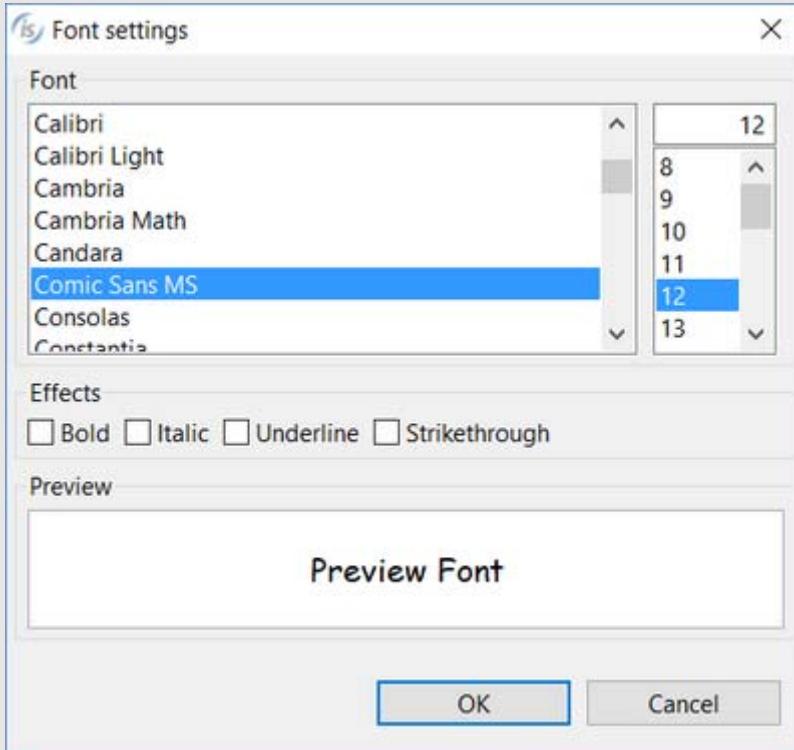
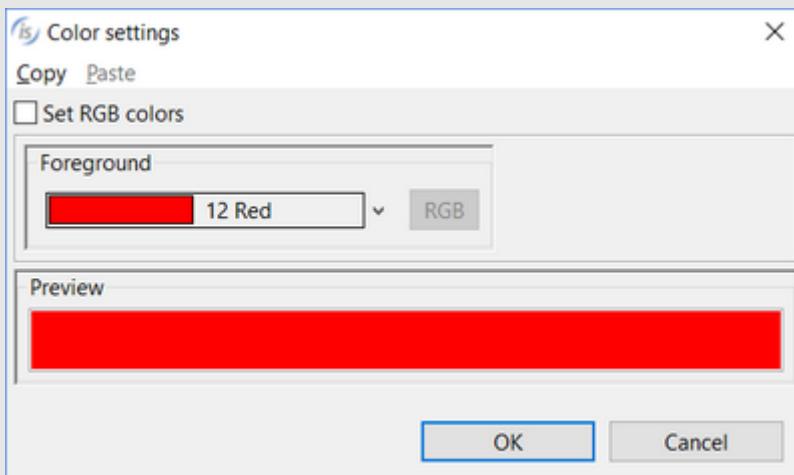
RADIO BUTTON

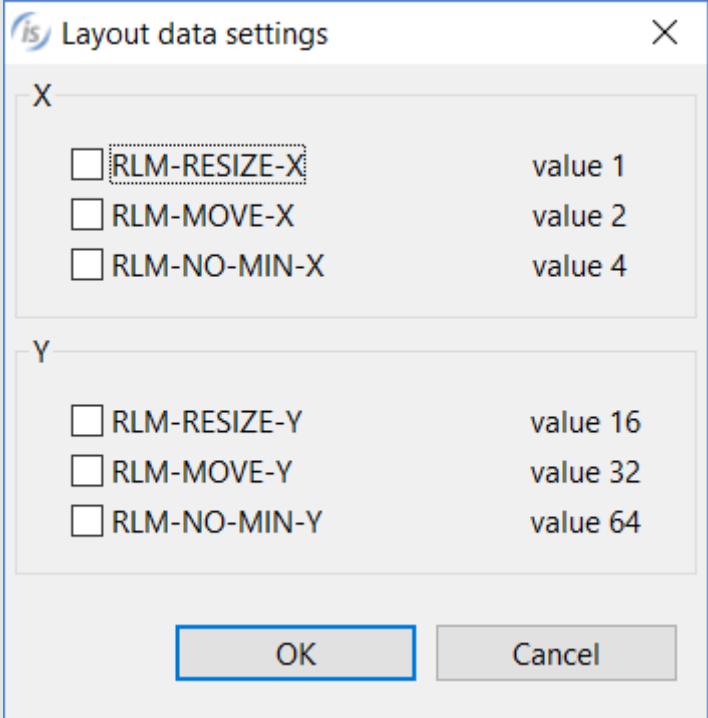
Refer to [RADIO-BUTTON](#) for details about properties, styles and events of this control.

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
auto-fit	TRUE... The image is scaled in order to fit the control's boundaries. This is achieved through additional code generated in the Procedure Division. FALSE...The image is shown as is, if it's too large for control boundaries, it will be truncated.
background-color	Opens a dialog that allows the user to choose the control background color.
	
bitmap	Opens a dialog box that allows the user to select an image file to load into the control
	
bitmap-disabled	Specifies the value for the <i>Bitmap-Disabled</i> property
bitmap-disabled-selected	Specifies the value for the <i>Bitmap-Disabled-Selected</i> property

bitmap-frame	NONE... Neither <i>Framed</i> nor <i>Unframed</i> are generated FRAMED...The <i>Framed</i> style is generated UNFRAMED...The <i>Unframed</i> style is generated
bitmap-number	Specifies the value for the <i>Bitmap-Number</i> property
bitmap-pressed	Specifies the value for the <i>Bitmap-Pressed</i> property
bitmap-rollover	Specifies the value for the <i>Bitmap-Rollover</i> property
bitmap-rollover-selected	Specifies the value for the <i>Bitmap-Rollover-Selected</i> property
bitmap-selected	Specifies the value for the <i>Bitmap-Selected</i> property
bitmap-square	TRUE...The <i>Square</i> style is generated FALSE...The <i>Square</i> style is not generated
bitmap-width	Specifies the value for the <i>Bitmap-Width</i> property
color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.

destroy type	AUTOMATIC...neither the <i>Temporary</i> nor Permanent styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. The "Available Events" section contains a list of event names: cmd-clicked, cmd-goto, cmd-help, msg-end-menu, msg-init-menu, msg-menu-input, and msg-validate. Between the two sections are four buttons: > (move selected item to Event List), >> (move all items to Event List), < (move selected item from Event List to Available Events), and << (move all items from Event List to Available Events). At the bottom are "OK" and "Cancel" buttons.</p>
exception-value	Specifies the value for the <i>Exception-Value</i> property
exclude event list	NONE... The <i>Exclude-Event-List</i> property is not generated. 0... <i>Exclude-Event-List=0</i> is generated. 1... <i>Exclude-Event-List=1</i> is generated.
flat	TRUE...The <i>Flat</i> style is generated FALSE...The <i>Flat</i> style is not generated

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical scroll bar and a size selector with values 8, 9, 10, 11, 12 (selected), and 13. Below the font list are sections for 'Effects' (Bold, Italic, Underline, Strikethrough) and 'Preview' (a text area containing 'Preview Font'). At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It includes a 'Copy' and 'Paste' button, a checkbox for 'Set RGB colors', and a 'Foreground' color picker set to '12 Red'. A 'RGB' button is next to the picker. Below the picker is a 'Preview' area showing a red rectangle. At the bottom are 'OK' and 'Cancel' buttons.</p>
group	Specifies the value for the <i>Group</i> property
group-value	Specifies the value for the <i>Group-Value</i> property
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .

hint	Specifies the value for the <i>Hint</i> property
id	Specifies the control id. This property is set automatically when the control is drawn.
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	
left-text	TRUE...The <i>Left-Text</i> style is generated FALSE...The <i>Left-Text</i> style is not generated
left-text-alignment	LEFT...Generates <i>Left-Text-Alignment</i> =1 RIGHT...Generates <i>Left-Text-Alignment</i> =0
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value

lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
multiline	TRUE...The <i>Multiline</i> style is generated FALSE...The <i>Multiline</i> style is not generated
no-group-tab	TRUE...The <i>No-Group-Tab</i> style is generated FALSE...The <i>No-Group-Tab</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
notify	TRUE...The <i>Notify</i> style is generated FALSE...The <i>Notify</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
self-act	TRUE...The <i>Self-Act</i> style is generated FALSE...The <i>Self-Act</i> style is not generated
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
termination-value	Specifies the value for the <i>Termination-Value</i> property
title	Specifies the value for the <i>Title</i> property
title-position	0...NONE 1...LEFT 2...RIGHT 3...TOP 5...CENTER
transparent	TRUE...The <i>Transparent</i> style is generated FALSE...The <i>Transparent</i> style is not generated
value	Specifies the value for the <i>Value</i> property

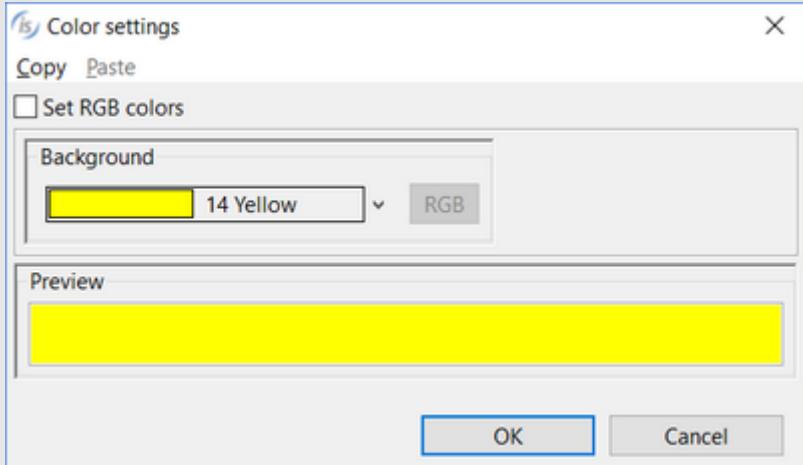
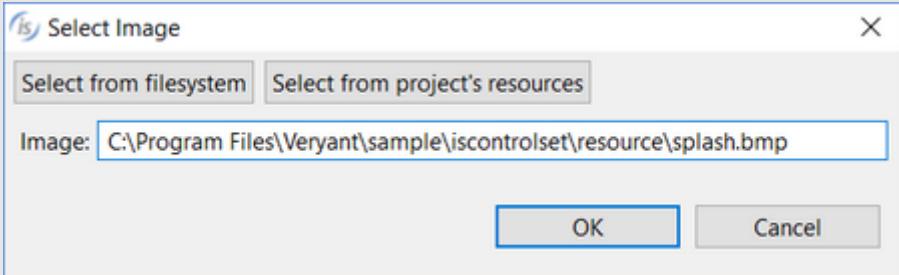
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
vtop	TRUE...The <i>Vtop</i> style is generated FALSE...The <i>Vtop</i> style is not generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-clicked event	Allows the user to create a paragraph to handle the CMD-CLICKED event in the Procedure Division
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-clicked exception	Allows the user to create a paragraph to handle the CMD-CLICKED event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE

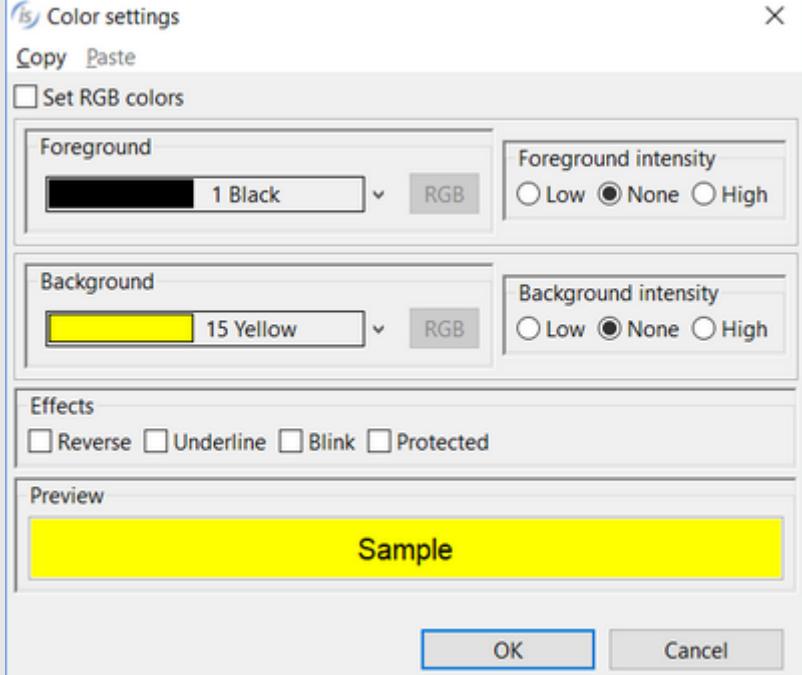
Exception procedure	Allows the user to create a paragraph to handle the control EXCEPTION PROCEDURE
Variables	
bitmap-disabled variable	Numeric variable that hosts the value for the <i>Bitmap-Disabled</i> property
bitmap-disabled-selected variable	Numeric variable that hosts the value for the <i>Bitmap-Disabled-Selected</i> property
bitmap-number variable	Numeric variable that hosts the value for the <i>Bitmap-Number</i> property
bitmap-pressed value	Numeric variable that hosts the value for the <i>Bitmap-Pressed</i> property
bitmap-rollover-selected variable	Numeric variable that hosts the value for the <i>Bitmap-Rollover-Selected</i> property
bitmap-rollover variable	Numeric variable that hosts the value for the <i>Bitmap-Rollover</i> property
bitmap-selected variable	Numeric variable that hosts the value for the <i>Bitmap-Selected</i> property
bitmap-width variable	Numeric variable that hosts the value for the <i>Bitmap-Width</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
exception-value variable	Numeric variable that hosts the value for the <i>Exception-Value</i> property
group variable	Numeric variable that hosts the value for the <i>Group</i> property
group-value variable	Numeric variable that hosts the value for the <i>Group-Value</i> property
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the value for the <i>Hint</i> property
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value

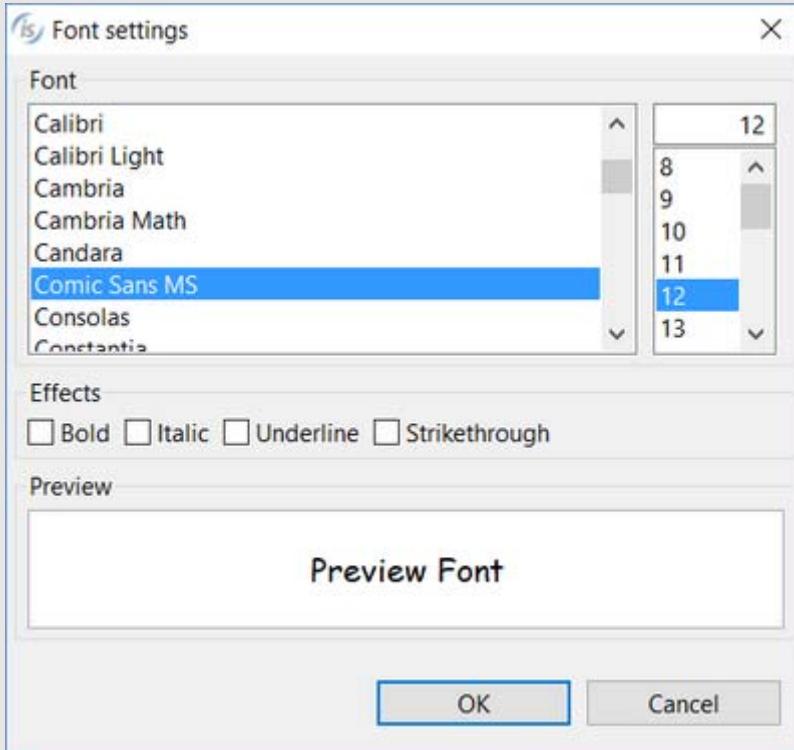
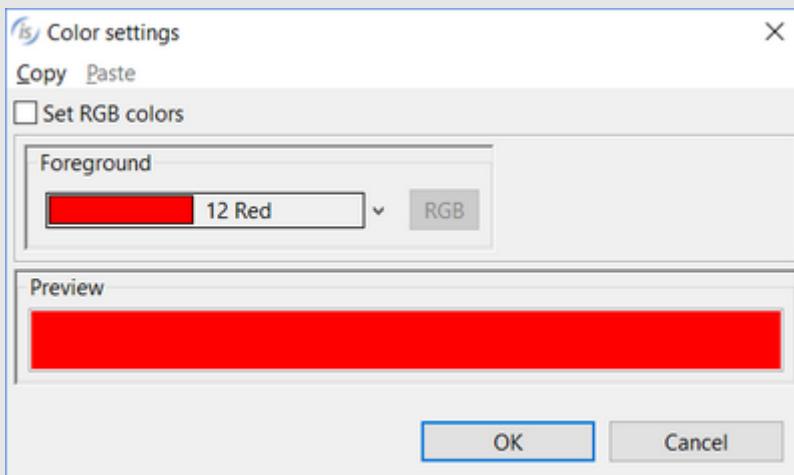
title variable	Numeric variable that hosts the value for the <i>Title</i> property
title-position variable	Numeric variable that hosts the value for the <i>Title-Position</i> property
value variable	Numeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

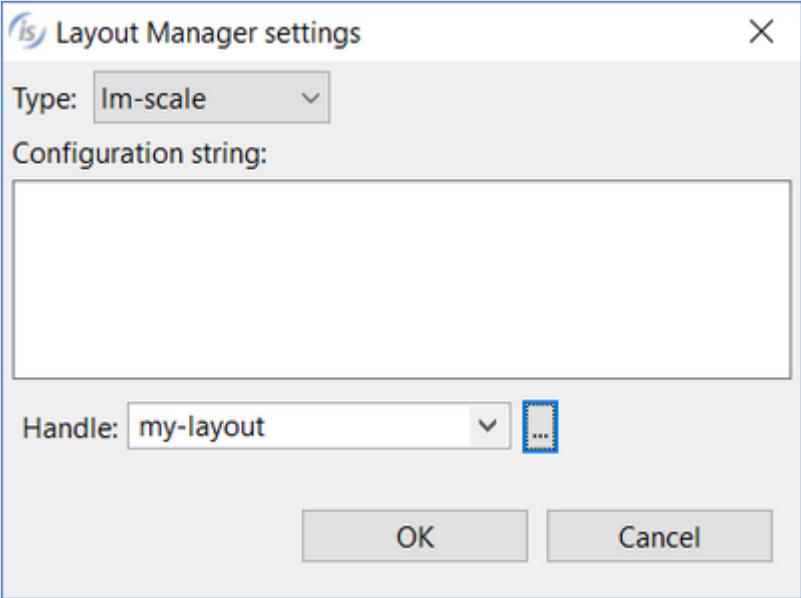
RIBBON

Refer to [RIBBON](#) for details about properties, styles and events of this control.

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
background-color	Opens a dialog that allows the user to choose the control background color. 
bitmap	Opens a dialog box that allows the user to select an image file to load into the control 
bitmap-width	Specifies the value for the <i>Bitmap-Width</i> property
collapse	TRUE... <i>collapse=1</i> is generated FALSE... <i>collapse</i> is not generated

color	Opens a dialog that allows the user to choose the control color.
	
css-base-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment.
css-style-name	See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left ('Calibri', 'Calibri Light', 'Cambria', 'Cambria Math', 'Candara', 'Comic Sans MS' (selected), 'Consolas', 'Constantia') and a font size dropdown on the right (8, 9, 10, 11, 12, 13). Below the font list are checkboxes for Bold, Italic, Underline, and Strikethrough. A preview window shows the text 'Preview Font' in the selected font and size. At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a 'Foreground' color picker set to '12 Red'. A preview window shows a red bar. At the bottom are 'OK' and 'Cancel' buttons.</p>
header-align	CENTER... <i>Header-Align</i> is not generated LEFT... <i>Header-Align</i> =1 is generated RIGHT... <i>Header-Align</i> =2 is generated
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
hint	Specifies the value for the <i>Hint</i> property

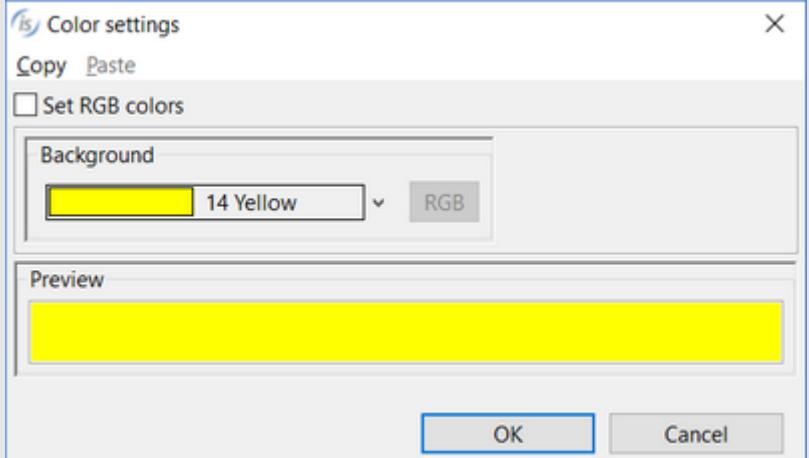
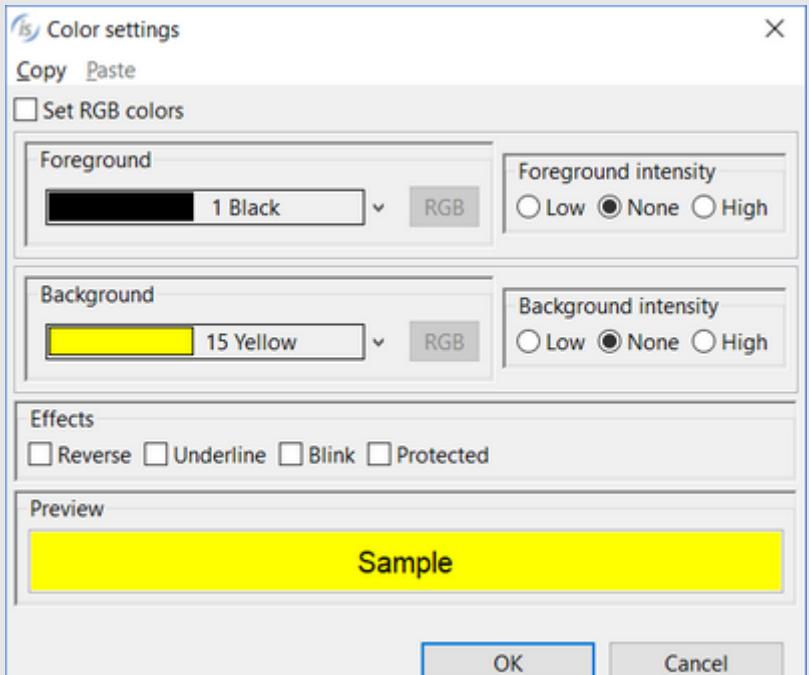
id	Specifies the control id. This property is set automatically when the control is drawn.
layout manager	Opens a dialog that allows to choose which layout manager should be associated to the ribbon. When either LM-SCALE or LM-RESPONSIVE is selected, it's possible to specify the configuration string. In this dialog you also associate a handle to the layout manager.
	
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
value	Specifies the value for the <i>Value</i> property
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-tabchanged event	Allows the user to create a paragraph to handle the CMD-TABCHANGED event in the Procedure Division

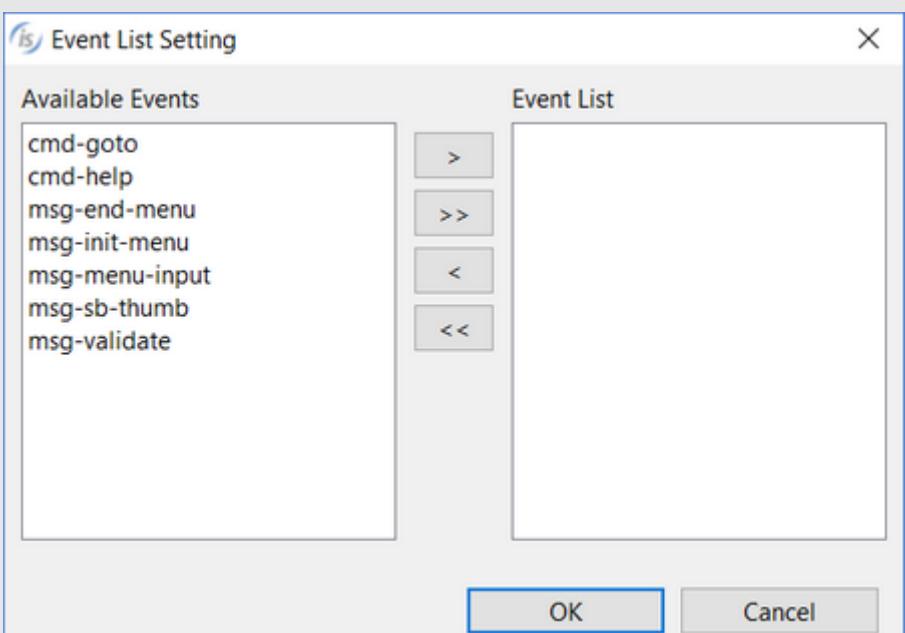
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-tabchanged exception	Allows the user to create a paragraph to handle the CMD-TABCHANGED event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Variables	
background-color variable	Numeric variable that hosts the value for the <i>Background-Color</i> property
bitmap-width variable	Numeric variable that hosts the value for the <i>Bitmap-Width</i> property
color variable	Numeric variable that hosts the color value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
foreground-color variable	Numeric variable that hosts the value for the <i>Foreground-Color</i> property
header-align variable	Numeric variable that hosts the value for the <i>Header-Align</i> property
hint variable	Alphanumeric variable that hosts the value for the <i>Hint</i> property
id variable	Numeric variable that hosts the control id
lines variable	Numeric variable that hosts the lines value
ribbon handle	Handle variable for the ribbon
value variable	Numeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

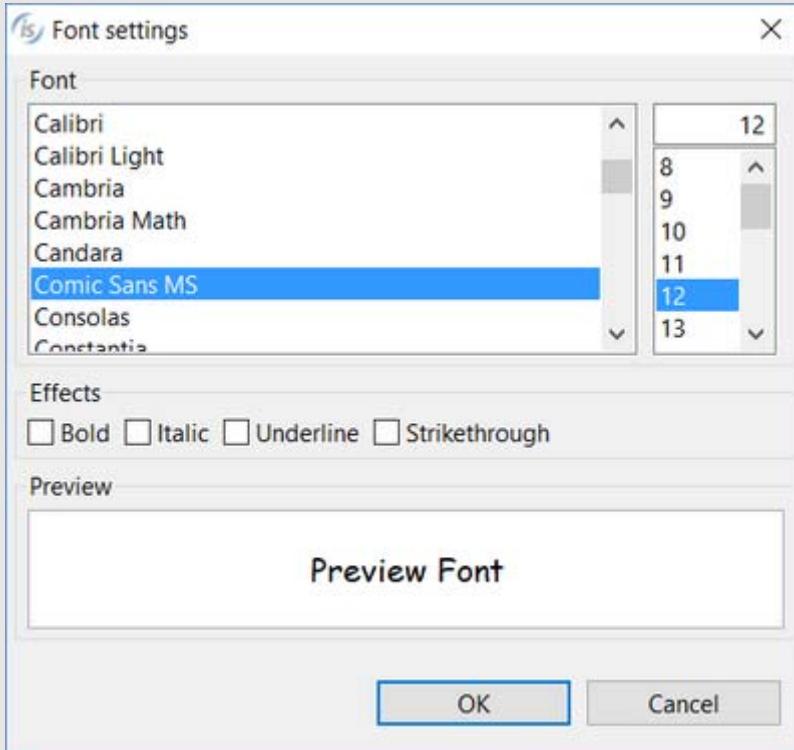
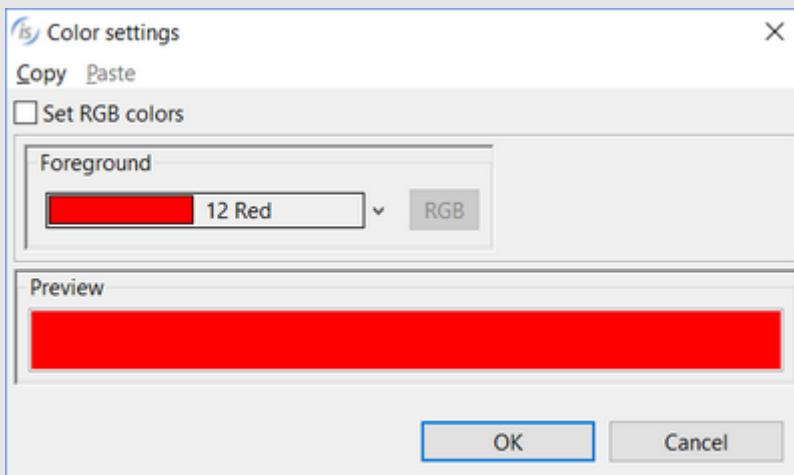
SCROLL BAR

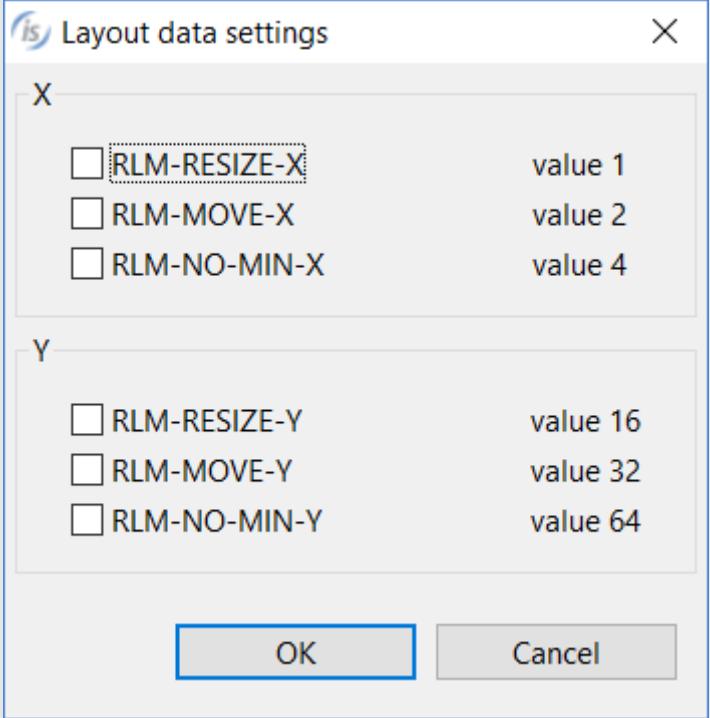
Refer to [SCROLL-BAR](#) for details about properties, styles and events of this control.

Properties

(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
background-color	Opens a dialog that allows the user to choose the control background color.  The dialog is titled 'Color settings'. It has 'Copy' and 'Paste' buttons. A checkbox 'Set RGB colors' is checked. Under 'Background', there is a color swatch showing yellow, labeled '14 Yellow', and an 'RGB' button. A preview window shows a yellow bar. At the bottom are 'OK' and 'Cancel' buttons.
color	Opens a dialog that allows the user to choose the control color.  The dialog is titled 'Color settings'. It has 'Copy' and 'Paste' buttons. A checkbox 'Set RGB colors' is checked. Under 'Foreground', there is a color swatch showing black, labeled '1 Black', and an 'RGB' button. To its right is a 'Foreground intensity' section with radio buttons for 'Low', 'None' (selected), and 'High'. Under 'Background', there is a color swatch showing yellow, labeled '15 Yellow', and an 'RGB' button. To its right is a 'Background intensity' section with radio buttons for 'Low', 'None' (selected), and 'High'. Under 'Effects', there are checkboxes for 'Reverse', 'Underline', 'Blink', and 'Protected'. A preview window shows a yellow bar with the word 'Sample'. At the bottom are 'OK' and 'Cancel' buttons.
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.

css-base-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment.
css-style-name	See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor <i>Permanent</i> styles are generated TEMPORARY... <i>Temporary</i> style is generated <i>Permanent</i> ... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. The "Available Events" section contains a list of event names: cmd-goto, cmd-help, msg-end-menu, msg-init-menu, msg-menu-input, msg-sb-thumb, and msg-validate. Between the two sections are four buttons: a top-right button with a right-pointing arrow (>), a middle-right button with a double-right-pointing arrow (>>), a bottom-right button with a left-pointing arrow (<), and a bottom-left button with a double-left-pointing arrow (<<). At the bottom of the dialog are two buttons: "OK" (highlighted with a blue border) and "Cancel".</p>	
exclude event list	NONE... The <i>Exclude-Event-List</i> property is not generated. 0... <i>Exclude-Event-List=0</i> is generated. 1... <i>Exclude-Event-List=1</i> is generated.

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical font size selector with values 8, 9, 10, 11, 12, and 13. '12' is selected. Below the font list are sections for 'Effects' (Bold, Italic, Underline, Strikethrough) and 'Preview' (a box labeled 'Preview Font'). At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It includes a 'Set RGB colors' checkbox, which is unchecked. Below it is a 'Foreground' color picker with a red square swatch labeled '12 Red' and an 'RGB' button. A 'Preview' section shows a horizontal bar filled with red. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
horizontal	TRUE...The <i>Horizontal</i> style is generated FALSE...The <i>Horizontal</i> style is not generated

id	Specifies the control id. This property is set automatically when the control is drawn.
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	<p>DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting</p> <p>None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value</p> <p><i>CELLS</i>... <i>CELLS</i> is generated after the <i>Lines</i> value</p> <p><i>PIXELS</i>... <i>PIXELS</i> is generated after the <i>Lines</i> value</p>
lock	<p>TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse.</p> <p>FALSE...You can move the control on the Screen Designer by dragging it with the mouse</p>
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells

min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
page size	Specifies the value for the <i>Page-Size</i> property
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
track-thumb	TRUE...The <i>Track-Thumb</i> style is generated FALSE... The <i>Track-Thumb</i> style is not generated
value	Specifies the value for the <i>Value</i> property
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-sb-next event	Allows the user to create a paragraph to handle the MSG-SB-NEXT event in the Procedure Division
msg-sb-nextpage event	Allows the user to create a paragraph to handle the MSG-SB-NEXTPAGE event in the Procedure Division
msg-sb-prev event	Allows the user to create a paragraph to handle the MSG-SB-PREV event in the Procedure Division

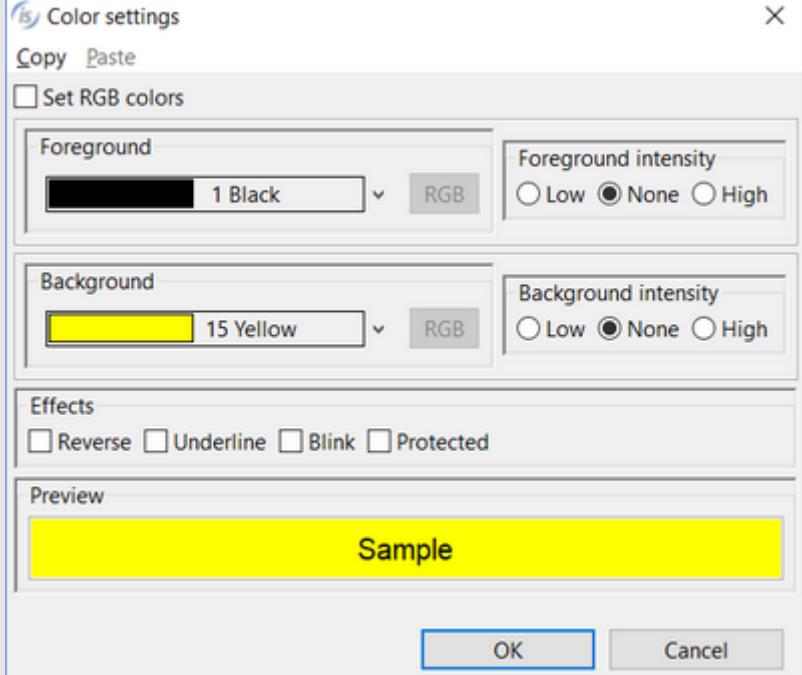
msg-sb-prevpage event	Allows the user to create a paragraph to handle the MSG-SB-PREVPAGE event in the Procedure Division
msg-sb-thumb event	Allows the user to create a paragraph to handle the MSG-SB-THUMB event in the Procedure Division
msg-sb-thumbtrack event	Allows the user to create a paragraph to handle the MSG-SB-THUMBTRACK event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Variables	
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the hint value.
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the Key property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value

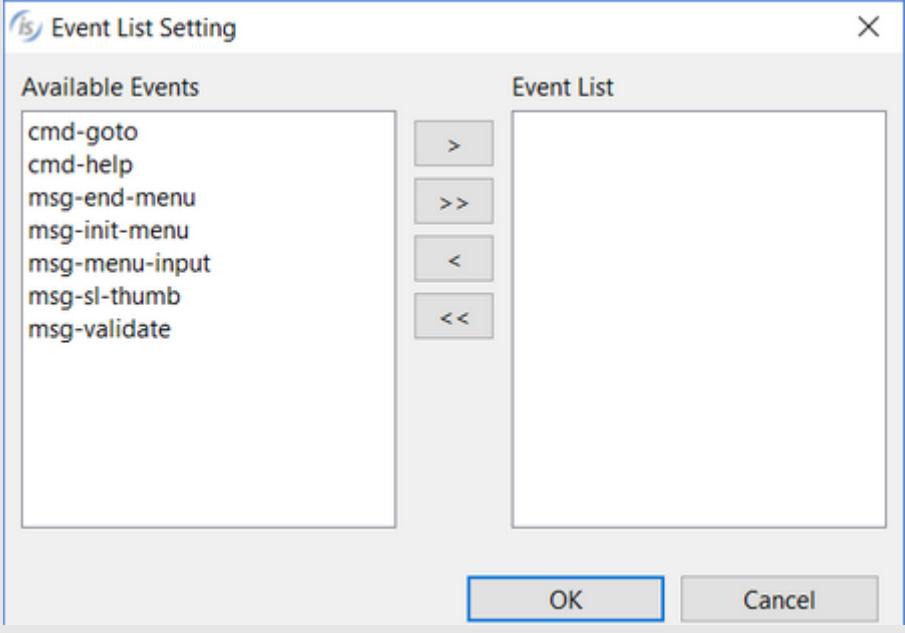
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
page size variable	Numeric variable that hosts the value for the <i>Page-Size</i> variable
size variable	Numeric variable that hosts the size value
value variable	Numeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

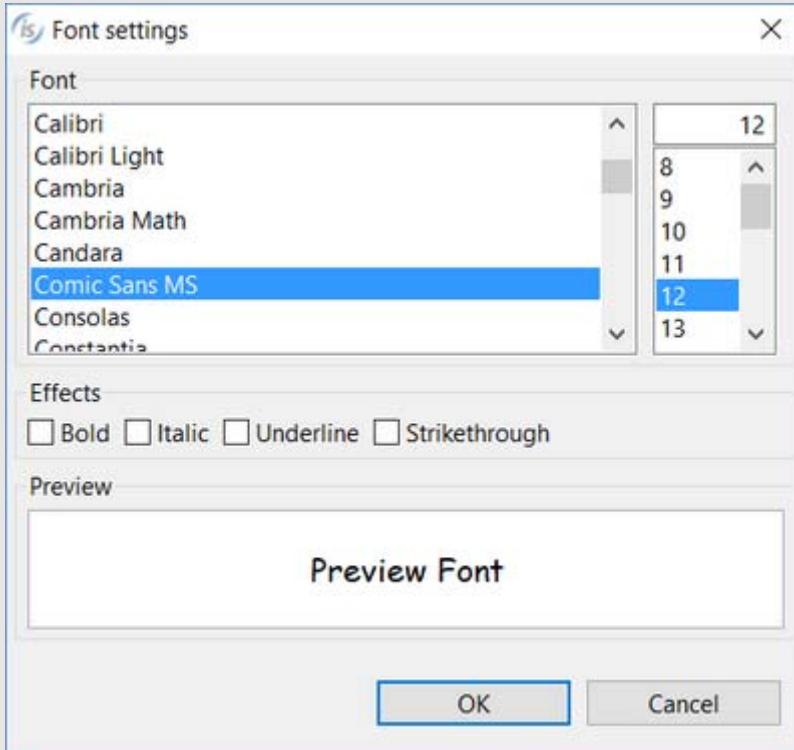
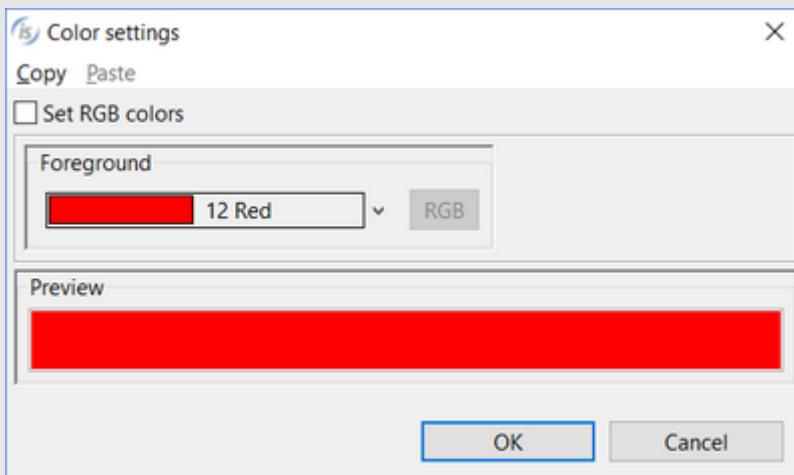
SLIDER

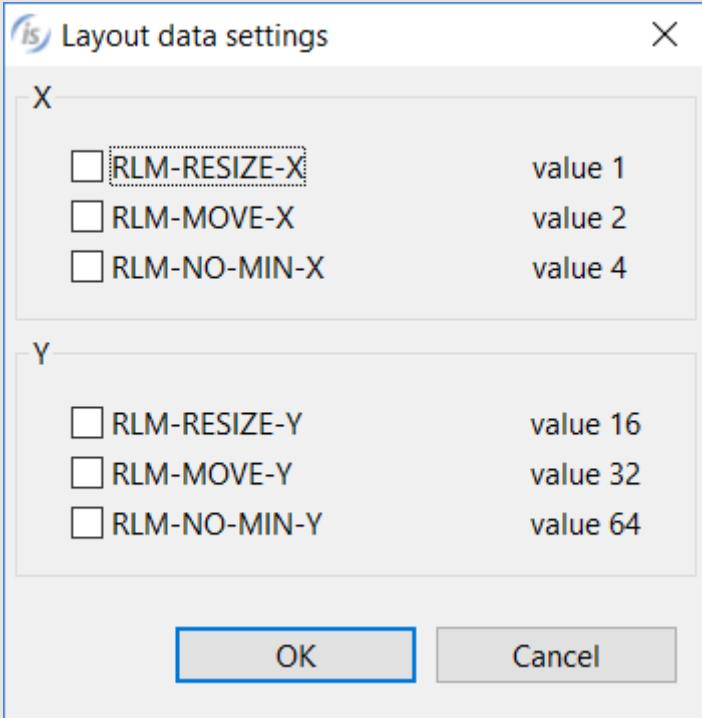
Refer to [SLIDER](#) for details about properties, styles and events of this control.

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
background-color	Opens a dialog that allows the user to choose the control background color.
	

color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor <i>Permanent</i> styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. The "Available Events" section contains a list of event names: cmd/goto, cmd/help, msg-end-menu, msg-init-menu, msg-menu-input, msg-sl-thumb, and msg-validate. Between the two sections are four buttons: '>', '>>', '<', and '<<'. At the bottom are "OK" and "Cancel" buttons. The "OK" button is highlighted with a blue border.</p>

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical size selector with values 8, 9, 10, 11, 12, and 13. '12' is selected. Below the font list are checkboxes for Bold, Italic, Underline, and Strikethrough. A preview window shows the text 'Preview Font' in a bold, italicized, underlined, and strikethrough font. At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a checkbox 'Set RGB colors' which is unchecked. Below it is a 'Foreground' color picker with a red square swatch labeled '12 Red' and an 'RGB' button. A preview window shows a red horizontal bar. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
horizontal	TRUE...The <i>Horizontal</i> style is generated FALSE...The <i>Horizontal</i> style is not generated

id	Specifies the control id. This property is set automatically when the control is drawn.
inverted	TRUE...The <i>Inverted</i> style is generated FALSE...The <i>Inverted</i> style is not generated
key	Specifies the value for the <i>Key</i> property.
labels increment	Specifies the value for the <i>Labels-Increment</i> property
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse

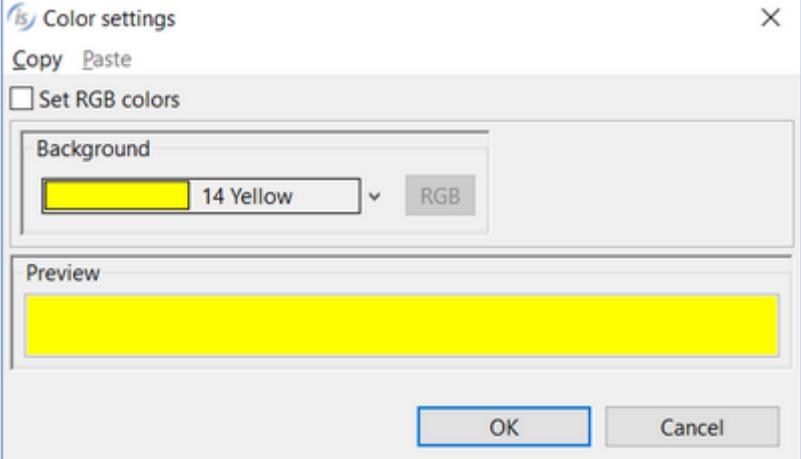
major tick spacing	Specifies the value for the <i>Major-Tick-Spacing</i> property
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
page size	Specifies the value for the <i>Page-Size</i> property
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
show labels	TRUE...The <i>Show-Labels</i> style is generated FALSE...The <i>Show-Labels</i> style is not generated
show ticks	TRUE...The <i>Show-Ticks</i> style is generated FALSE...The <i>Show-Ticks</i> style is not generated
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
value	Specifies the value for the <i>Value</i> property
visible	NONE...The <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division

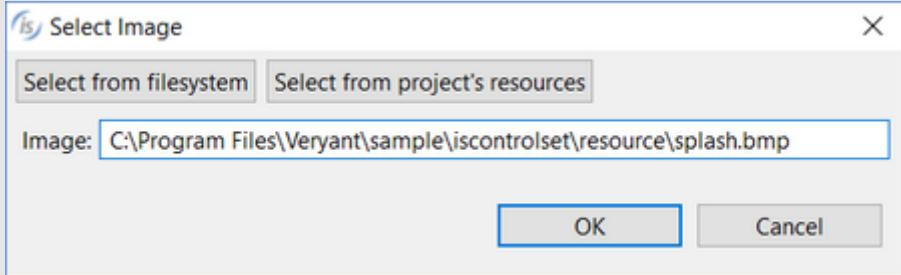
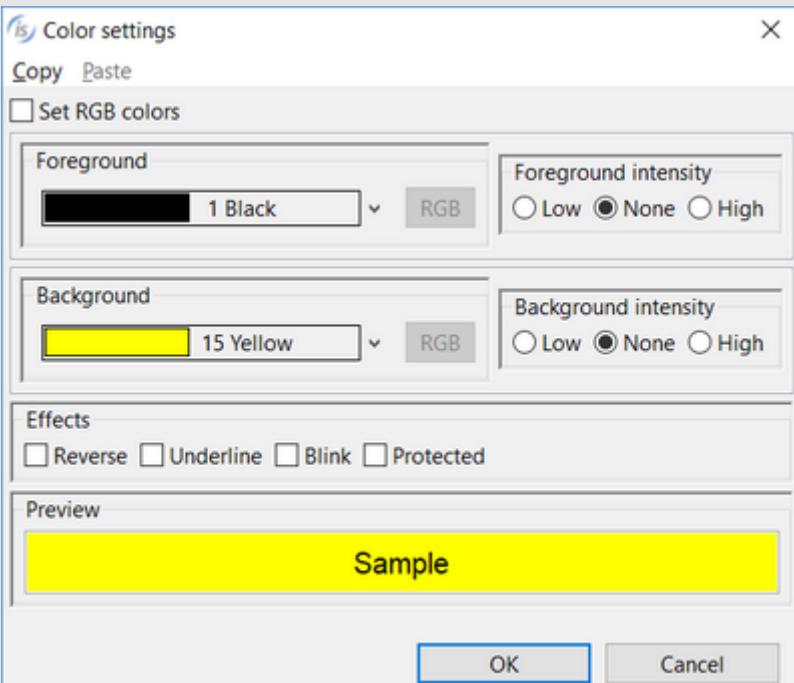
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-sl-thumb event	Allows the user to create a paragraph to handle the MSG-SL-THUMB event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Variables	
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the hint value.
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the Key property
labels increment variable	Numeric variable that hosts the value for the <i>Labels-Increment</i> property
layout-data variable	Numeric variable that hosts the control resize rules

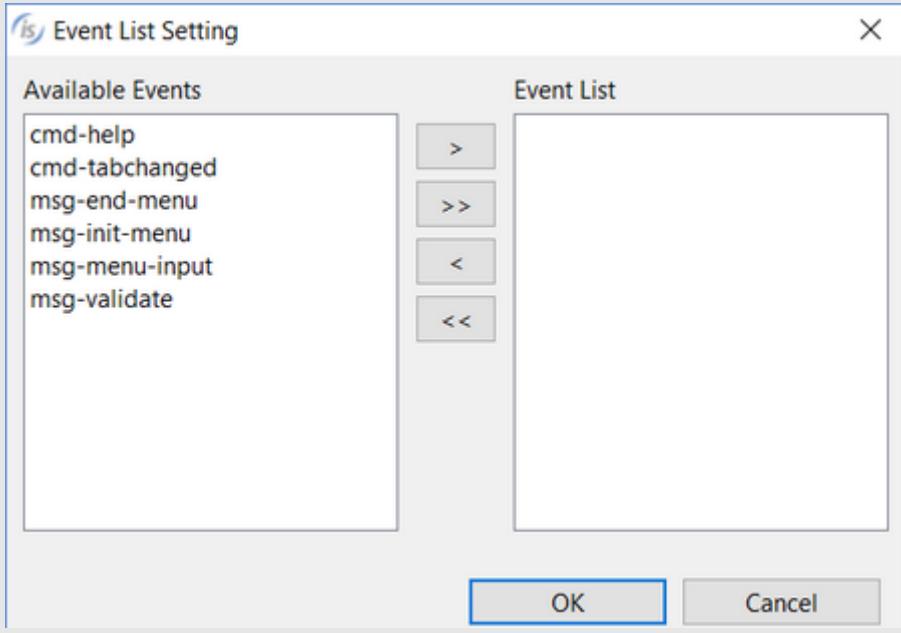
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
major tick spacing variable	Numeric variable that hosts the value for the <i>Major-Tick-Spacing</i> property
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
page size variable	Numeric variable that hosts the value for the <i>Page-Size</i> variable
size variable	Numeric variable that hosts the size value
value variable	Numeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

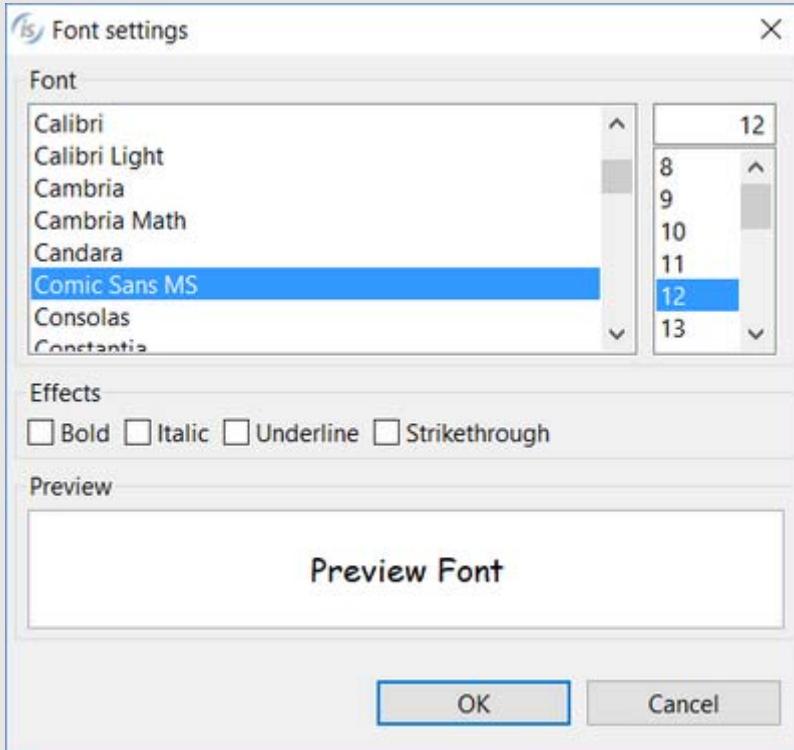
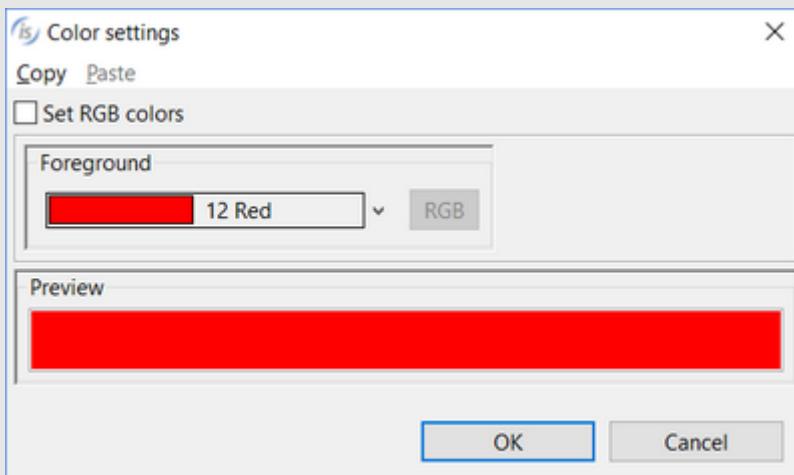
TAB ACCORDION

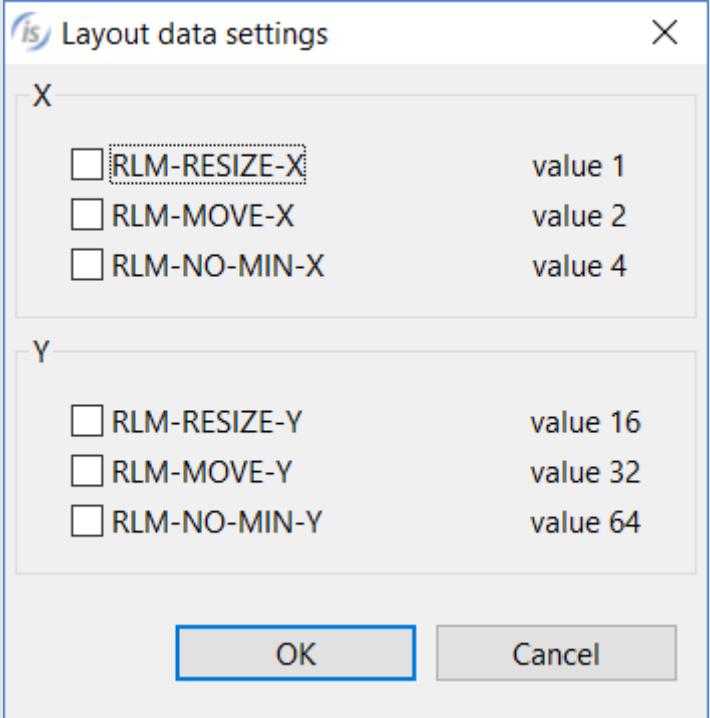
Refer to [TAB-CONTROL](#) for details about properties, styles and events of this control.

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
background-color	Opens a dialog that allows the user to choose the control background color.
	

bitmap	Opens a dialog box that allows the user to select an image file to load into the control
	
bitmap-width	Specifies the value for the <i>Bitmap-Width</i> property
color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor Permanent styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated

enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. The "Available Events" section contains a list of event names: cmd-help, cmd-tabchanged, msg-end-menu, msg-init-menu, msg-menu-input, and msg-validate. Between the two sections are four buttons: > (top), >> (middle), < (bottom), and << (bottom). At the bottom of the dialog are "OK" and "Cancel" buttons.</p>
exclude event list	NONE... The <i>Exclude-Event-List</i> property is not generated. 0... <i>Exclude-Event-List=0</i> is generated. 1... <i>Exclude-Event-List=1</i> is generated.

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a title bar with a close button ('X'). Below it is a 'Font' section containing a list of font names: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical scroll bar and a size dropdown menu showing values 8, 9, 10, 11, 12 (selected), and 13. Below the font list is an 'Effects' section with checkboxes for Bold, Italic, Underline, and Strikethrough, all of which are unchecked. Underneath that is a 'Preview' section with a text area containing the text 'Preview Font'.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a title bar with a close button ('X'). Below it is a toolbar with 'Copy' and 'Paste' buttons, and a checkbox labeled 'Set RGB colors' which is unchecked. The main area is titled 'Foreground' and contains a color swatch set to '12 Red' and a 'RGB' button. Below this is a 'Preview' section with a large red rectangular area. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
id	Specifies the control id. This property is set automatically when the control is drawn.

key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules. 
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells

min-width	Specifies the control minimum width as expressed in cells
no-dividers	TRUE...The <i>No-Dividers</i> style is generated FALSE...The <i>No-Dividers</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
relative-offset	TRUE...The <i>Relative-Offset</i> style is generated FALSE...The <i>Relative-Offset</i> style is not generated
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
value	Specifies the value for the <i>Value</i> property
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
cmd-tabchanged event	Allows the user to create a paragraph to handle the CMD-TABCHANGED event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division

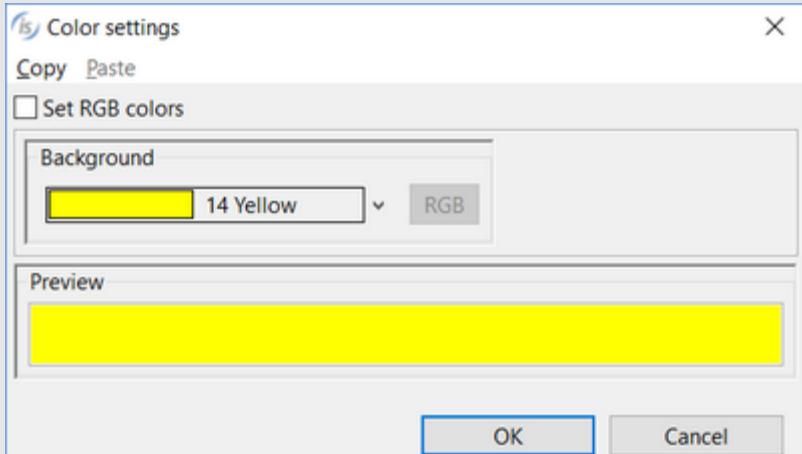
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-tabchanged	Allows the user to create a paragraph to handle the CMD-TABCHANGED event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
After-Tabchg Display	Allows the user to create a paragraph that will be executed after CMD-TABCHANGED handling
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Before-Tabchg Display	Allows the user to create a paragraph that will be executed before CMD-TABCHANGED handling
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Variables	
background-color variable	Numeric variable that hosts the value for the <i>Background-Color</i> property
bitmap-width variable	Numeric variable that hosts the value for the <i>Bitmap-Width</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the value for the <i>Hint</i> property
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property

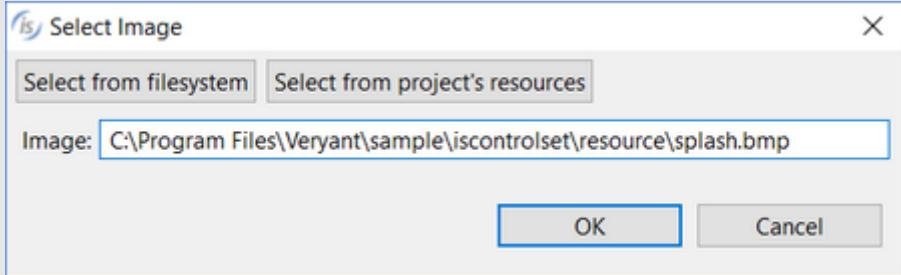
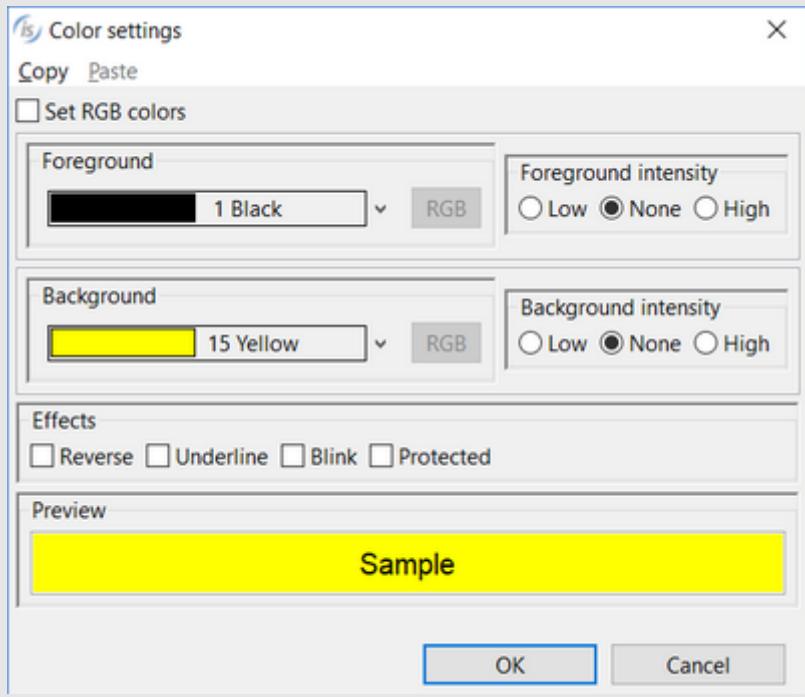
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
value variable	Numeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

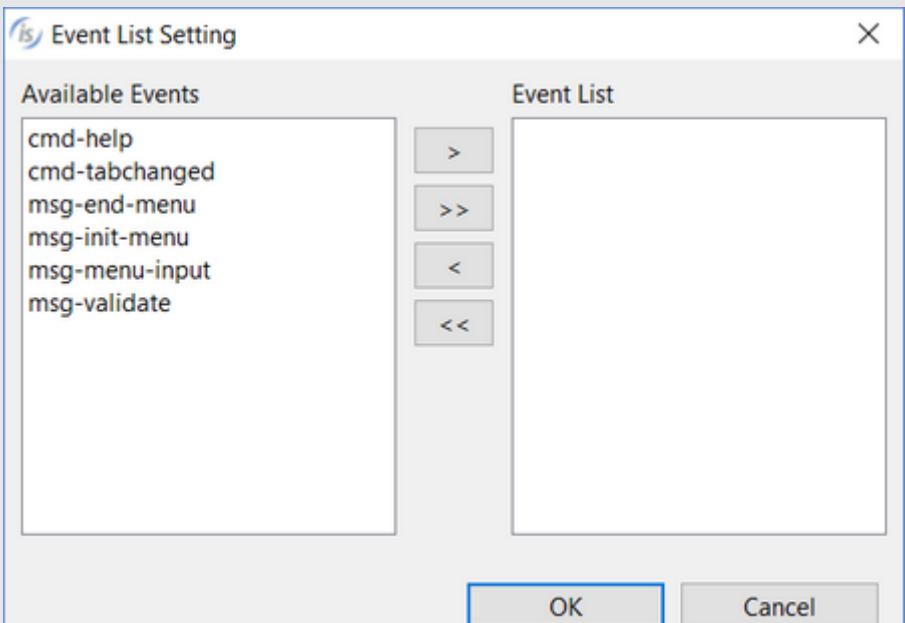
TAB CONTROL

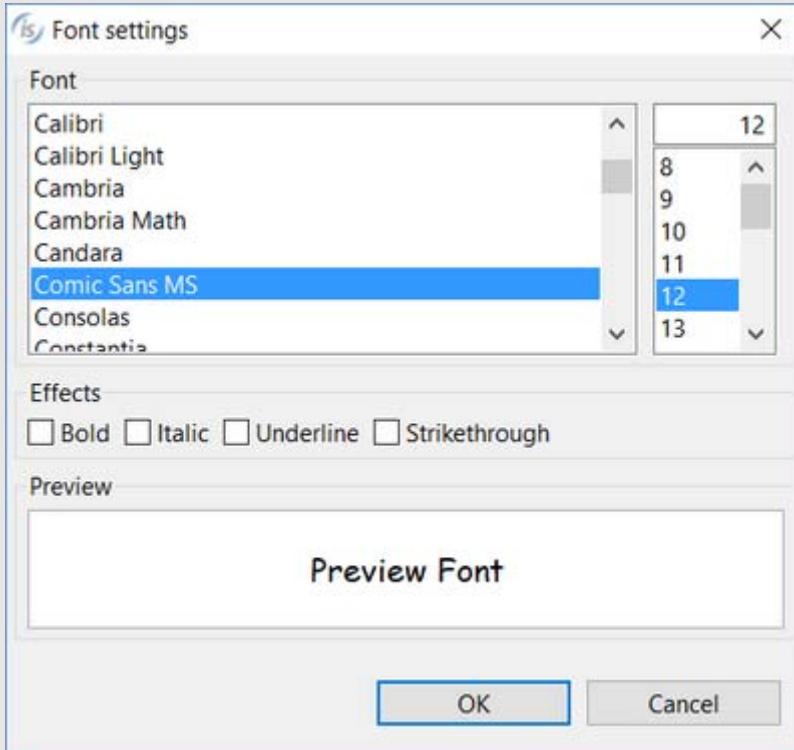
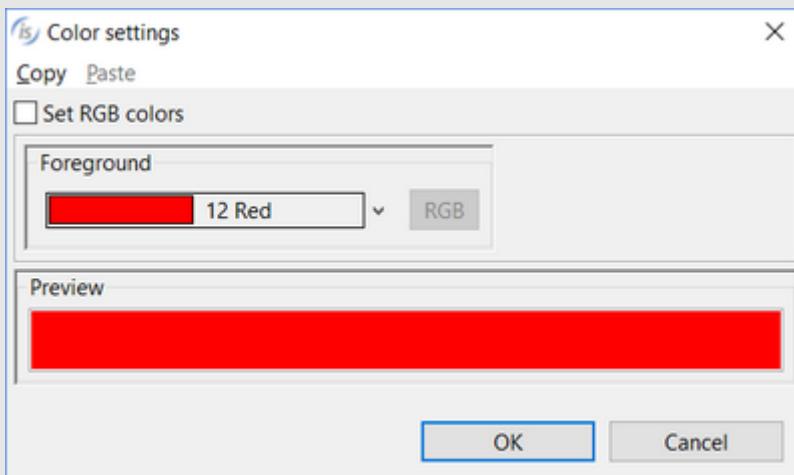
Refer to [TAB-CONTROL](#) for details about properties, styles and events of this control.

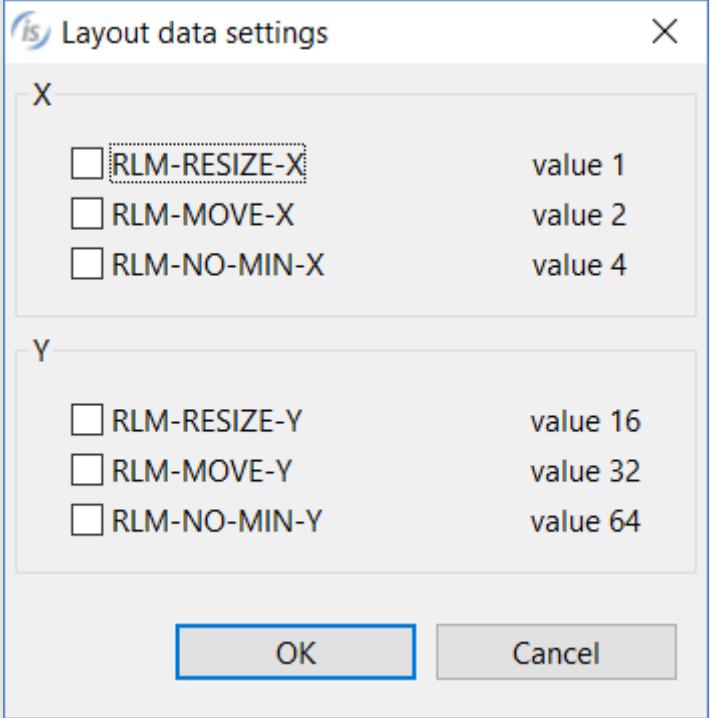
Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
allow-container	TRUE...The <i>Allow-Container</i> style is generated. As a consequence, the code generated to manage the tab switch will rely on this style. FALSE...The <i>Allow-Container</i> style is not generated. As a consequence, the code generated to manage the tab switch will rely on visible variables and CMD-TABCHANGED event.
background-color	Opens a dialog that allows the user to choose the control background color.



bitmap	Opens a dialog box that allows the user to select an image file to load into the control
	
bitmap-width	Specifies the value for the <i>Bitmap-Width</i> property
buttons	TRUE...The <i>Buttons</i> style is generated FALSE...The <i>Buttons</i> style is not generated
color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.

destroy type	AUTOMATIC...neither the <i>Temporary</i> nor Permanent styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE... <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. In the "Available Events" list, there are several items: cmd-help, cmd-tabchanged, msg-end-menu, msg-init-menu, msg-menu-input, and msg-validate. Between the two sections are four buttons: a single arrow pointing right (>), a double arrow pointing right (">>>), a single arrow pointing left (<), and a double arrow pointing left (<<). At the bottom of the dialog are two buttons: "OK" (highlighted in blue) and "Cancel".</p>	
exclude event list	NONE... The <i>Exclude-Event-List</i> property is not generated. 0... <i>Exclude-Event-List=0</i> is generated. 1... <i>Exclude-Event-List=1</i> is generated.
fixed-width	TRUE...The <i>Fixed-Width</i> style is generated FALSE...The <i>Fixed-Width</i> style is not generated
flat-buttons	TRUE...The <i>Flat-Buttons</i> style is generated FALSE...The <i>Flat-Buttons</i> style is not generated

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a list of fonts on the left: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. On the right, there's a vertical font size dropdown with options 8, 9, 10, 11, 12, and 13, where '12' is selected. Below the font list are sections for 'Effects' (Bold, Italic, Underline, Strikethrough) and 'Preview' (a text area containing 'Preview Font'). At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It includes a 'Set RGB colors' checkbox, which is unchecked. Below it is a 'Foreground' color picker set to '12 Red'. A preview window shows a red rectangle. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
hot-track	TRUE...The <i>Hot-Track</i> style is generated FALSE...The <i>Hot-Track</i> style is not generated

id	Specifies the control id. This property is set automatically when the control is drawn.
key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules.
	
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height expressed in pixels. This property is set automatically when the control is drawn
lines unit	<p>DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting</p> <p>None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value</p> <p><i>CELLS</i>... <i>CELLS</i> is generated after the <i>Lines</i> value</p> <p><i>PIXELS</i>... <i>PIXELS</i> is generated after the <i>Lines</i> value</p>
lock	<p>TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse.</p> <p>FALSE...You can move the control on the Screen Designer by dragging it with the mouse</p>
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells

min-height	Specifies the control minimum height as expressed in cells
min-width	Specifies the control minimum width as expressed in cells
multiline	TRUE...The <i>Multiline</i> style is generated FALSE...The <i>Multiline</i> style is not generated
no-dividers	TRUE...The <i>No-Dividers</i> style is generated FALSE...The <i>No-Dividers</i> style is not generated
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
relative-offset	TRUE...The <i>Relative-Offset</i> style is generated FALSE...The <i>Relative-Offset</i> style is not generated
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
tab orientation	Allows the user to choose one of the following styles: TOP VERTICAL LEFT VERTICAL RIGHT BOTTOM
text-norotate	TRUE... The <i>Text-Norotate</i> style is generated FALSE.. The <i>Text-Norotate</i> style is not generated
value	Specifies the value for the <i>Value</i> property
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
cmd-tabchanged event	Allows the user to create a paragraph to handle the CMD-TABCHANGED event in the Procedure Division

msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-tabchanged	Allows the user to create a paragraph to handle the CMD-TABCHANGED event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
After-Tabchg Display	Allows the user to create a paragraph that will be executed after CMD-TABCHANGED handling
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Before-Tabchg Display	Allows the user to create a paragraph that will be executed before CMD-TABCHANGED handling
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Variables	
bitmap-width variable	Numeric variable that hosts the value for the <i>Bitmap-Width</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state

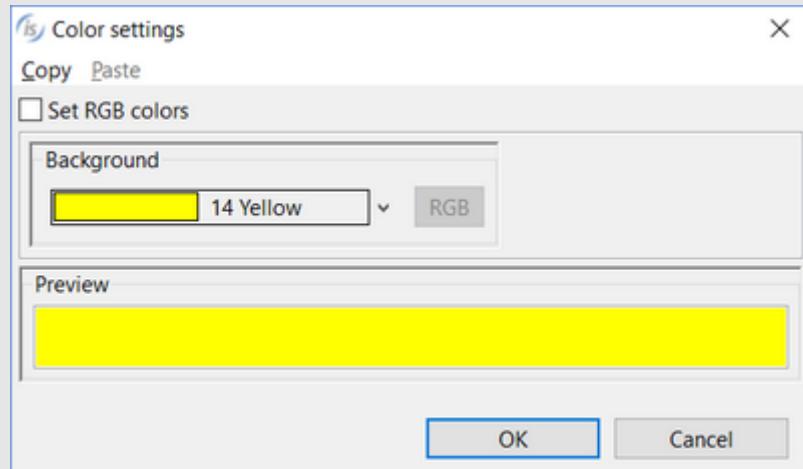
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the value for the <i>Hint</i> property
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
value variable	Numeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

TREE VIEW

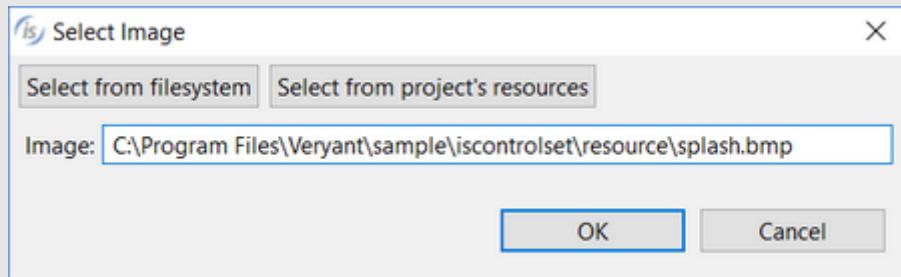
Refer to [TREE-VIEW](#) for details about properties, styles and events of this control.

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
action	Specifies the value for the <i>Action</i> property
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.

background-color Opens a dialog that allows the user to choose the control background color.



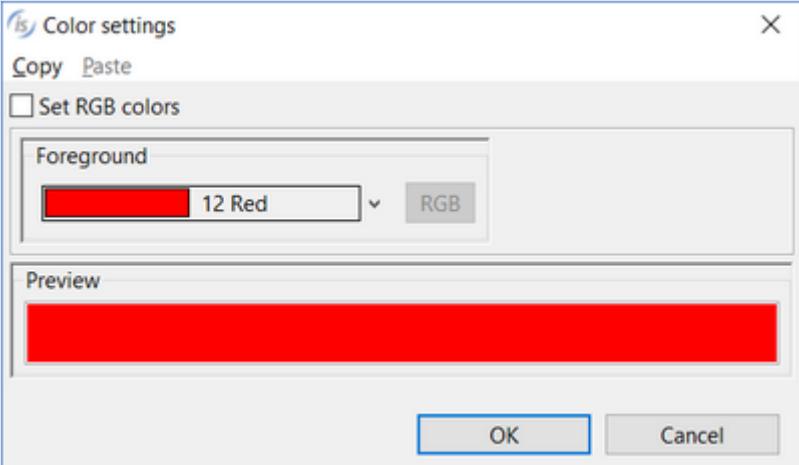
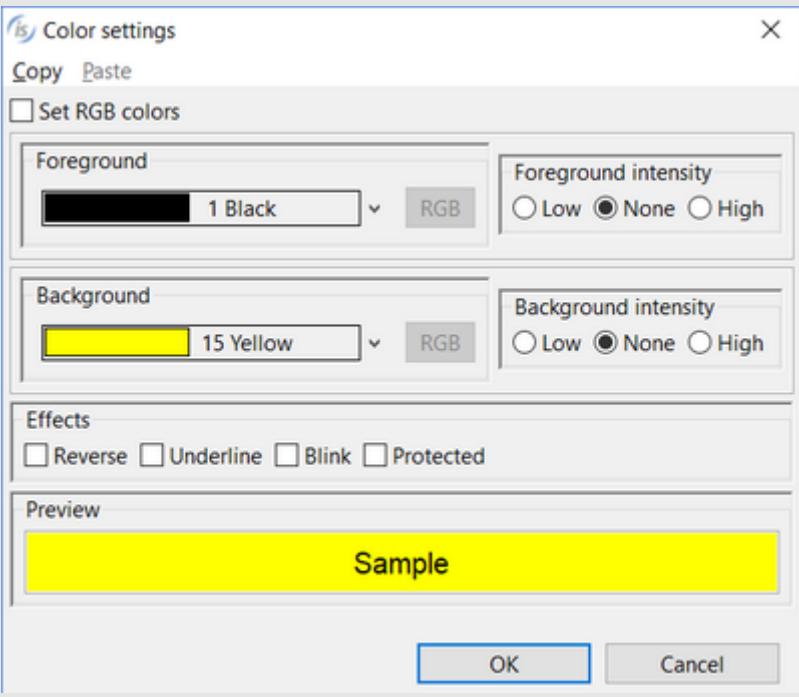
bitmap Opens a dialog box that allows the user to select an image file to load into the control

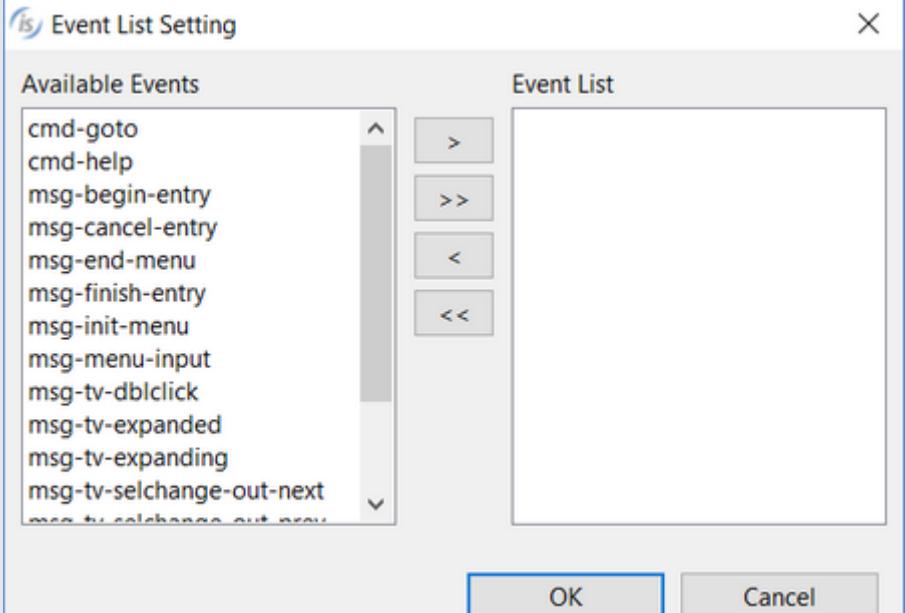


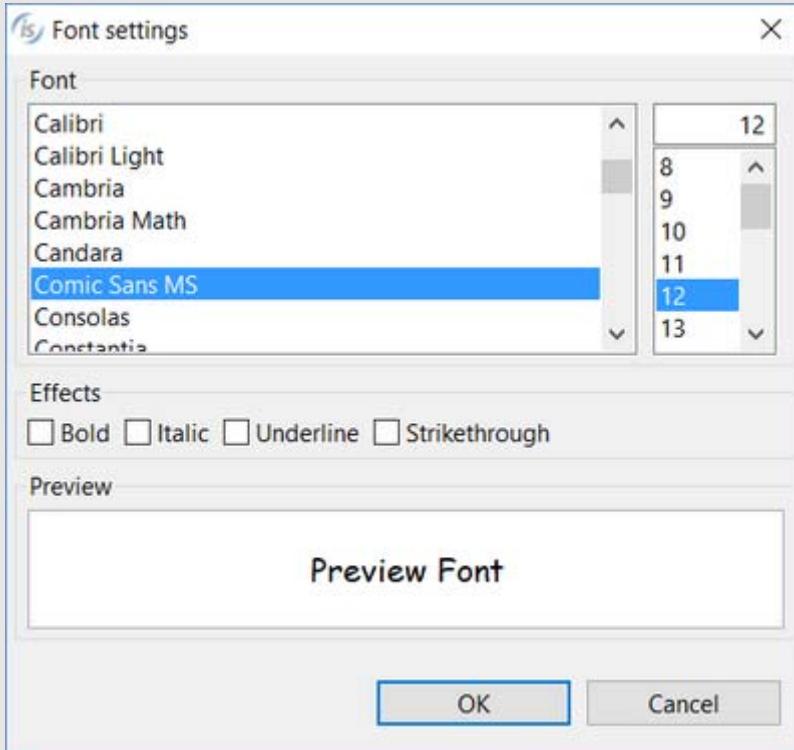
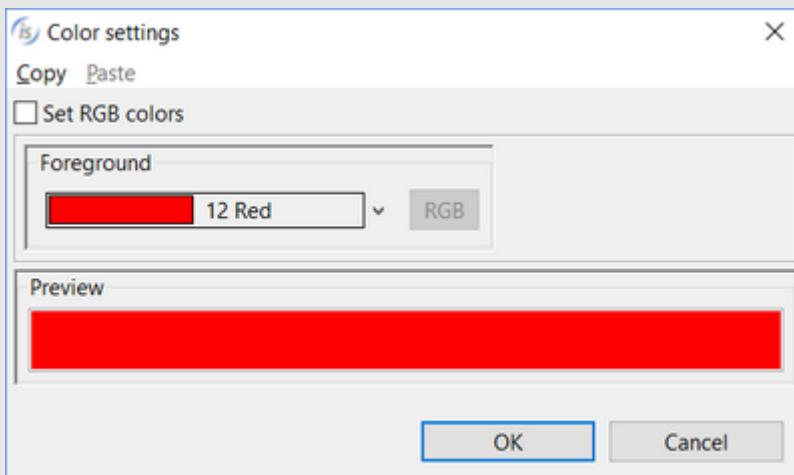
bitmap-width Specifies the value for the *Bitmap-Width* property

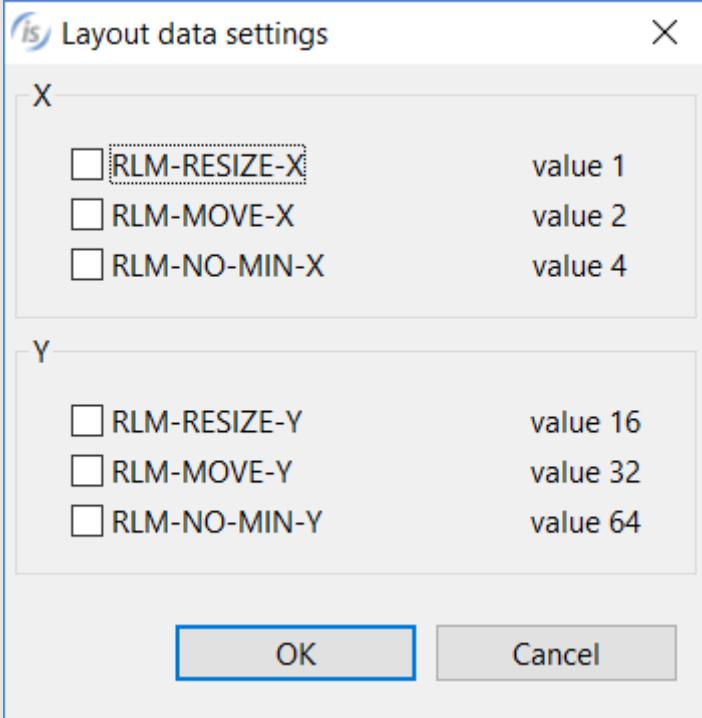
buttons TRUE...The *Buttons* style is generated
FALSE...The *Buttons* style is not generated

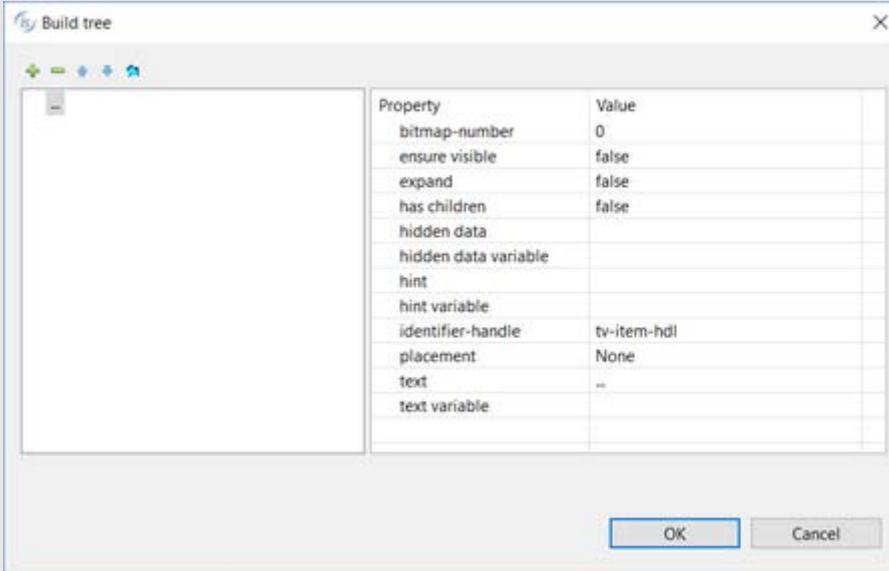
border Allows the user to choose one of the following styles:
3-D
BOXED
NO-BOX

border-color	Opens a dialog that allows the user to choose the control border color.
	
color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.

destroy type	AUTOMATIC...neither the <i>Temporary</i> nor Permanent styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE...The <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. The "Available Events" section contains a list of event names: cmd-goto, cmd-help, msg-begin-entry, msg-cancel-entry, msg-end-menu, msg-finish-entry, msg-init-menu, msg-menu-input, msg-tv-dblclick, msg-tv-expanded, msg-tv-expanding, and msg-tv-selchange-out-next. To the right of this list are four buttons: a top-right arrow (>), a double-right arrow (>>), a left arrow (<), and a double-left arrow (<<). Below the "Event List" section are two buttons: "OK" and "Cancel".</p>
exclude event list	NONE... The <i>Exclude-Event-List</i> property is not generated. 0... <i>Exclude-Event-List=0</i> is generated. 1... <i>Exclude-Event-List=1</i> is generated.

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a title bar with a close button ('X'). Below it is a 'Font' section containing a list of font names: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical scroll bar and a size dropdown menu showing options 8, 9, 10, 11, 12 (selected), and 13. Below the font list is an 'Effects' section with checkboxes for Bold, Italic, Underline, and Strikethrough. Underneath that is a 'Preview' section with a text area containing the text 'Preview Font'. At the bottom are 'OK' and 'Cancel' buttons.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a title bar with a close button ('X'). Below it is a 'Copy' and 'Paste' button, followed by a checkbox labeled 'Set RGB colors'. A 'Foreground' section contains a color swatch set to '12 Red' and an 'RGB' button. Below that is a 'Preview' section with a red rectangular preview area. At the bottom are 'OK' and 'Cancel' buttons.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
id	Specifies the control id. This property is set automatically when the control is drawn.

key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules. 
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
mass-update	Specifies the value for the <i>Mass-Update</i> property
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells

min-height	Specifies the control minimum height as expressed in cells																										
min-width	Specifies the control minimum width as expressed in cells																										
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated																										
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.																										
show lines	TRUE...The <i>Show-Lines</i> style is generated FALSE...The <i>Show-Lines</i> style is not generated																										
show selection always	TRUE...The <i>Show-Sel-Always</i> style is generated FALSE...The <i>Show-Sel-Always</i> style is not generated																										
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn																										
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn																										
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value																										
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn																										
tree item settings	Opens a dialog that allows the user to define items																										
 <table border="1"> <thead> <tr> <th>Property</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>bitmap-number</td> <td>0</td> </tr> <tr> <td>ensure visible</td> <td>false</td> </tr> <tr> <td>expand</td> <td>false</td> </tr> <tr> <td>has children</td> <td>false</td> </tr> <tr> <td>hidden data</td> <td></td> </tr> <tr> <td>hidden data variable</td> <td></td> </tr> <tr> <td>hint</td> <td></td> </tr> <tr> <td>hint variable</td> <td></td> </tr> <tr> <td>identifier-handle</td> <td>tv-item-hdl</td> </tr> <tr> <td>placement</td> <td>None</td> </tr> <tr> <td>text</td> <td>-</td> </tr> <tr> <td>text variable</td> <td></td> </tr> </tbody> </table>		Property	Value	bitmap-number	0	ensure visible	false	expand	false	has children	false	hidden data		hidden data variable		hint		hint variable		identifier-handle	tv-item-hdl	placement	None	text	-	text variable	
Property	Value																										
bitmap-number	0																										
ensure visible	false																										
expand	false																										
has children	false																										
hidden data																											
hidden data variable																											
hint																											
hint variable																											
identifier-handle	tv-item-hdl																										
placement	None																										
text	-																										
text variable																											
value	Specifies the value for the <i>Value</i> property																										
visible	NONE... <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated																										

width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-begin-entry event	Allows the user to create a paragraph to handle the MSG-BEGIN-ENTRY event in the Procedure Division
msg-cancel-entry event	Allows the user to create a paragraph to handle the MSG-CANCEL-ENTRY event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-finish-entry	Allows the user to create a paragraph to handle the MSG-FINISH-ENTRY event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
msg-tv-dblclick event	Allows the user to create a paragraph to handle the MSG-TV-DBLCLICK event in the Procedure Division
msg-tv-expanded event	Allows the user to create a paragraph to handle the MSG-TV-EXPANDED event in the Procedure Division
msg-tv-expanding event	Allows the user to create a paragraph to handle the MSG-TV-EXPANDING event in the Procedure Division
msg-tv-out-next event	Allows the user to create a paragraph to handle the MSG-TV-OUT-NEXT event in the Procedure Division
msg-tv-out-prev event	Allows the user to create a paragraph to handle the MSG-TV-OUT-PREV event in the Procedure Division
msg-tv-selchanged event	Allows the user to create a paragraph to handle the MSG-TV-SELCHANGED event in the Procedure Division
msg-tv-selchanging event	Allows the user to create a paragraph to handle the MSG-TV-SELCHANGING event in the Procedure Division
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above

cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPTION PROCEDURE
Variables	
bitmap-width variable	Numeric variable that hosts the value for the <i>Bitmap-Width</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.
enabled variable	Numeric variable that hosts the enabled state
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the hint value.
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
mass-update	Numeric variable that hosts the value for the <i>Mass-Update</i> property
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
value variable	Numeric variable that hosts the value for the <i>Value</i> property

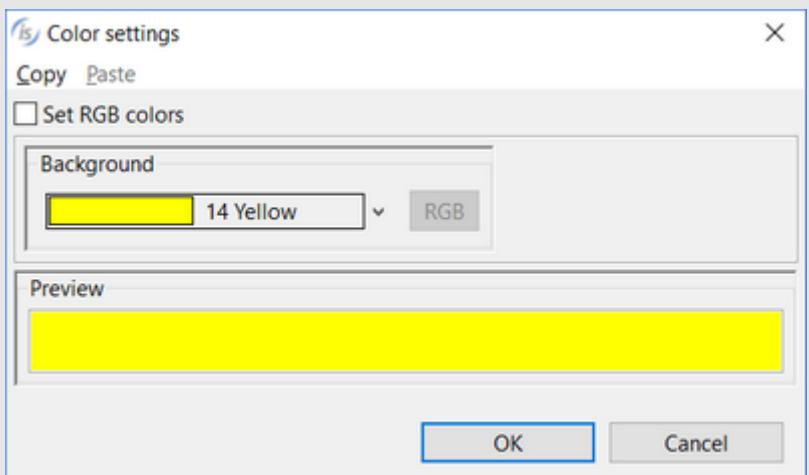
visible variable	Numeric variable that hosts the visible state
------------------	---

WEB BROWSER

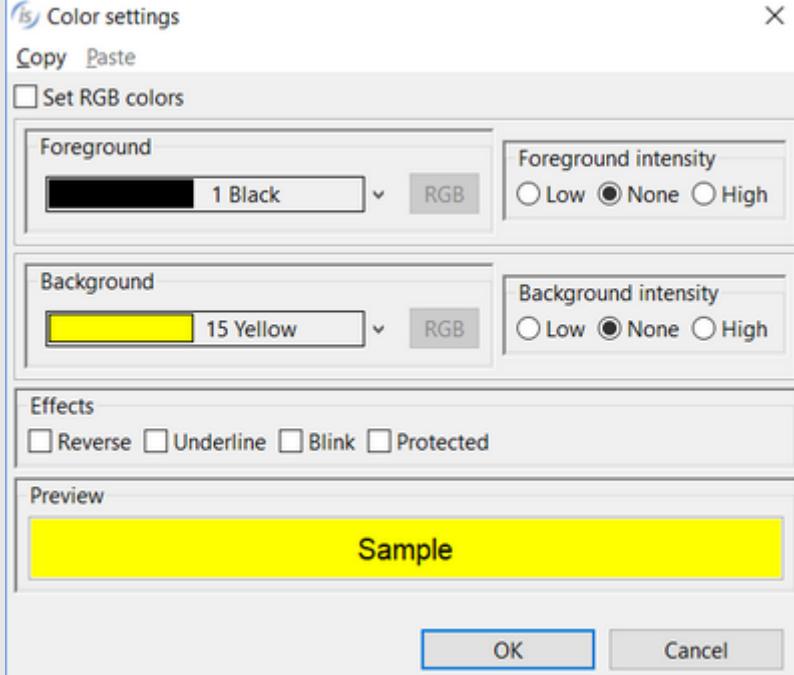
Refer to [WEB-BROWSER](#) for details about properties, styles and events of this control.

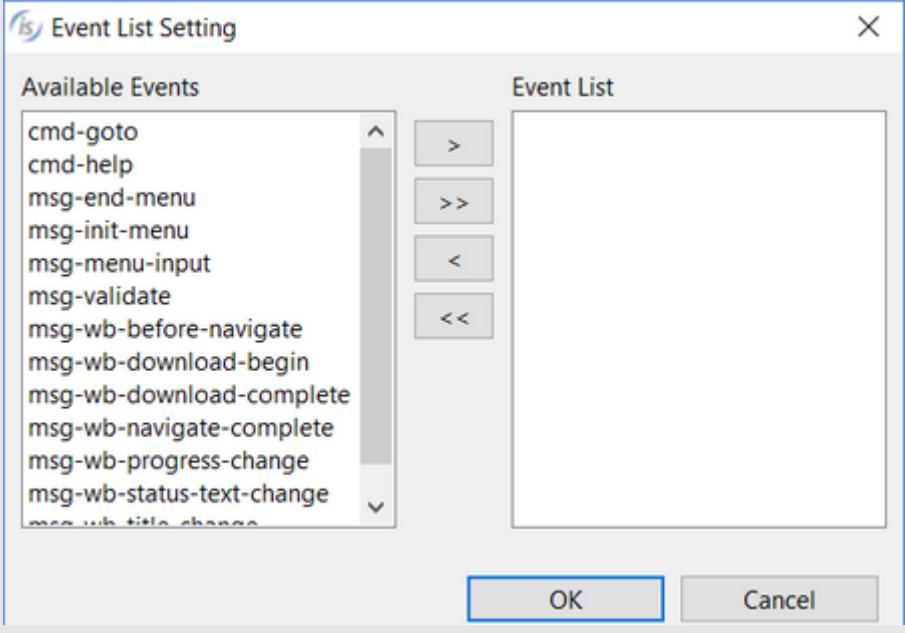
Properties

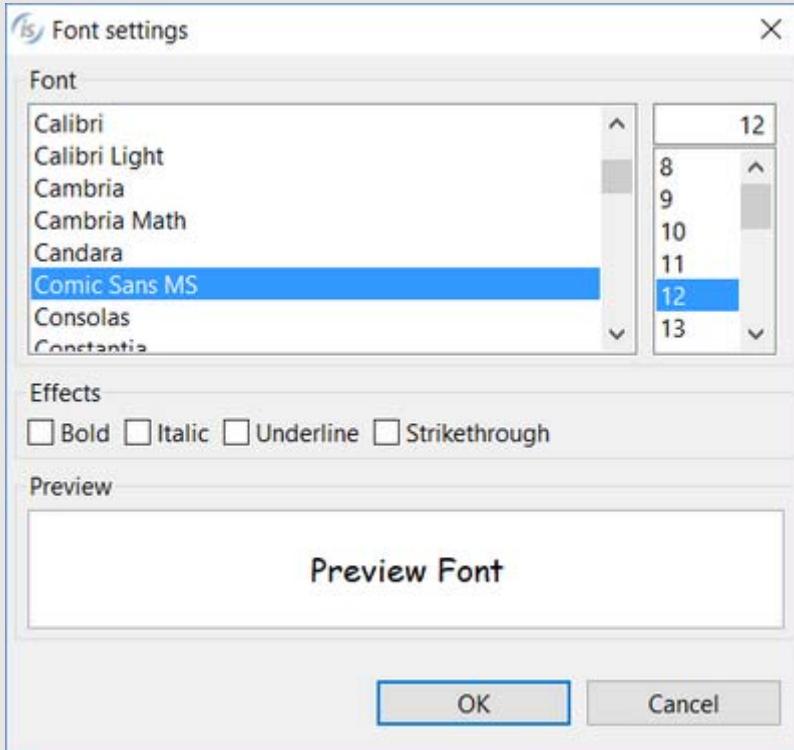
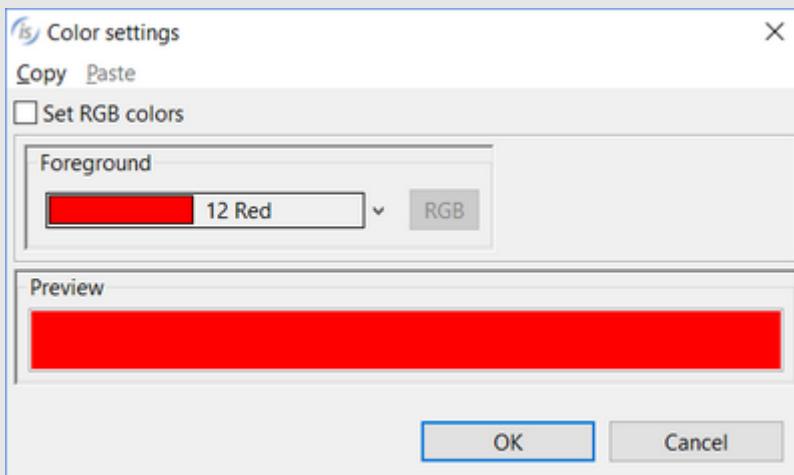
(name)	Specifies the control name. This property is set automatically when the control is drawn
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
background-color	Opens a dialog that allows the user to choose the control background color.

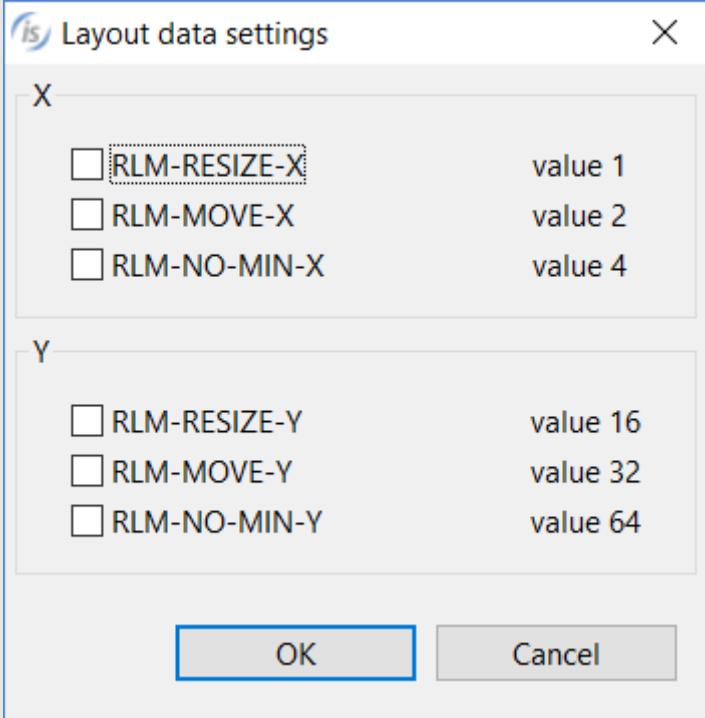


The image shows a 'Color settings' dialog box. At the top, there are 'Copy' and 'Paste' buttons and a checkbox for 'Set RGB colors'. Below this is a 'Background' section containing a color swatch labeled '14 Yellow' and an 'RGB' button. A 'Preview' section shows a horizontal bar filled with yellow. At the bottom right are 'OK' and 'Cancel' buttons.

color	Opens a dialog that allows the user to choose the control color.
	
column	Specifies the X coordinate of the control as expressed in cells. This property is set automatically when the control is drawn.
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn.
css-base-style-name css-style-name	Specify the CSS style associated with the control. It works only in a Web Direct 2 environment. See Customize the EIS WD2 Layout using CSS for more information.
custom-data	Specifies the value for the <i>Custom-Data</i> property.
destroy type	AUTOMATIC...neither the <i>Temporary</i> nor <i>Permanent</i> styles are generated TEMPORARY... <i>Temporary</i> style is generated PERMANENT... <i>Permanent</i> style is generated
enabled	NONE...The <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated

event list	Opens a dialog that allows to choose which events must be added to the event list of this control.
	 <p>The dialog box is titled "Event List Setting". It has two main sections: "Available Events" on the left and "Event List" on the right. The "Available Events" section contains a list of event names: cmd-goto, cmd-help, msg-end-menu, msg-init-menu, msg-menu-input, msg-validate, msg-wb-before-navigate, msg-wb-download-begin, msg-wb-download-complete, msg-wb-navigate-complete, msg-wb-progress-change, msg-wb-status-text-change, and msg-wb-title-change. To the right of this list are four buttons: >, >>, <, and <<. Below the "Event List" section are "OK" and "Cancel" buttons.</p>

font	Opens a dialog that allows the user to choose the control font.
	 <p>The screenshot shows the 'Font settings' dialog box. It has a title bar 'Font settings' with a close button 'X'. Below it is a 'Font' section containing a list of font names: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, and Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical scroll bar and a size dropdown menu showing values 8, 9, 10, 11, 12 (highlighted), and 13. Below the font list is an 'Effects' section with checkboxes for Bold, Italic, Underline, and Strikethrough, all of which are unchecked. Underneath that is a 'Preview' section with a text area containing the placeholder 'Preview Font'. At the bottom are 'OK' and 'Cancel' buttons, with 'OK' being highlighted.</p>
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a title bar 'Color settings' with a close button 'X'. Below it is a 'Copy' and 'Paste' button, followed by a checkbox 'Set RGB colors' which is unchecked. A 'Foreground' section contains a color swatch labeled '12 Red' and an 'RGB' button. Below that is a 'Preview' section with a red rectangular preview area. At the bottom are 'OK' and 'Cancel' buttons, with 'OK' being highlighted.</p>
height-in-cells	TRUE...The <i>Height-In-Cells</i> style is generated FALSE...The <i>Height-In-Cells</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
hint	Specifies the value for the <i>Hint</i> property.
id	Specifies the control id. This property is set automatically when the control is drawn.

key	Specifies the value for the <i>Key</i> property.
layout-data	Opens a dialog that allows the user to choose the control resize rules. 
line	Specifies the Y coordinate of the control as expressed in cells. This property is set automatically when the control is drawn
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Lines</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Lines</i> value
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
max-height	Specifies the control maximum height as expressed in cells
max-width	Specifies the control maximum width as expressed in cells
min-height	Specifies the control minimum height as expressed in cells

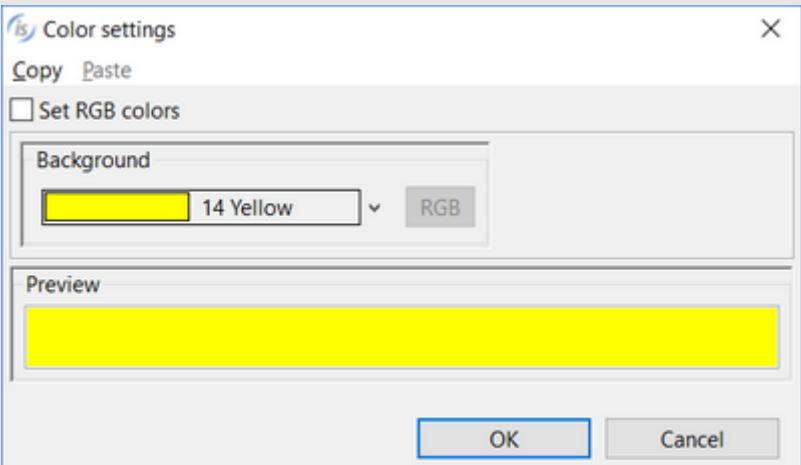
min-width	Specifies the control minimum width as expressed in cells
no-tab	TRUE...The <i>No-Tab</i> style is generated FALSE...The <i>No-Tab</i> style is not generated
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
size	Specifies the control width as expressed in cells. This property is set automatically when the control is drawn
size pixels	Specifies the control width as expressed in pixels. This property is set automatically when the control is drawn
size unit	DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Size</i> value depending on the window's "cell" property setting None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Size</i> value <i>CELLS</i> ... <i>CELLS</i> is generated after the <i>Size</i> value <i>PIXELS</i> ... <i>PIXELS</i> is generated after the <i>Size</i> value
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
use ALT	TRUE...The <i>Use-Alt</i> style is generated FALSE...The <i>Use-Alt</i> style is not generated
use RETURN	TRUE...The <i>Use-Return</i> style is generated FALSE...The <i>Use-Return</i> style is not generated
use TAB	TRUE...The <i>Use-Tab</i> style is generated FALSE...The <i>Use-Tab</i> style is not generated
value	Specifies the value for the <i>Value</i> property
visible	NONE...The <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-goto event	Allows the user to create a paragraph to handle the CMD-GOTO event in the Procedure Division
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division

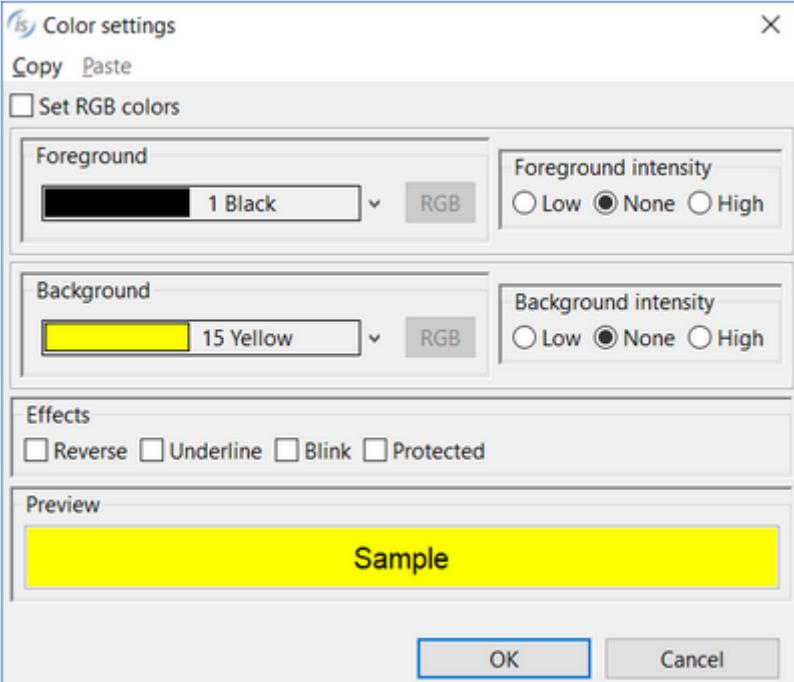
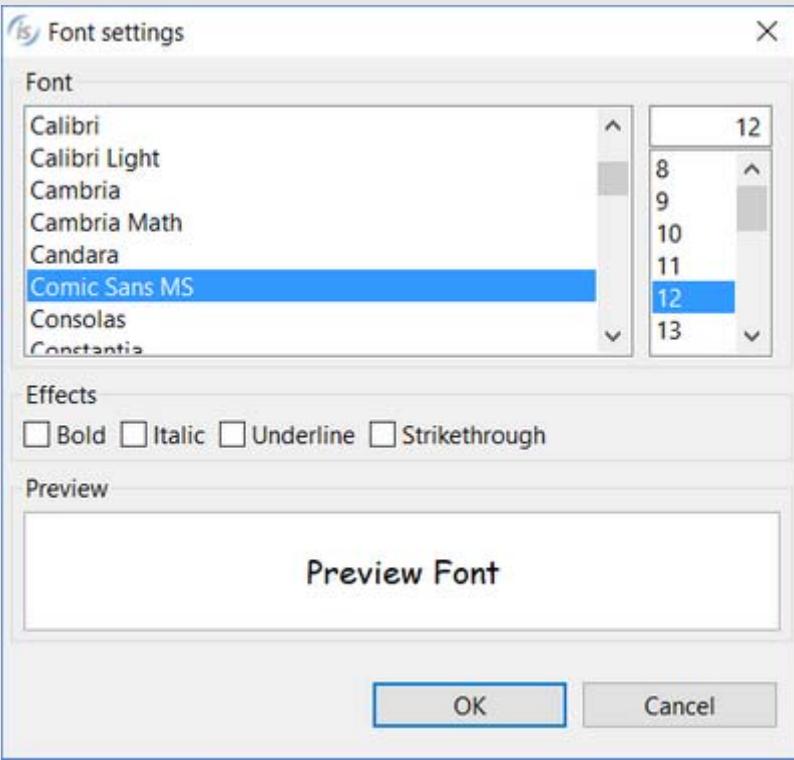
msg-validate event	Allows the user to create a paragraph to handle the MSG-VALIDATE event in the Procedure Division
msg-wb-before-navigate event	Allows the user to create a paragraph to handle the MSG-WB-BEFORE-NAVIGATE event in the Procedure Division
msg-wb-download-begin event	Allows the user to create a paragraph to handle the MSG-WB-DOWNLOAD-BEGIN event in the Procedure Division
msg-wb-download-complete event	Allows the user to create a paragraph to handle the MSG-WB-DOWNLOAD-COMPLETE event in the Procedure Division
msg-wb-navigate-complete event	Allows the user to create a paragraph to handle the MSG-WB-NAVIGATE-COMPLETE event in the Procedure Division
msg-wb-progress-change event	Allows the user to create a paragraph to handle the MSG-WB-PROGRESS-CHANGE event in the Procedure Division
msg-wb-status-text-change event	Allows the user to create a paragraph to handle the MSG-WB-STATUS-TEXT-CHANGE event in the Procedure Division
msg-wb-title-change event	Allows the user to create a paragraph to handle the MSG-WB-TITLE-CHANGE event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-goto exception	Allows the user to create a paragraph to handle the CMD-GOTO event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
After procedure	Allows the user to create a paragraph to handle the control AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
Before procedure	Allows the user to create a paragraph to handle the control BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCETPION PROCEDURE
Variables	
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the column value
css-style-name variable	Alphanumeric variable that hosts the css style associated with the control. It works only in a Web Direct 2 environment.

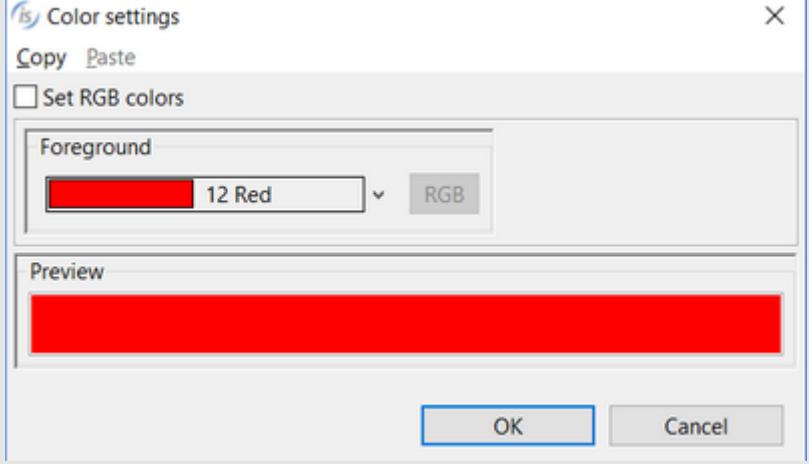
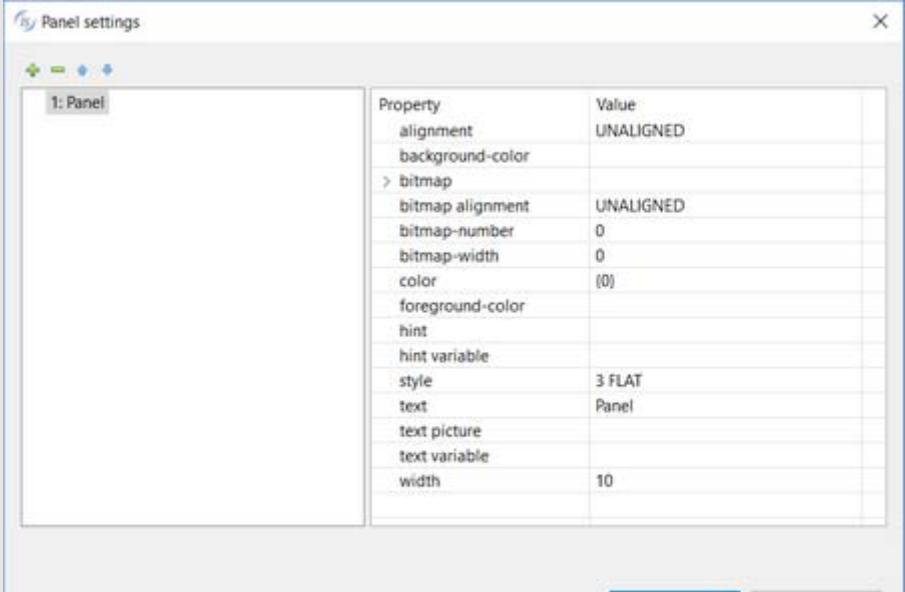
enabled variable	Numeric variable that hosts the enabled state
help-id variable	Numeric variable that hosts the help id
hint variable	Alphanumeric variable that hosts the hint value.
id variable	Numeric variable that hosts the control id
key variable	Alphanumeric variable that hosts the value for the <i>Key</i> property
layout-data variable	Numeric variable that hosts the control resize rules
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the line value
max-height variable	Numeric variable that hosts the maximum height
max-width variable	Numeric variable that hosts the maximum width
min-height variable	Numeric variable that hosts the minimum height
min-width variable	Numeric variable that hosts the minimum width
size variable	Numeric variable that hosts the size value
value variable	Numeric variable that hosts the value for the <i>Value</i> property
visible variable	Numeric variable that hosts the visible state

STATUS BAR

Refer to [STATUS-BAR](#) for details about properties, styles and events of this control.

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
background-color	Opens a dialog that allows the user to choose the control background color.
	

color	Opens a dialog that allows the user to choose the control color.
	
custom-data	Specifies the value for the <i>Custom-Data</i> property.
font	Opens a dialog that allows the user to choose the control font.
	

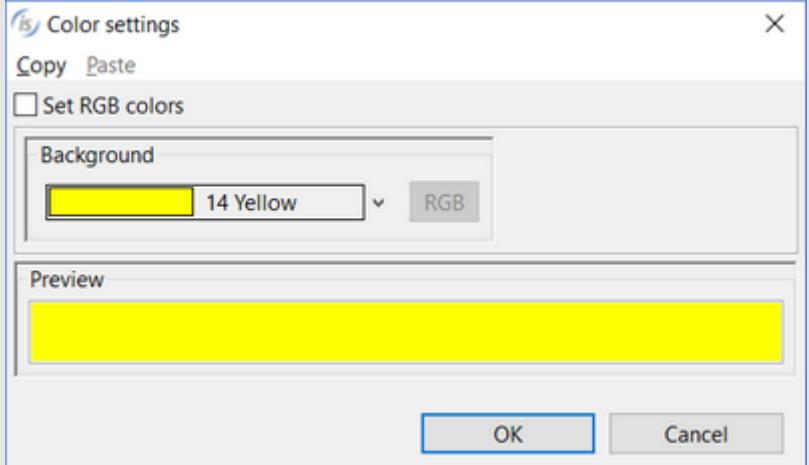
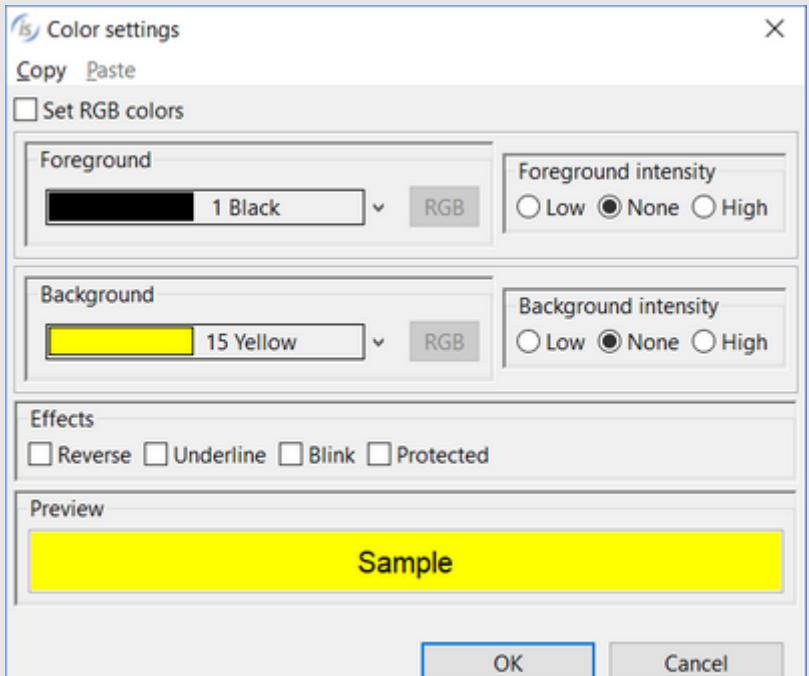
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
	
grip	TRUE... The <i>Grip</i> style is generated FALSE... The <i>Grip</i> style is not generated
help-id	Specifies the control <i>Help-id</i> .
id	Specifies the control id. This property is set automatically when the control is drawn.
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
panel settings	Opens a dialog that allows the user to define panels
	
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.

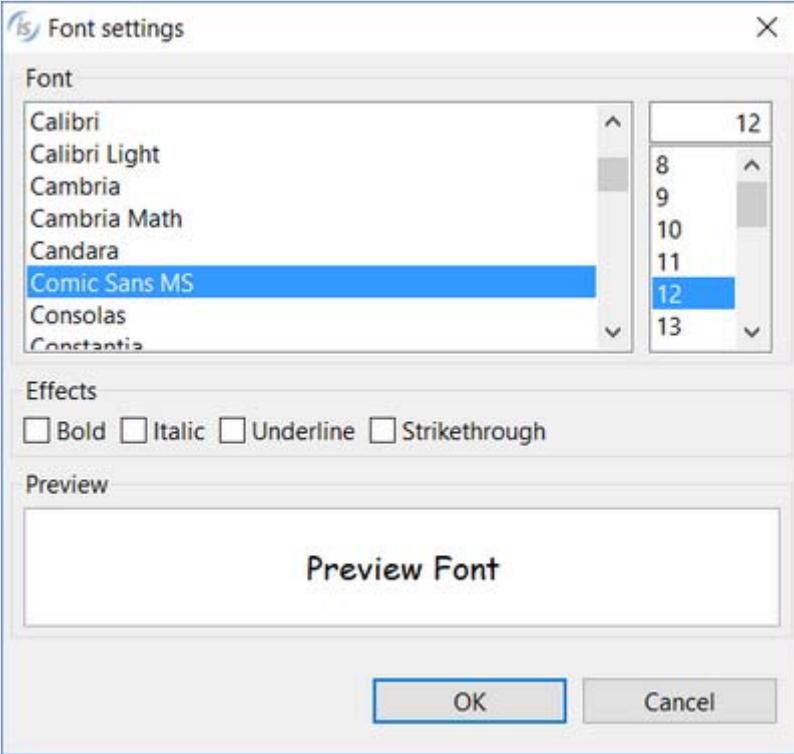
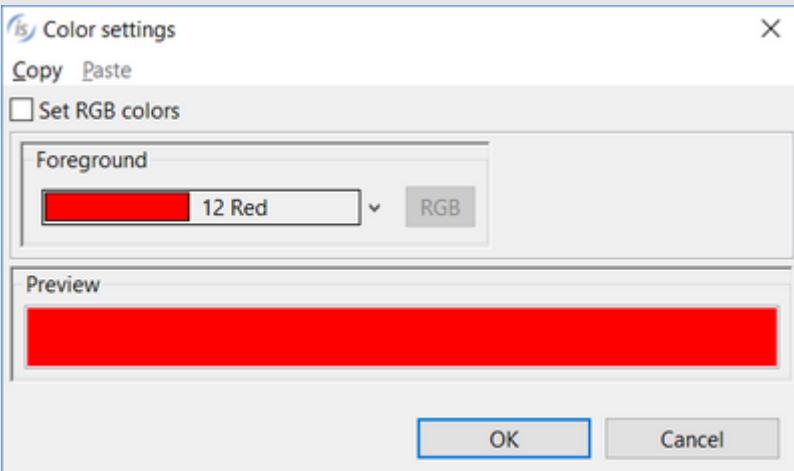
visible	NONE...The <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE...The <i>Width-In-Cells</i> style is generated FALSE... The <i>Width-In-Cells</i> style is not generated
Events	
cmd-help event	Allows the user to create a paragraph to handle the CMD-HELP event in the Procedure Division
msg-st-dblclick event	Allows the user to create a paragraph to handle the MSG-ST-DBLCLICK event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-help exception	Allows the user to create a paragraph to handle the CMD-HELP event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	
Event procedure	Allows the user to create a paragraph to handle the control EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the control EXCEPTION PROCEDURE
Variables	
background-color variable	Numeric variable that hosts the background-color value
color variable	Numeric variable that hosts the color value
foreground-color variable	Numeric variable that hosts the foreground-color value
help-id variable	Numeric variable that hosts the control help id
id variable	Numeric variable that hosts the control id
status bar handle	Numeric variable that hosts the control handle

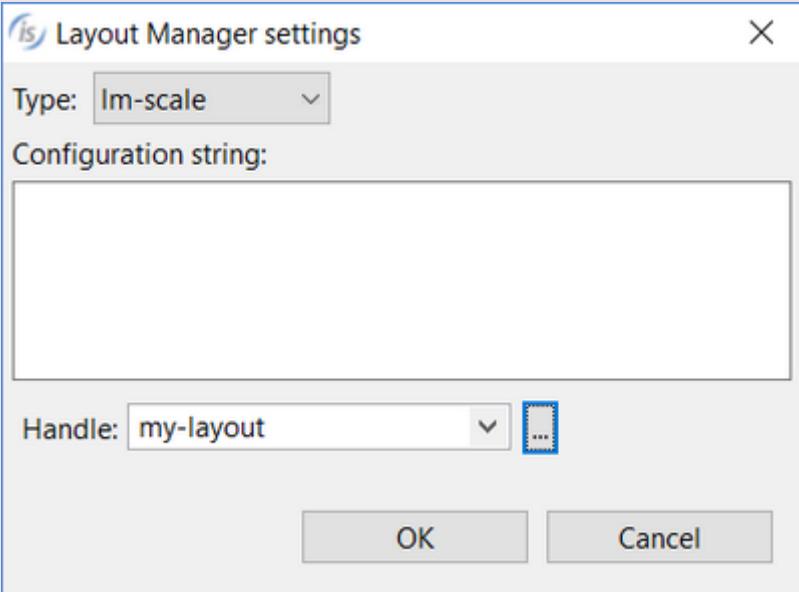
TOOL BAR

Refer to [TOOL-BAR](#) for details about properties, styles and events of this control.

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn
cell	TRUE... The <i>Cell-Width</i> and <i>Cell-Height</i> properties are generated and <i>Lines</i> is measured in cells. FALSE... The <i>Cell-Width</i> and <i>Cell-Height</i> properties are not generated and <i>Lines</i> is not measured in cells.
cell-height	Specifies the value for the <i>Cell-Height</i> property

call-width	Specifies the value for the <i>Cell-Width</i> property
background-color	Opens a dialog that allows the user to choose the control background color.
color	 <p>The screenshot shows the 'Color settings' dialog box. It has a 'Background' section where the color is set to '14 Yellow'. A preview window below shows a yellow rectangle. At the bottom are 'OK' and 'Cancel' buttons.</p>  <p>The screenshot shows the 'Color settings' dialog box. It has sections for 'Foreground' (color set to '1 Black') and 'Background' (color set to '15 Yellow'). There are also 'Foreground intensity' and 'Background intensity' options, both with 'None' selected. Under 'Effects', 'Reverse' is checked. A preview window shows the text 'Sample' in black on a yellow background. At the bottom are 'OK' and 'Cancel' buttons.</p>

control font	Opens a dialog that allows the user to choose the control font.
 A screenshot of the 'Font settings' dialog box. It has a title bar 'Font settings' with a close button 'X'. The main area is titled 'Font' and contains a list of font names: Calibri, Calibri Light, Cambria, Cambria Math, Candara, Comic Sans MS, Consolas, Constantia. 'Comic Sans MS' is highlighted with a blue selection bar. To the right of the font list is a vertical scroll bar and a size selector with values 8, 9, 10, 11, 12, 13, with '12' currently selected. Below the font list is an 'Effects' section with checkboxes for Bold, Italic, Underline, and Strikethrough, all of which are unchecked. A 'Preview' section shows the text 'Preview Font' in the selected font and size. At the bottom are 'OK' and 'Cancel' buttons, with 'OK' being the primary button. <p>Font settings</p> <p>Font</p> <p>Effects</p> <p>Preview</p> <p>Preview Font</p> <p>OK Cancel</p>	
custom-data	Specifies the value for the <i>Custom-Data</i> property.
foreground-color	Opens a dialog that allows the user to choose the control foreground color.
 A screenshot of the 'Color settings' dialog box. It has a title bar 'Color settings' with a close button 'X'. The main area includes 'Copy' and 'Paste' buttons and a checkbox 'Set RGB colors' which is unchecked. Below this is a 'Foreground' color picker showing a red square with the text '12 Red' and an 'RGB' button. A 'Preview' section shows a horizontal bar filled with red. At the bottom are 'OK' and 'Cancel' buttons, with 'OK' being the primary button. <p>Color settings</p> <p>Copy Paste</p> <p>Set RGB colors</p> <p>Foreground</p> <p>12 Red RGB</p> <p>Preview</p> <p>OK Cancel</p>	

layout manager	Opens a dialog that allows to choose which layout manager should be associated to the toolbar. When either LM-SCALE or LM-RESPONSIVE is selected, it's possible to specify the configuration string. In this dialog you also associate a handle to the layout manager.
	
lines	Specifies the control height as expressed in cells. This property is set automatically when the control is drawn
lines pixels	Specifies the control height as expressed in pixels. This property is set automatically when the control is drawn
lines unit	<p>DEFAULT... Either <i>CELLS</i> or nothing is generated after the <i>Lines</i> value depending on the window's "cell" property setting</p> <p>None... Neither <i>CELLS</i> nor <i>PIXELS</i> are generated after the <i>Lines</i> value</p> <p><i>CELLS</i>... <i>CELLS</i> is generated after the <i>Lines</i> value</p> <p><i>PIXELS</i>... <i>PIXELS</i> is generated after the <i>Lines</i> value</p>
lock	<p>TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse.</p> <p>FALSE...You can move the control on the Screen Designer by dragging it with the mouse</p>
moveable	<p>TRUE... The <i>Moveable</i> style is generated</p> <p>FALSE... The <i>Moveable</i> style is not generated</p>
multiline	<p>TRUE... The <i>Multiline</i> style is generated</p> <p>FALSE... The <i>Multiline</i> style is not generated</p>
pop up menu	Associates a pop-up menu with the control. The menu must have been drawn on the same screen.
tab order	Sets the ordinal position of the control in the Screen Section. This property is set automatically when the control is drawn
width-in-cells	<p>TRUE...The <i>Width-In-Cells</i> style is generated</p> <p>FALSE... The <i>Width-In-Cells</i> style is not generated</p>

Events

No Events available.

Exceptions

No Exceptions available.

Procedures

No Procedures available.

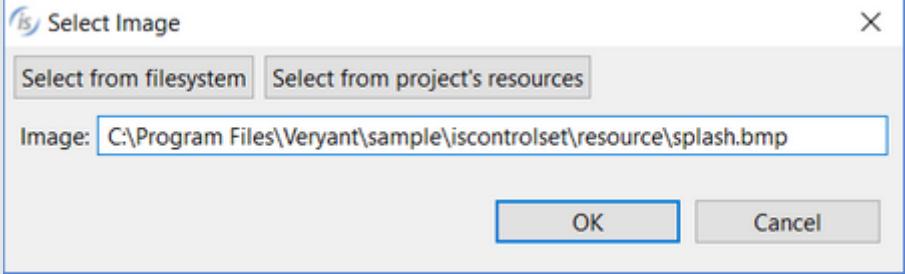
Variables

toolbar-handle	Numeric variable that hosts the control handle
----------------	--

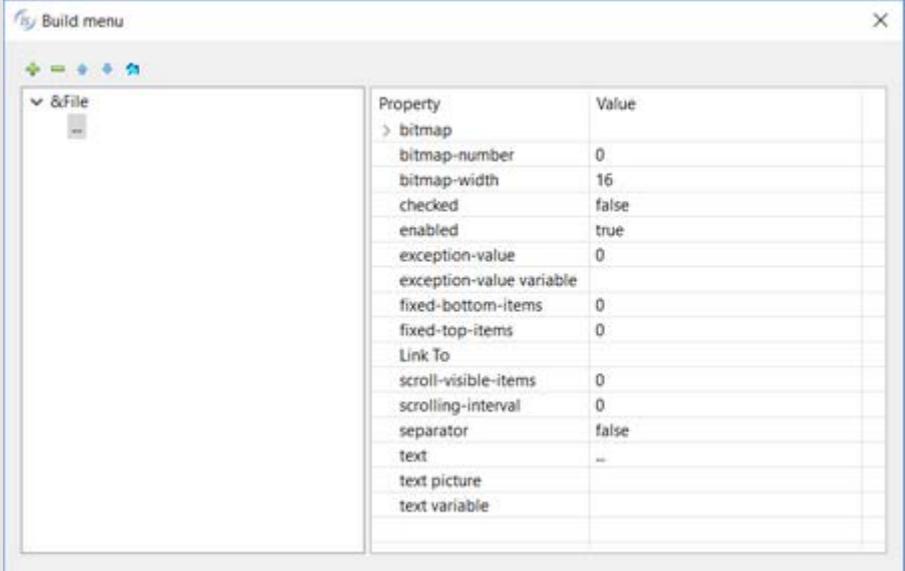
MENU

Properties

(name)	Specifies the control name. This property is set automatically when the control is drawn
bitmap	Opens a dialog box that allows the user to select an image file to be used as tray icon. It's considered only when style=SYSTEM-TRAY



bitmap-number	Specifies which bitmap of a bitmap strip must be used as tray icon. It's considered only when style=SYSTEM-TRAY
bitmap-width	Specifies the width in pixels of the tray icon. It's considered only when style=SYSTEM-TRAY
column pixels	Specifies the X coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
exception-value	Specifies the exception-value returned when clicking on the tray icon. It's considered only when style=SYSTEM-TRAY
exception-value-2	Specifies the exception-value returned when double-clicking on the tray icon. It's considered only when style=SYSTEM-TRAY
item settings	Opens a dialog that allows the user to define menu items

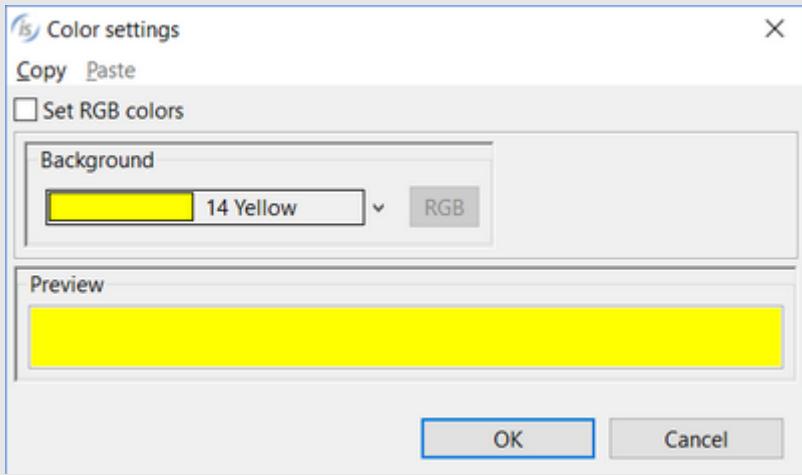
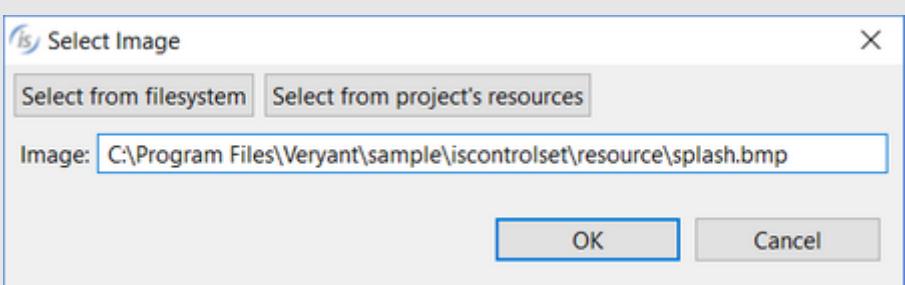


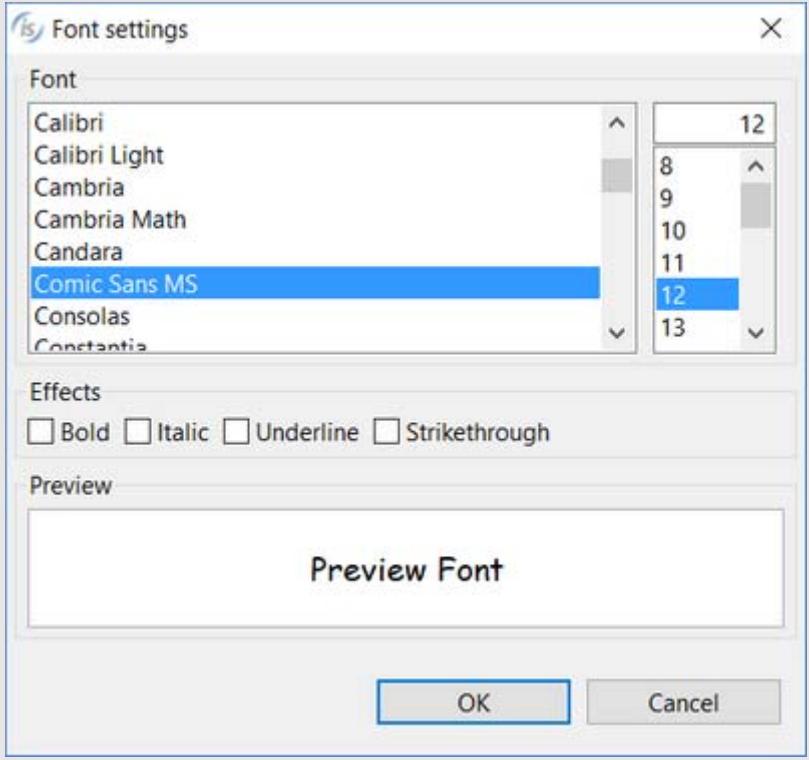
line pixels	Specifies the Y coordinate of the control as expressed in pixels. This property is set automatically when the control is drawn
style	Allows the user to choose the type of menu MENU-BAR POP-UP SYSTEM-TRAY
tooltip-text	Specifies the tray icon tool tip text. It's considered only when style=SYSTEM-TRAY
tooltip-text picture	Specifies the picture for the tooltip-text variable
Events	
No Events available.	
Exceptions	
No Exceptions available.	
Procedures	
No Procedures available.	
Variables	
exception-value	Numeric variable that hosts the exception-value generated when clicking on the tray icon
exception-value-2	Numeric variable that hosts the exception-value generated when double-clicking on the tray icon
menu-handle	Numeric variable that hosts the control handle
tooltip-text variable	Alphanumeric variable that hosts the tool-tip for the tray icon

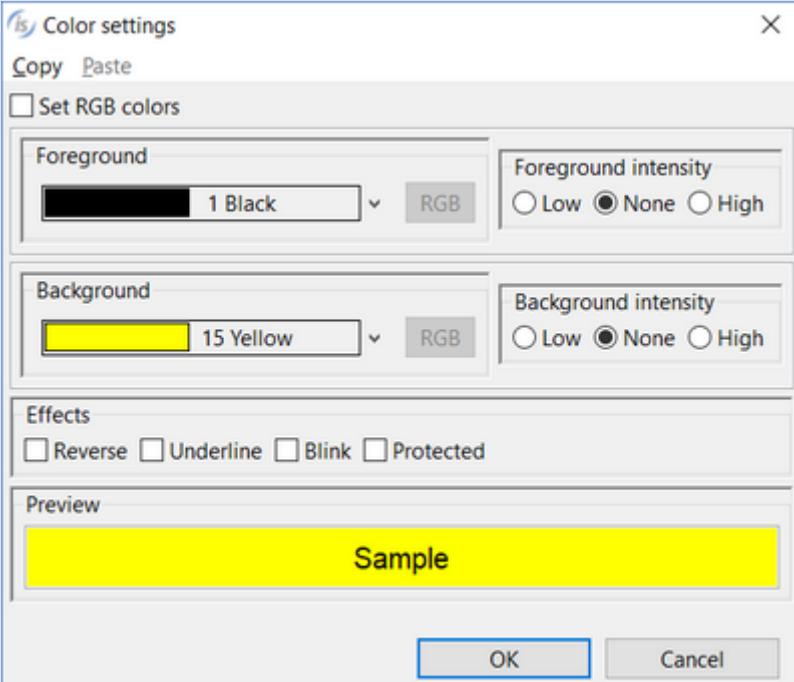
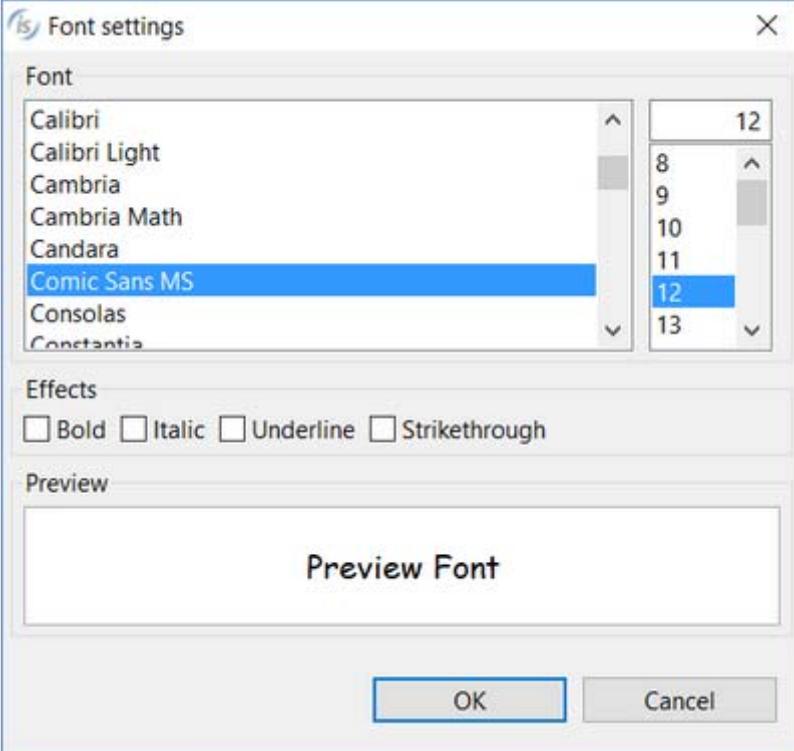
WINDOW

Refer to [WINDOW](#) for details about properties, styles and events of this control.

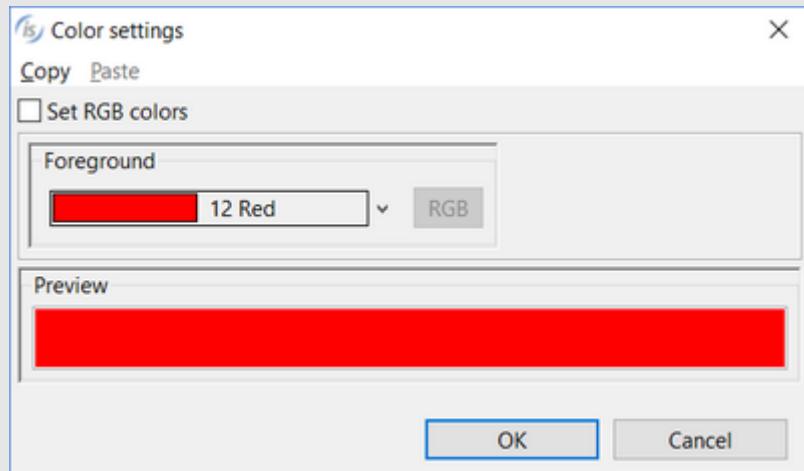
Properties	
(name)	Specifies the window name. This property is automatically set as soon as the window is drawn
action	Allows you to set the initial state of the window. The choices are as follows: None (default) ACTION-MAXIMIZE ACTION-MINIMIZE
additional properties	Allows the user to specify additional properties and styles. The text you write here is generated as is and may generate compile errors if not correct.
allowing messages	Allows you to set the value for the ALLOWING MESSAGES clause. The choices are as follows: None (default) LAST THREAD ANY THREAD

auto fit	TRUE... the background image, if any, is resized to fit the window FALSE... the background image, if any, is not resized
auto minimize	TRUE... the AUTO-MINIMIZE style is generated FALSE... the AUTO-MINIMIZE style is not generated
auto-resize	TRUE... the AUTO-RESIZE style is generated FALSE... the AUTO-RESIZE style is not generated
background-color	Opens a dialog that allows the user to choose the window background color. 
background image	Allows you to choose a picture that will be used as a background for the window 
background intensity	None ... no background intensity clauses are generated in the DISPLAY WINDOW statement LOW... BACKGROUND-LOW is generated in the DISPLAY WINDOW statement HIGH... BACKGROUND-HIGH is generated in the DISPLAY WINDOW statement STANDARD... BACKGROUND-STANDARD is generated in the DISPLAY WINDOW statement
before time	Sets the timeout in hundreds of seconds for the Accept on this screen
bind to thread	TRUE... The BIND TO THREAD clause is generated FALSE... The BIND TO THREAD clause is not generated
boxed	TRUE... The BOXED clause is generated FALSE... The BOXED clause is not generated
cell-height	Sets the CELL HEIGHT value

cell measure	PIXELS ... the cell size is as expressed in pixels MEASURING CONTROL FONT... the font is used to calculate cell size
cell measuring control	None ... nothing is generated LABEL... CELL SIZE IS LABEL FONT is generated ENTRY-FIELD... CELL SIZE IS ENTRY-FIELD FONT is generated
cell measuring font	Allows you to choose a font to measure the cell size:
	
cell measuring style	NONE... nothing is generated after CELL SIZE OVERLAPPED... the OVERLAPPED clause is generated after CELL SIZE SEPARATE... the SEPARATE clause is generated after CELL SIZE
cell-width	Sets the CELL WIDTH value

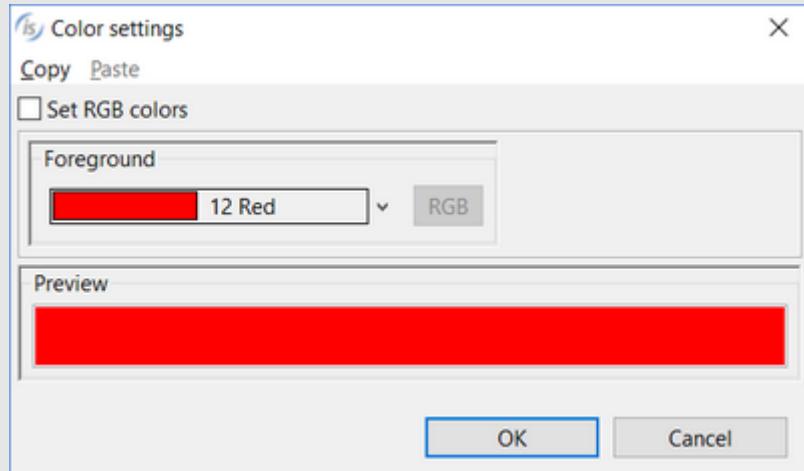
color	Opens a dialog that allows the user to choose the window color.
	
control font	Opens a dialog that allows the user to choose the control font.
	

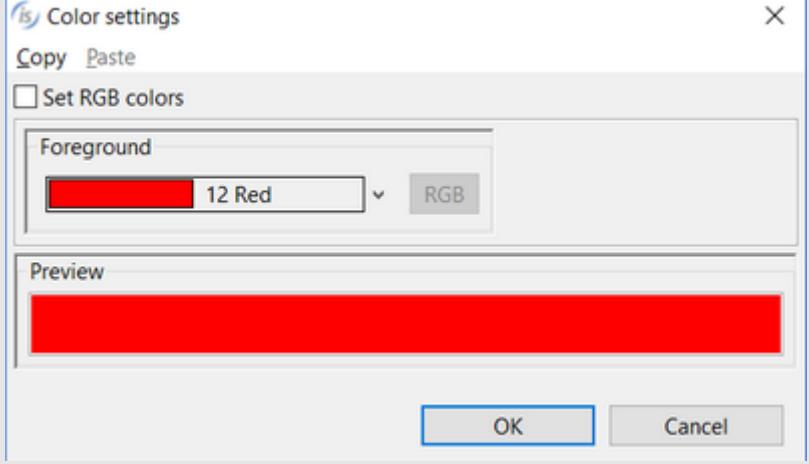
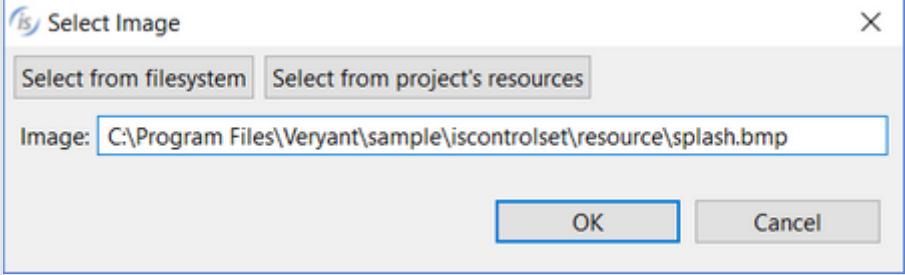
controls uncropped	TRUE... The CONTROLS-UNCROPPED clause is generated FALSE... The CONTROLS-UNCROPPED clause is not generated
custom-data	Specifies the value for the <i>Custom-Data</i> property.
enabled	NONE...The <i>Enabled</i> property is not generated TRUE... <i>Enabled=1</i> is generated FALSE... <i>Enabled=0</i> is generated
erase screen	TRUE... The ERASE clause is generated FALSE... The ERASE clause is not generated
foreground-color	Opens a dialog that allows the user to choose the window foreground color.

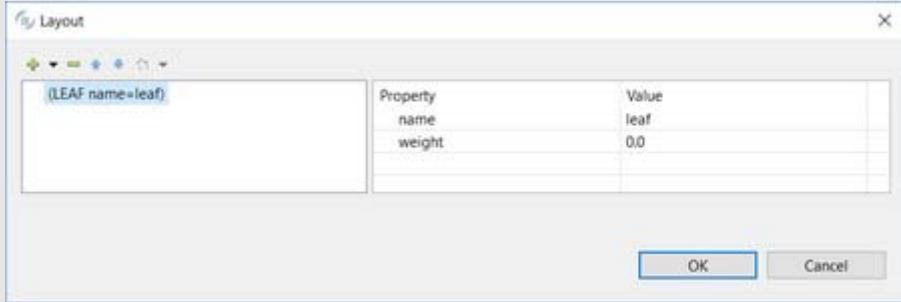
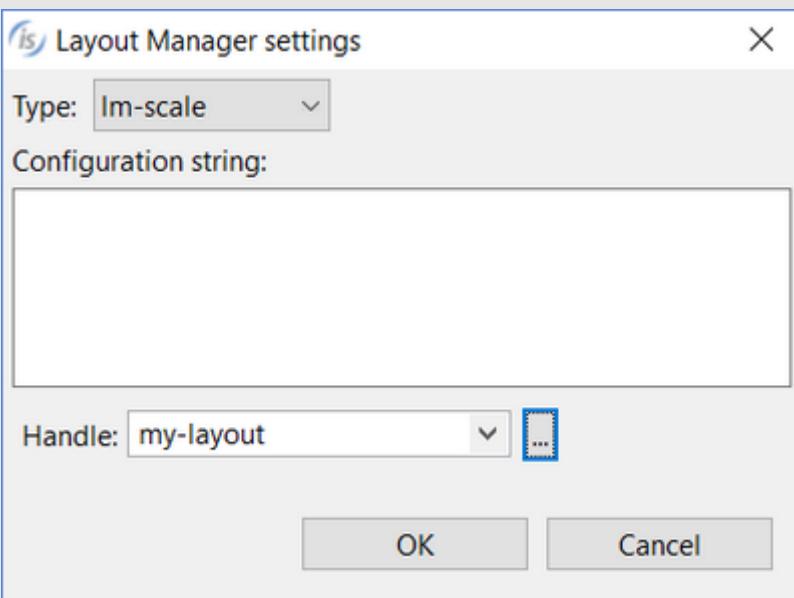


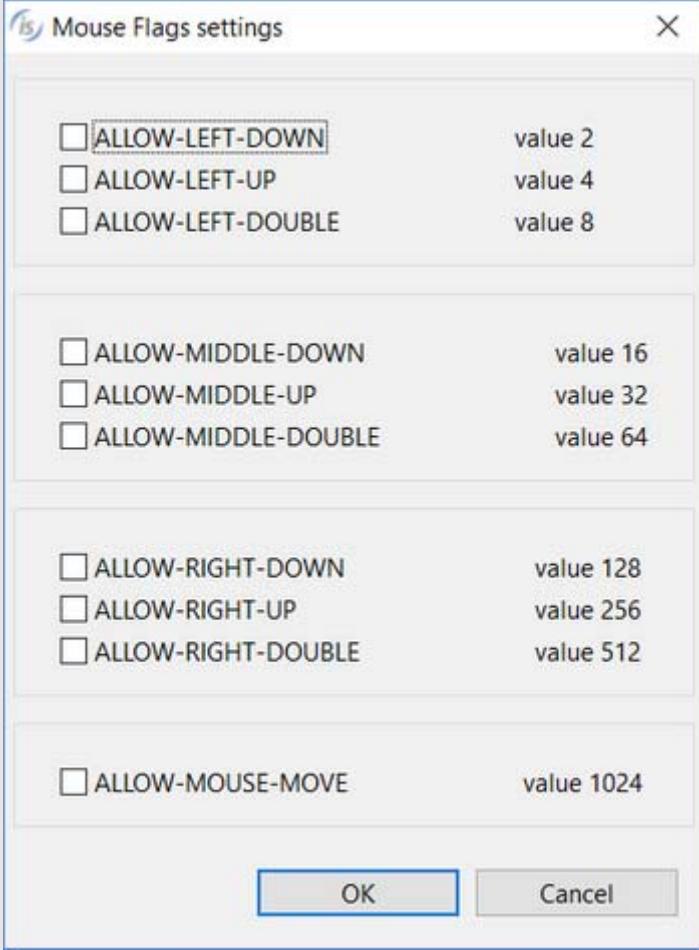
generate display statement TRUE... The DISPLAY WINDOW statement is generated
FALSE... The DISPLAY WINDOW statement is not generated

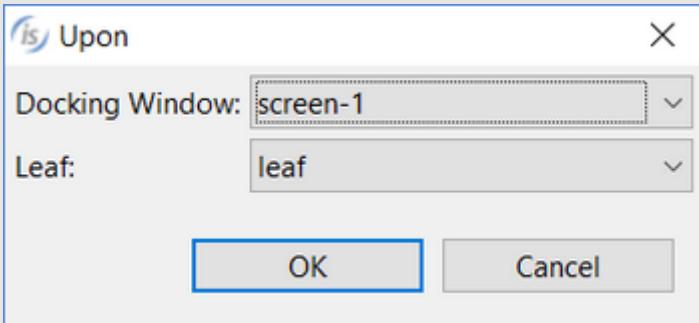
gradient-color-1 Opens a dialog that allows the user to choose the window gradient start color.



gradient-color-2	Opens a dialog that allows the user to choose the window gradient end color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has a 'Foreground' section with a color swatch labeled '12 Red' and an 'RGB' button. Below it is a 'Preview' section showing a horizontal bar filled with red. At the bottom are 'OK' and 'Cancel' buttons.</p>
gradient-orientation	<p>Specifies the gradient orientation. Possible values are:</p> <ul style="list-style-type: none"> None 0: NORTH-TO-SOUTH 1: NORTHEAST-TO-SOUTHWEST 2: EAST-TO-WEST 3: SOUTHEAST-TO-NORTHWEST 4: SOUTH-TO-NORTH 5: SOUTHWEST-TO-NORTHEAST 6: WEST-TO-EAST 7: NORTHWEST-TO-SOUTHEAST
graphical	<p>TRUE... The GRAPHICAL clause is generated FALSE... The GRAPHICAL clause is not generated</p>
height-in-cells	<p>TRUE... The HEIGHT-IN-CELLS clause is generated FALSE... The HEIGHT-IN-CELLS clause is not generated</p>
help-id	Specifies the control <i>help-id</i> .
icon	Opens a dialog that allows you to choose an image to be used as the window icon:
	 <p>The screenshot shows the 'Select Image' dialog box. It has two buttons: 'Select from filesystem' and 'Select from project's resources'. Below them is an 'Image:' label followed by an input field containing the path 'C:\Program Files\Veryant\sample\iscontrolset\resource\splash.bmp'. At the bottom are 'OK' and 'Cancel' buttons.</p>
label-offset	Sets the value for the <i>LABEL-OFFSET</i> property

layout	Opens a dialog that allows you to define a layout for the Docking window.
	
layout manager	Opens a dialog that allows to choose which layout manager should be associated to the window. When either LM-SCALE or LM-RESPONSIVE is selected, it's possible to specify the configuration string. In this dialog you also associate a handle to the layout manager.
	
lines	Specifies the control height as expressed in cells
lines pixels	Specifies the control height as expressed in pixels
link to thread	TRUE... The LINK TO THREAD clause is generated FALSE... The LINK TO THREAD clause is not generated
lock	TRUE...Locks the control on the Screen Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Screen Designer by dragging it with the mouse
main menu	Associates a menu bar with the window. The menu must have been drawn on the same screen.
max lines	Sets the value for the MAX-LINES property
max size	Sets the value for the MAX-SIZE property

min lines	Sets the value for the MIN-LINES property
min size	Sets the value for the MIN-SIZE property
modeless	TRUE... The MODELESS clause is generated FALSE... The MODELESS clause is not generated
mouse flags	Opens a dialog that allows you to configure mouse flags
 <p>The screenshot shows a dialog titled "Mouse Flags settings". It contains four sections of checkboxes, each with a value listed to its right:</p> <ul style="list-style-type: none"> Section 1: ALLOW-LEFT-DOWN (value 2), ALLOW-LEFT-UP (value 4), ALLOW-LEFT-DOUBLE (value 8) Section 2: ALLOW-MIDDLE-DOWN (value 16), ALLOW-MIDDLE-UP (value 32), ALLOW-MIDDLE-DOUBLE (value 64) Section 3: ALLOW-RIGHT-DOWN (value 128), ALLOW-RIGHT-UP (value 256), ALLOW-RIGHT-DOUBLE (value 512) Section 4: ALLOW-.MOUSE-MOVE (value 1024) <p>At the bottom are "OK" and "Cancel" buttons.</p>	
no close	TRUE... The NO-CLOSE clause is generated FALSE... The NO-CLOSE clause is not generated
pop up menu	Associates a pop-up menu to the window. The menu must have been drawn on the same screen.
resizable	TRUE...RESIZABLE is not generated. FALSE...RESIZABLE is generated.
screen column	Specifies the value of <i>Screen Column</i> property as expressed in cells.
screen column pixels	Specifies the value of <i>Screen Column</i> property as expressed in pixels.
screen line	Specifies the value of <i>Screen Line</i> property as expressed in cells.

screen line pixels	Specifies the value of <i>Screen Line</i> property as expressed in pixels.
scroll	TRUE... <i>WITH NO SCROLL</i> is not generated. FALSE... <i>WITH NO SCROLL</i> is generated.
size	Specifies the window width as expressed in cells.
size pixels	Specifies the window width as expressed in pixels.
system-menu	TRUE... <i>WITH SYSTEM MENU</i> is generated FALSE... <i>WITH SYSTEM MENU</i> is not generated
title	Specifies the window title
title-bar	TRUE... <i>TITLE-BAR</i> is generated FALSE... <i>TITLE-BAR</i> is not generated
undecorated	TRUE... The <i>UNDECORATED</i> style is generated FALSE...The <i>UNDECORATED</i> style is not generated
unit	Allows you to choose the measuring unit. The choices are as follows: CELLS PIXELS
upon	Allows you to choose the MDI-PARENT window for the current MDI-CHILD window.
upon leaf	Opens a dialog that allows you to choose the destination leaf for the Dockable window.
	
user gray	TRUE... <i>USER GRAY</i> is not generated FALSE... <i>USER GRAY</i> is generated
user white	TRUE... <i>USER WHITE</i> is not generated FALSE... <i>USER WHITE</i> is generated
visible	NONE...The <i>Visible</i> property is not generated TRUE... <i>Visible=1</i> is generated FALSE... <i>Visible=0</i> is generated
width-in-cells	TRUE... The <i>WIDTH-IN-CELLS</i> clause is generated FALSE... The <i>WIDTH-IN-CELLS</i> clause is not generated
window handle	Allows to customize the name of the window handle variable

window type	Allows you to choose the window type. The choices are as follows: STANDARD INITIAL INDEPENDENT FLOATING DOCKING DOCKABLE MDI-PARENT MDI-CHILD
wrap	TRUE... NO WRAP is not generated FALSE...NO WRAP is generated
Events	
cmd-activate event	Allows the user to create a paragraph to handle the CMD-ACTIVATE event in the Procedure Division
cmd-close event	Allows the user to create a paragraph to handle the CMD-CLOSE event in the Procedure Division
msg-close event	Allows the user to create a paragraph to handle the MSG-CLOSE event in the Procedure Division
msg-deiconified event	Allows the user to create a paragraph to handle the MSG-DEICONIFIED event in the Procedure Division
msg-end-menu event	Allows the user to create a paragraph to handle the MSG-END-MENU event in the Procedure Division
msg-iconified event	Allows the user to create a paragraph to handle the MSG-ICONIFIED event in the Procedure Division
msg-init-menu event	Allows the user to create a paragraph to handle the MSG-INIT-MENU event in the Procedure Division
msg-menu-input event	Allows the user to create a paragraph to handle the MSG-MENU-INPUT event in the Procedure Division
ntf-resized event	Allows the user to create a paragraph to handle the NTF-RESIZED event in the Procedure Division
other event	Allows the user to create a custom paragraph
Exceptions	
cmd-activate exception	Allows the user to create a paragraph to handle the CMD-ACTIVATE event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
cmd-close exception	Allows the user to create a paragraph to handle the CMD-CLOSE event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
ntf-resized exception	Allows the user to create a paragraph to handle the NTF-RESIZED event when the Accept terminates with crt status = 96. This is an alternative to the event procedures described above
other exception	Allows the user to create a custom paragraph
Procedures	

After Create	Allows you to create a paragraph that will be executed after the display of the window
After InitData	Allows you to create a paragraph that will be executed after opening files and loading resources
After procedure	Allows the user to create a paragraph to handle the screen AFTER PROCEDURE
After procedure thru	Allows the user to optionally specify a THRU paragraph for the AFTER PROCEDURE.
After Routine	Allows you to create a paragraph that will be executed after the Accept on the screen
Before Create	Allows you to create a paragraph that will be executed before the display of the window
Before InitData	Allows you to create a paragraph that will be executed before opening files and loading resources
Before procedure	Allows the user to create a paragraph to handle the screen BEFORE PROCEDURE
Before procedure thru	Allows the user to optionally specify a THRU paragraph for the BEFORE PROCEDURE.
Before Routine	Allows you to create a paragraph that will be executed before the Accept on the screen
Event procedure	Allows the user to create a paragraph to handle the screen EVENT PROCEDURE
Exception procedure	Allows the user to create a paragraph to handle the screen EXCEPTION PROCEDURE
Link To	Allows the user to create a paragraph that will be executed when an exception occurs in the Accept
Variables	
allowing messages thread	Numeric variable that hosts the handle of the thread whose messages should be received
background-color variable	Numeric variable that hosts the value for the <i>Background-Color</i> property
before time variable	Numeric variable that hosts the Accept timeout
cell-height variable	Numeric variable that hosts the value for the <i>Cell Height</i> property
cell-width variable	Numeric variable that hosts the value for the <i>Cell Width</i> property
color variable	Numeric variable that hosts the color value
column variable	Numeric variable that hosts the value for the <i>Column</i> property
custom-data variable	Alphanumeric variable that hosts the value of the <i>Custom-Data</i> property
enabled variable	Numeric variable that hosts the enabled state
gradient-color-1 variable	Numeric variable that hosts the value for the <i>Gradient-Color-1</i> property
gradient-color-2 variable	Numeric variable that hosts the value for the <i>Gradient-Color-2</i> property
gradient-orientation variable	Numeric variable that hosts the value for the <i>Gradient-Orientation</i> property
lines variable	Numeric variable that hosts the lines value
line variable	Numeric variable that hosts the value for the <i>Line</i> property

max lines variable	Numeric variable that hosts the value for the <i>Max-Lines</i> property
max size variable	Numeric variable that hosts the value for the <i>Max-Size</i> property
min lines variable	Numeric variable that hosts the value for the <i>Min-Lines</i> property
min size variable	Numeric variable that hosts the value for the <i>Min-Size</i> property
mouse flags variable	Numeric variable that hosts the value for the <i>Mouse-Flags</i> property
screen column variable	Numeric variable that hosts the value for the <i>Screen Column</i> property
screen index variable	Numeric variable that hosts the value for the <i>Screen Index</i> property
screen line variable	Numeric variable that hosts the value for the <i>Screen Line</i> property
size variable	Numeric variable that hosts the size value
title variable	Numeric variable that hosts the value for the <i>Title</i> property
upon variable	Numeric variable that hosts the handle of the parent window
visible variable	Numeric variable that hosts the visible state

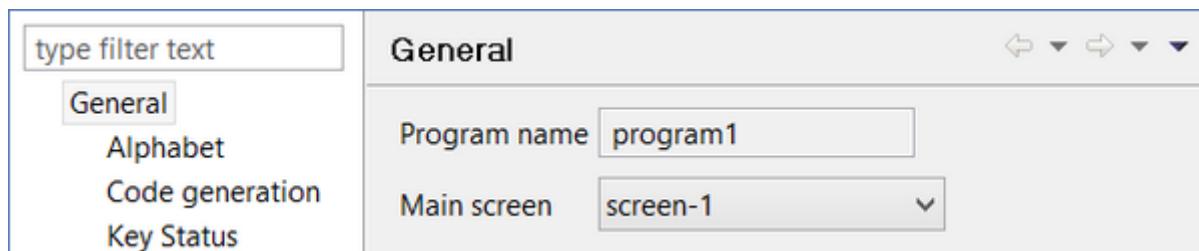
Adding a new Screen to the program

A program can contain more than one screen. To add screens in addition to the one created with the program:

1. Right click on the program name.
2. Select *New / Screen* from the pop-up menu.

The wizard procedure allows you to choose between creating a blank screen or starting from a template. See [Loading Screen Templates](#) for more details about templates.

When the program contains more than one screen, it's necessary to set which one is the main screen. To specify the main screen, right click on the program name and select *Properties* from the pop-up menu.

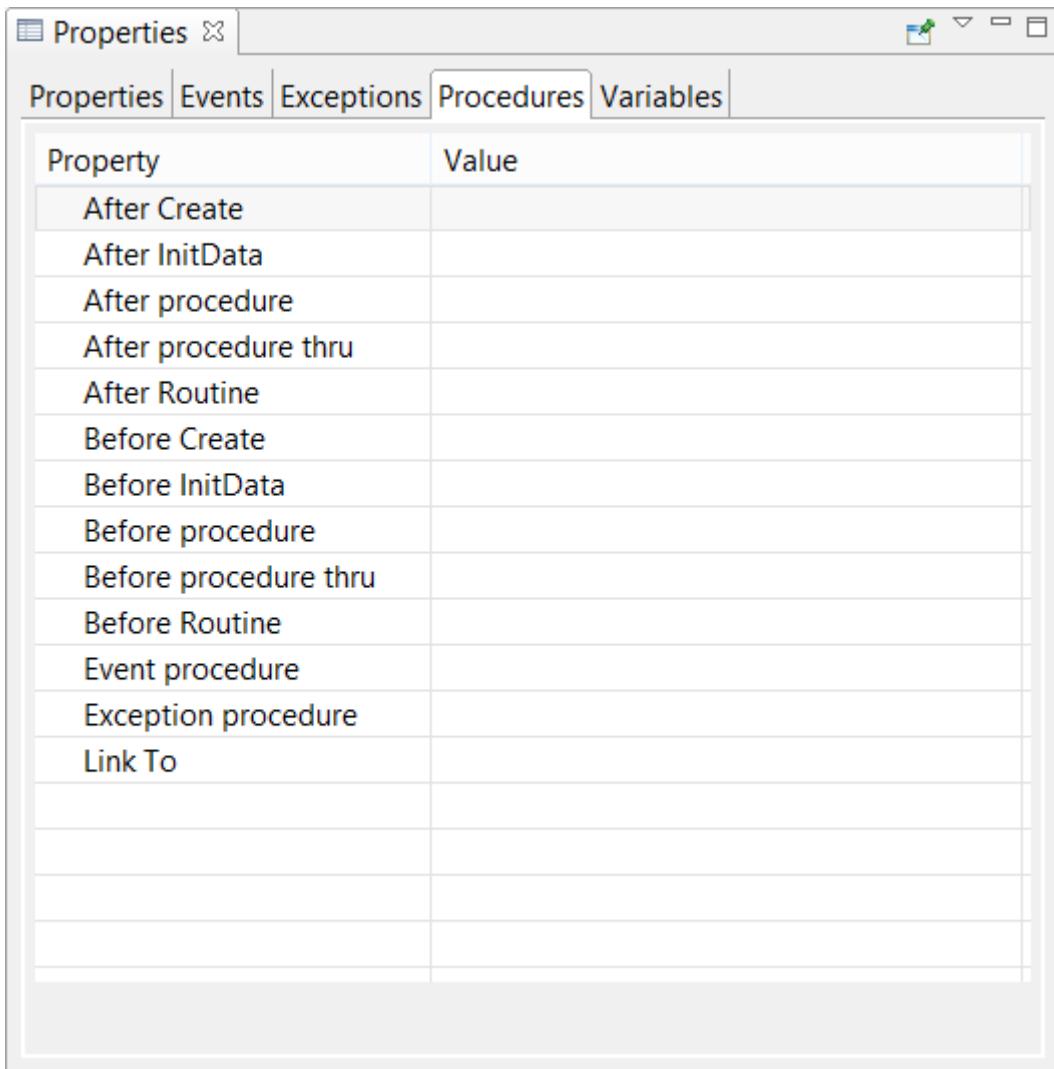


When the program contains more than one screen, it's very important to set the proper window type for each one of the screens. The main screen should be STANDARD or INITIAL, while the other ones must be FLOATING or INDEPENDENT. To set the window type:

1. Select the window by clicking on its title bar in the Screen Designer.
2. Set the *window type* property in the Properties view.

Screen Entry Points

In the Screen properties the following entry points are available:



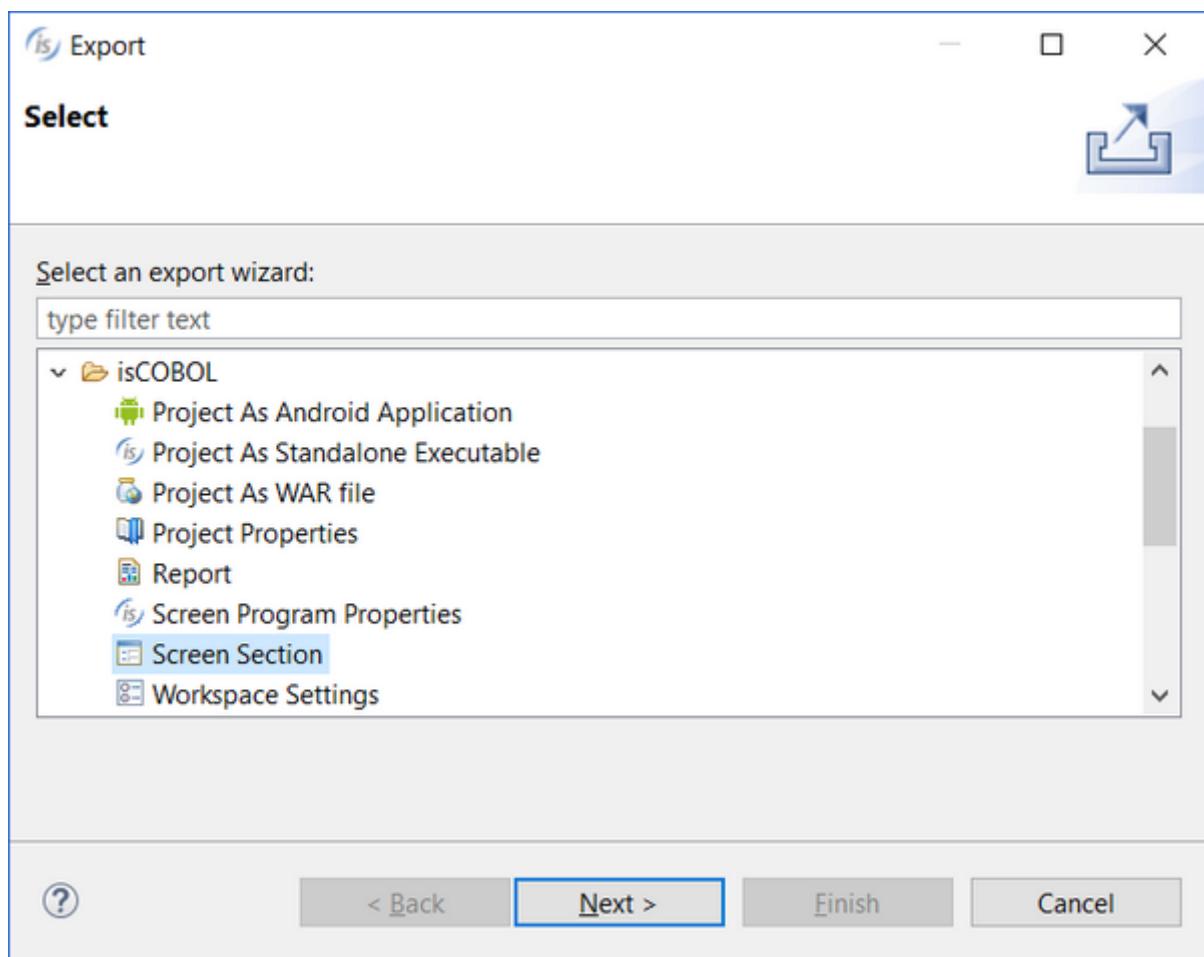
They allow you to insert code that will be executed at the program startup (Before Program) and before the program exit (After Program).

Import / Export of Screens

Screens can be exported to external files and imported in other projects.

To export one or more screens:

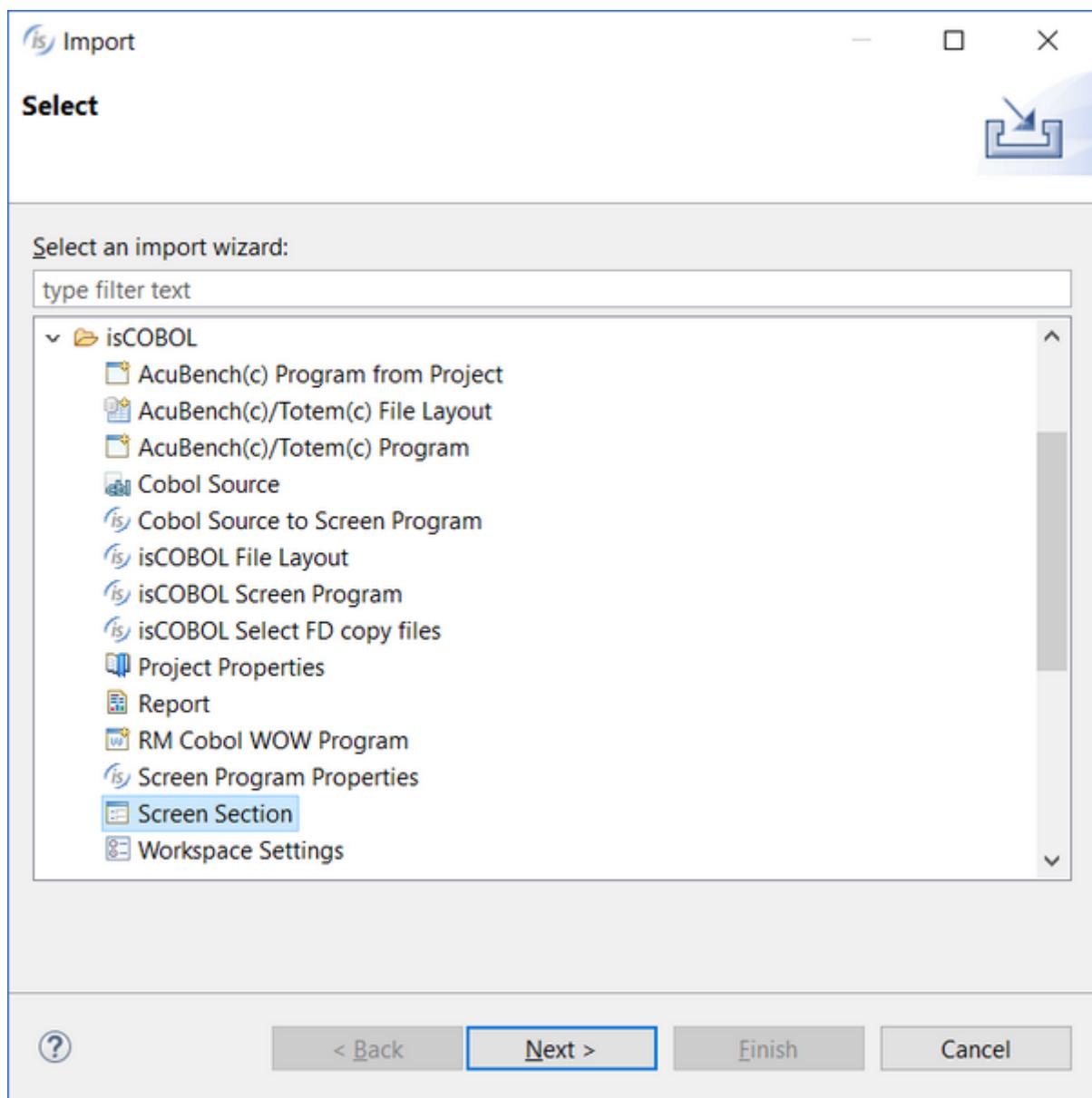
1. right click on the program name in the isCOBOL Explorer
2. choose *Export*
3. expand *isCOBOL*
4. select *Screen Section*



5. click *Next*
6. select the screens you want to export
7. click *Next*
8. choose the destination file (the file must have .isl extension)
9. click *Finish*

To import a .isl file and have its screens added to your program:

1. right click on the program name in the isCOBOL Explorer
2. choose *Import*
3. expand *isCOBOL*
4. select *Screen Section*



5. click *Next*
6. browse for the directory where .isl files are
7. select the screens you wish to import
8. click *Finish*

The isl file can also be used as a template for new screens. See [Loading Screen Templates](#) for more information.

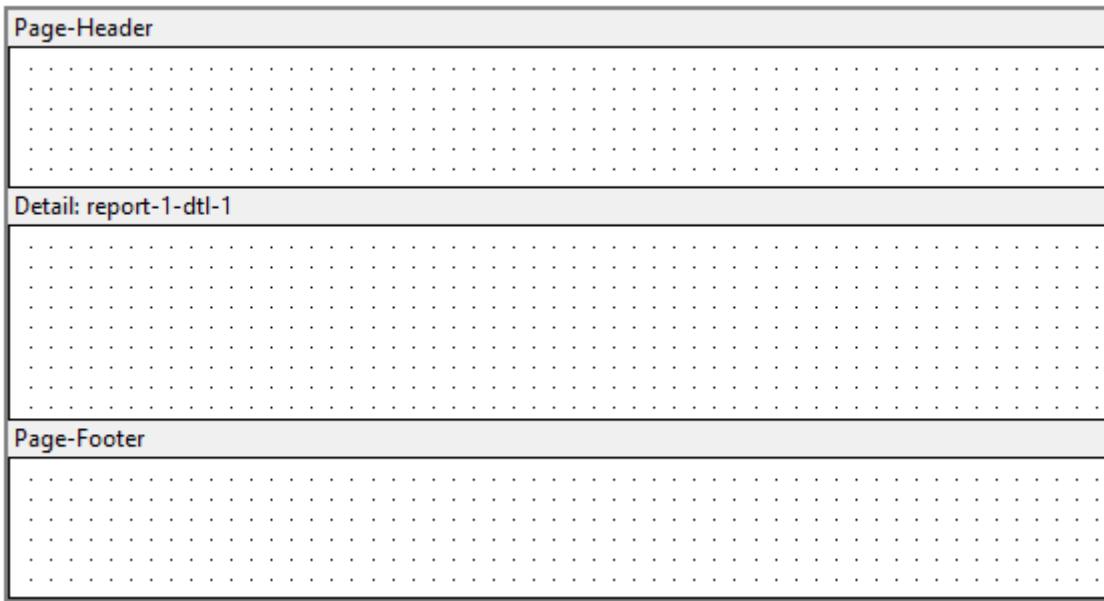
Reports

The IDE Report Designer is an interface for report design and layout that is very similar in appearance and behavior to the IDE Screen Designer.

Creating a new Report

To create a new Report, right click on the program name in the isCOBOL Explorere area and choose *New / Report* from the pop-up menu. Choose the name of your report and click *Finish*.

The new empty Report looks like this:



Report Sections

By default, a new Report has three sections: a page header, a detail section, and a page footer. With the Section Controller interface, you can add report headers and footers, multiple detail sections, and group sections that define breakpoints for the information in the Report.

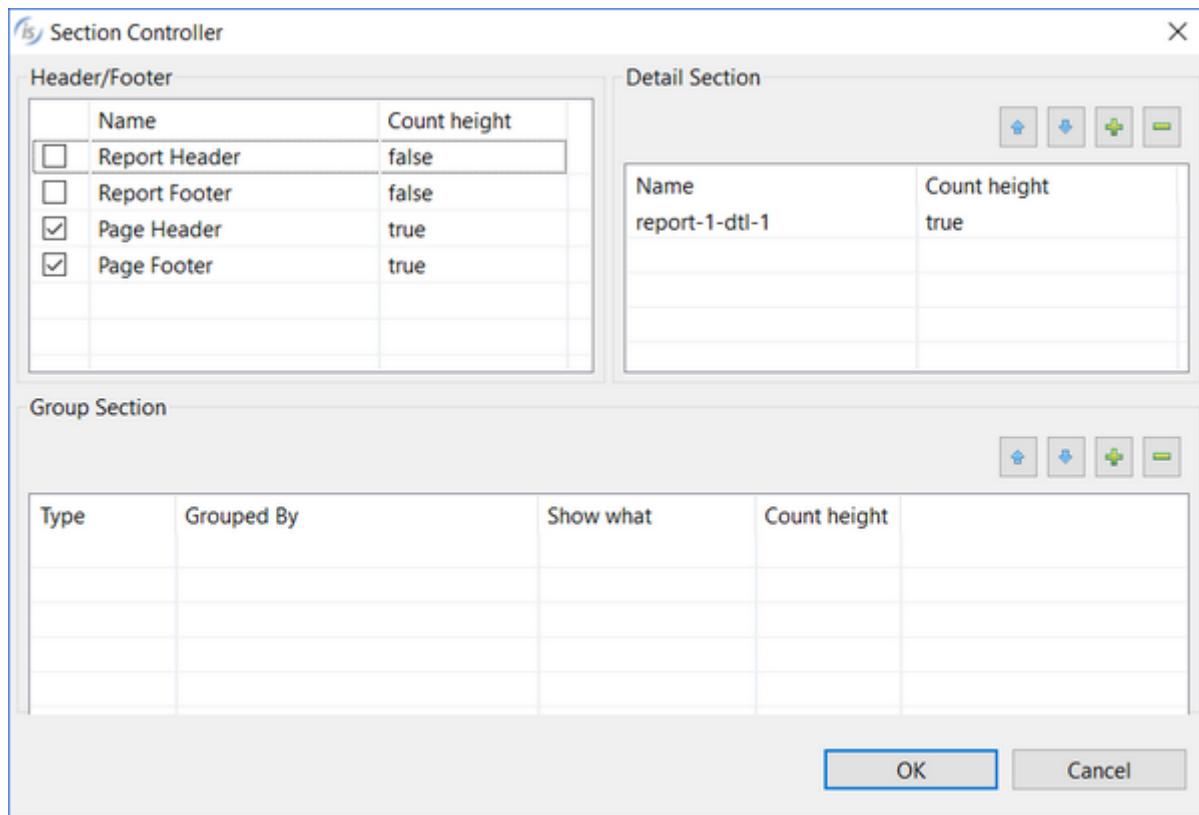
The report header is printed once at the beginning of a Report, and the report footer is printed once at the end of a Report. You define the components and appearance of your report header or footer in the corresponding Property window. Add a report header or footer to the Report via the [Section Controller](#) dialog.

A group header prints at the beginning of a marked breakpoint, and a group footer prints at the end of a marked breakpoint. You may have multiple group headers or group footers in a Report, representing multiple breakpoints. Where multiple breakpoints are named, the first named is the primary breakpoint, and subsequent breakpoints are always secondary to the breakpoint preceding them. When multiple group footers appear in a Report, they appear in reverse order in the Report Designer, with the secondary group footers appearing above the primary group footer. You add a group header or group footer to the Report via the [Section Controller](#) dialog.

The detail section prints repeatedly, providing the basic reporting function. This section constitutes the main body of the Report. Code for the detail section is generated into an ".rpt" COPY file. By default, one detail section appears in the Report Composer window when you create a new Report. You may have multiple detail sections in a Report, for example, if you wish to present your basic detail information in two parts and you choose a different font or background color for each part. You add detail sections via the [Section Controller](#) dialog.

Section Controller

Right-click in the Report Designer and choose *Section Controller* from the pop-up menu. Use this dialog to define what headers and footers should be printed.



Drawing the Report

To draw a control on the Report:

1. Click to select the control you want to use. All the controls are on the Controls Palette. Make sure you release the mouse button before going to step
2. Click and keep the left mouse button pressed on the position you want to place the control, this is going to be the upper left corner of your control.
3. Drag the mouse to the lower right corner of your control (it will feel like you are sizing the control). Make sure you do not release the left mouse button in between step 2 and step 3.
4. Release the left button and see that the control has been successfully drawn

By drawing the control you automatically set LINE, COL, LINES and SIZE properties. All other properties are set to a default value that can be configured by clicking on the Window menu, choosing *Preferences* and then selecting *isCOBOL / Report Designer / Default* from the tree.

The control is automatically named with the following pattern: <report_name>-<two-digits-control-type>-<progressive-number>. For example, when you draw the first label on report-1, it will be named "report-1-lb-1". If you wish to assign a different prefix than the report name to the control name, you can right click on the Report Designer area and choose "Change Report Prefix". When you change the prefix, all existing controls in

the report are automatically renamed, unless you check the option "Change report's prefix only", in this case only the new controls will use the new prefix. You can always rename a single control by editing its "(name)" property.

To change the property values for one or more controls drawn on the screen:

1. Select the control(s).
 - a. The control is automatically selected when drawn.
 - b. To select more than one control, hold Ctrl and click on the desired controls.
2. Change the properties values in the Properties view.

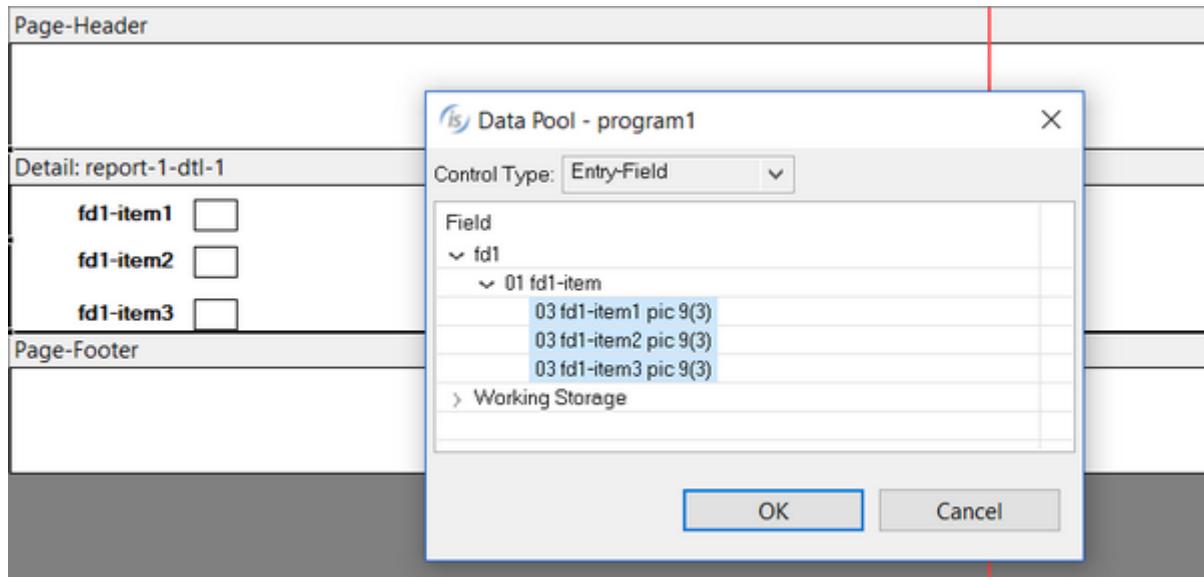
To easily change LINE, COL, LINES and SIZE

1. Select the control
2. Use arrow keys to move the control on the Report (LINE and COL are updated)
- c. Hold SHIFT and use arrow keys to change the control dimensions (LINES and SIZE are updated)

The Data Pool feature

The quickest way to draw a control in the Report Designer is by taking advantage of the Data Pool feature. The Data Pool allows you to draw a control starting from a data-item defined in the program.

Right click on the screen and select "DataPool" from the pop-up menu. The following dialog pops up.



Select the data-items that you wish to handle through report controls. Select the control type from the list on the top. Drag the data items on the screen with the mouse. When you release the left mouse button the desired controls appears on the report.

Report Designer Reference

This section lists all the Properties you can set for controls drawn on the Report and for the Report itself.

Property view items are grouped in the following categories:

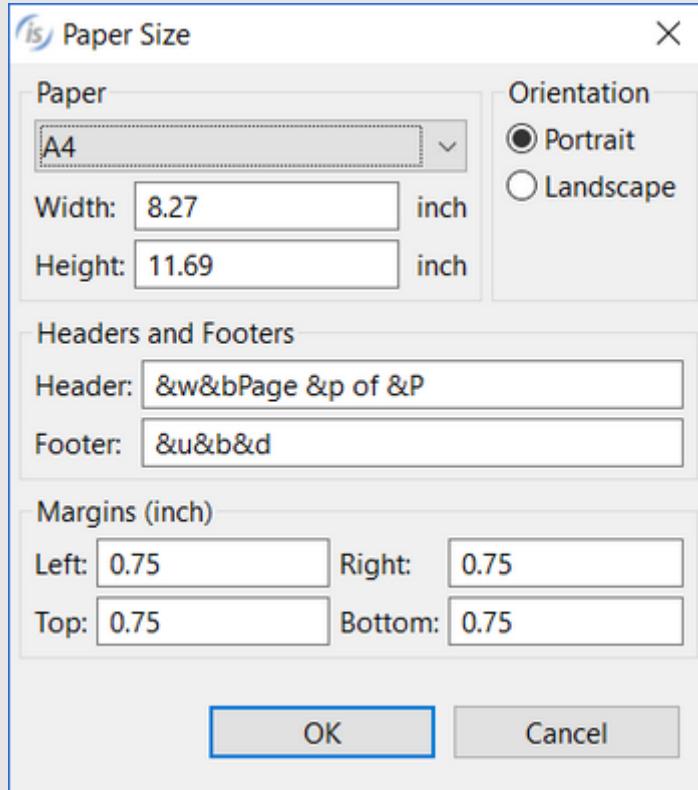
Properties	Control properties constant values and styles of the control
Events	Always empty
Exceptions	Always empty
Procedures	Paragraphs executed before and after printing the Report item
Variables	Variables for some of the properties of the control

REPORT

Properties	
(name)	Specifies the report name. This property is set automatically when the report is created.
output file name	Specifies the name and path of the HTML file generated on disk for this report. Since the content of this file is HTML code, you should give this file a ".html" extension. If this property is not set, then "<ReportName>.html" is assumed. If the report file name doesn't include an absolute path, the report is generated in the first path of the FILE-PREFIX.
	Use the "@[display]:" prefix followed by a client side path (e.g. "@[display]:C:\Temp\myReport.html") to generate the report on the client machine when the program runs in a thin client environment. The "@[display]:" prefix is automatically ignored when running out of a thin client environment.

paper size

Opens a dialog that allows to set paper size and orientation.



The following escape sequences are supported in the Header and Footer fields in order to add useful information:

&p = page number;

&P = total number of pages;

&b = the following information will be printed on the right side of the sheet;

&d = current date in short format according to the locale, e.g. 4/16/14;

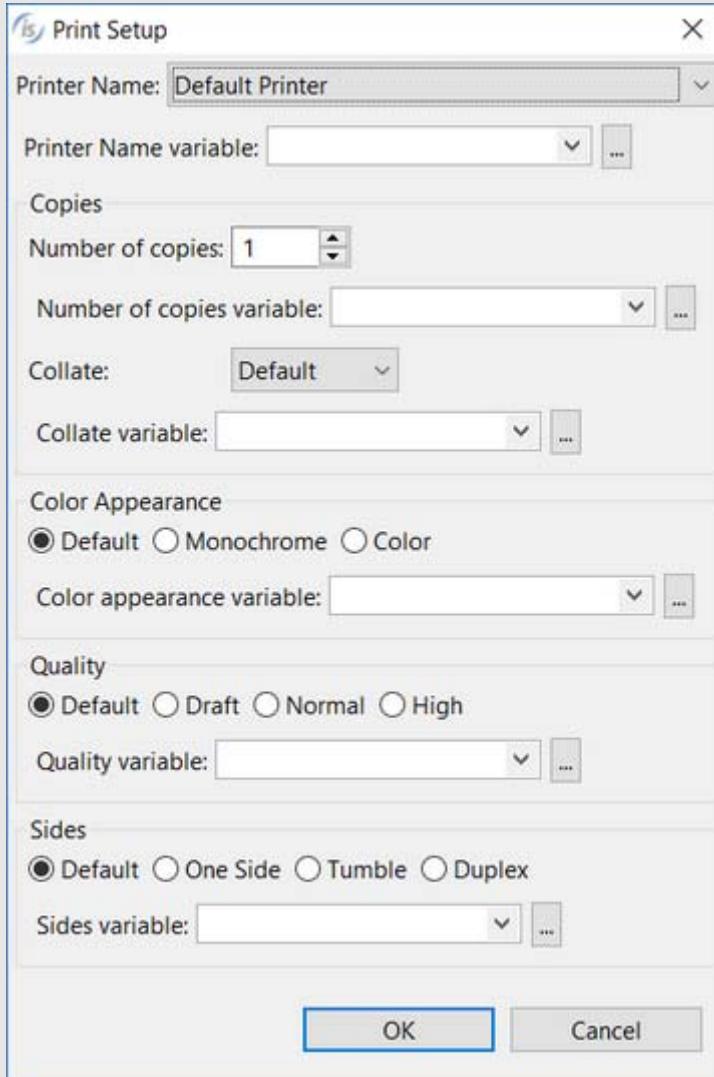
&D = current date in long format according to the locale, e.g. April 16, 2014;

&u = name of the report;

&w = not handled, ignored;

&& = the character '&'

Any other sequence of two characters starting with '=' will print a question mark (?).

print setup	Opens a dialog that allows to set up the printer.
	
title	Specifies the title of the HTML report. This title is shown in the browser title bar when the report is opened by a web browser.
watermark	Browse for a image file to be printed in the background of each page of the report.
watermark style	<p>None... the watermark image is not printed</p> <p>Center... the watermark image is printed at the center of the page</p> <p>Tile.... the watermark image is repeated in order to cover the whole page</p>
Events	No Events available.
Exceptions	No Exceptions available.

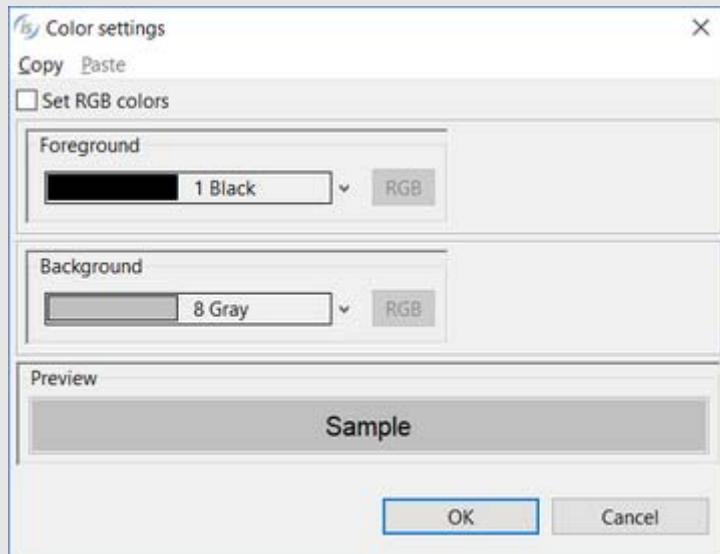
Procedures	
AfterDoPrint	This is a special paragraph that is executed automatically after the print of each detail. See How to print Reports for more details.
AfterPrint	Allows the user to create a paragraph that is performed after the report has been printed.
BeforeDoPrint	This is a special paragraph that is executed automatically when the print of the report is requested. See How to print Reports for more details.
BeforePrint	Allows the user to create a paragraph that is performed before printing the report.
Variables	
output file name	Specifies the name of the html file generated for this report.

HEADER

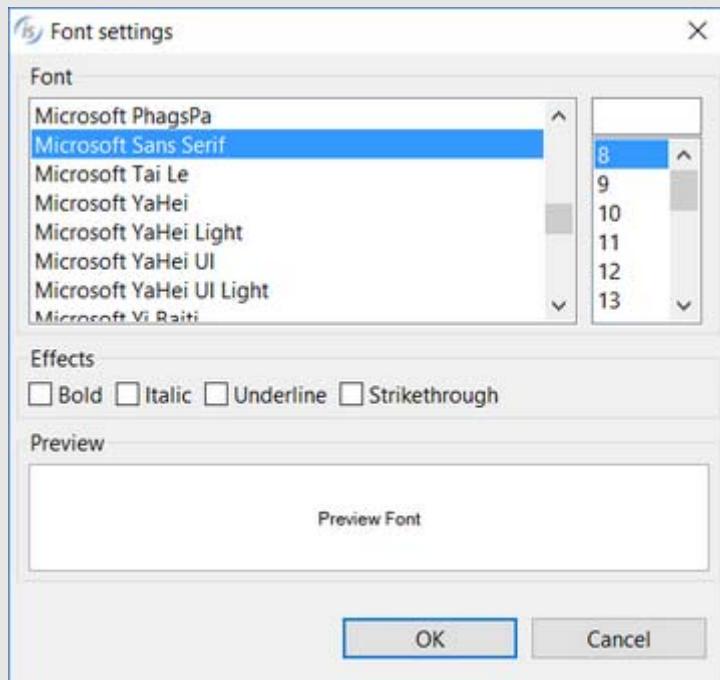
Properties

(name) Specifies the control name. This property is set automatically when the control is drawn.

color Opens a dialog that allows the user to choose the color.



font Opens a dialog that allows the user to choose the font.

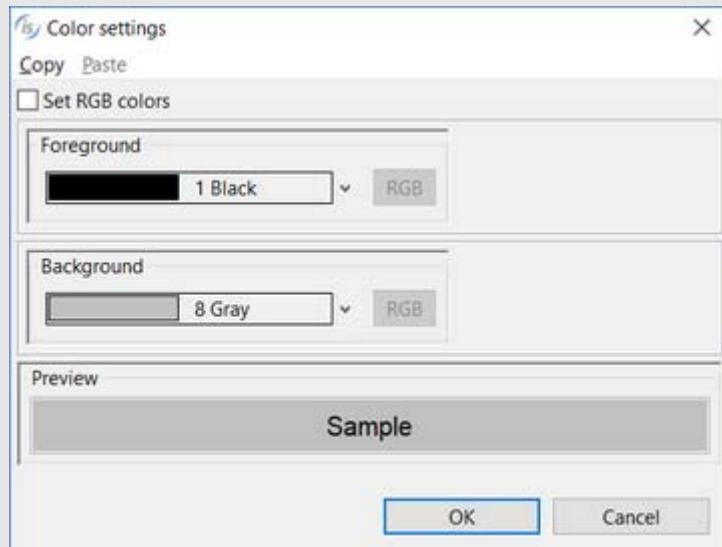


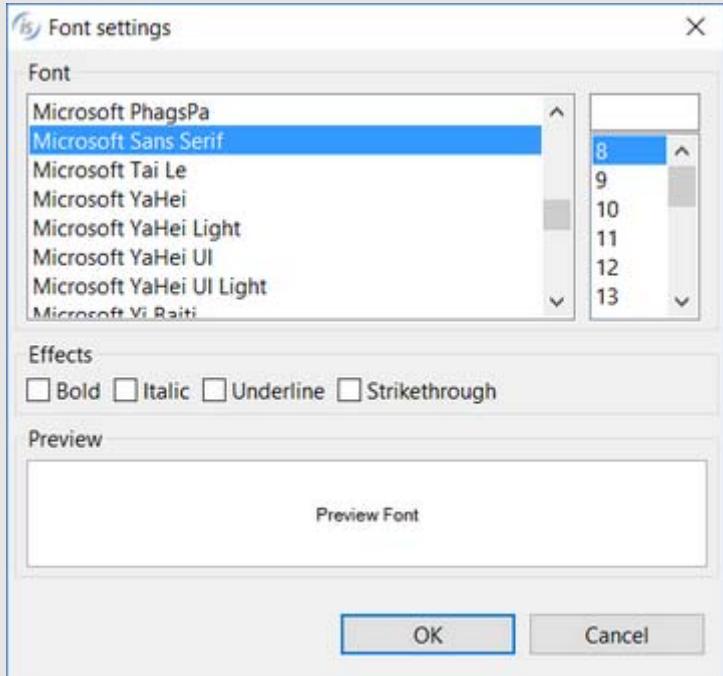
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print before page header	TRUE...The report header prints before the page header on the first page of the report. FALSE...The report header prints after the page header on the first page of the report.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
size	Specifies the width of the report item.
skip after print	TRUE...Prints a blank page after this report section FALSE...Doesn't print a blank page after this report section
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	
No Events available.	
Exceptions	
No Exceptions available.	
Procedures	
AfterPrint	Allows the user to create a paragraph that is performed after the report item has been printed.
BeforePrint	Allows the user to create a paragraph that is performed before printing the report item.
Variables	
color variable	Numeric variable that hosts the color value.
visible variable	Numeric variable that hosts the visible state.

PAGE-HEADER

Properties

(name)	Specifies the control name. This property is set automatically when the control is drawn.
color	Opens a dialog that allows the user to choose the color.



font	Opens a dialog that allows the user to choose the font.
	
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
size	Specifies the width of the report item.
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	No Events available.
Exceptions	No Exceptions available.
Procedures	
AfterPrint	Allows the user to create a paragraph that is performed after the report item has been printed.

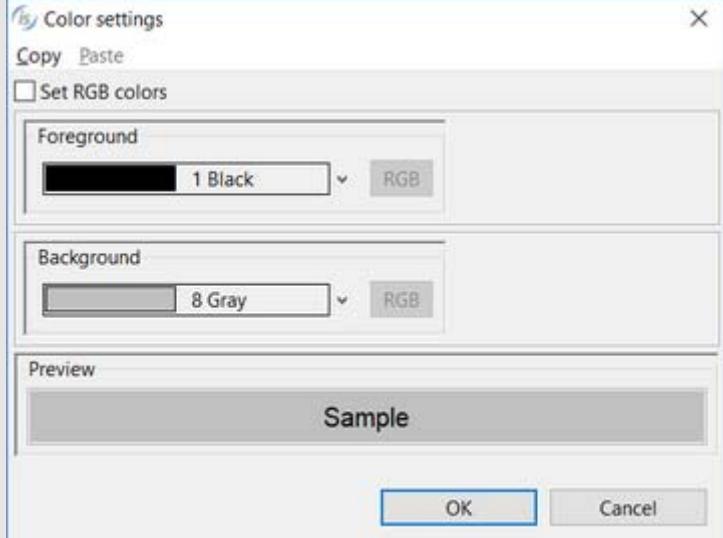
BeforePrint	Allows the user to create a paragraph that is performed before printing the report item.
-------------	--

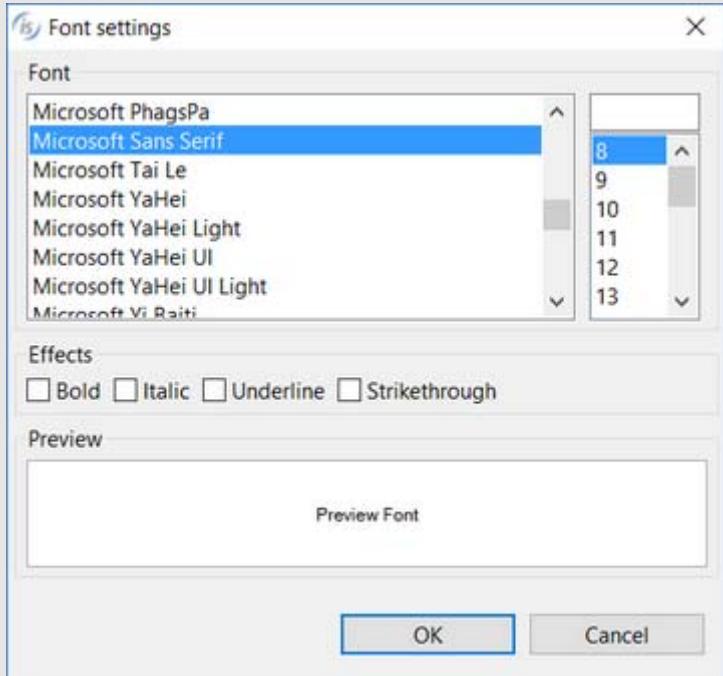
Variables

color variable	Numeric variable that hosts the color value.
----------------	--

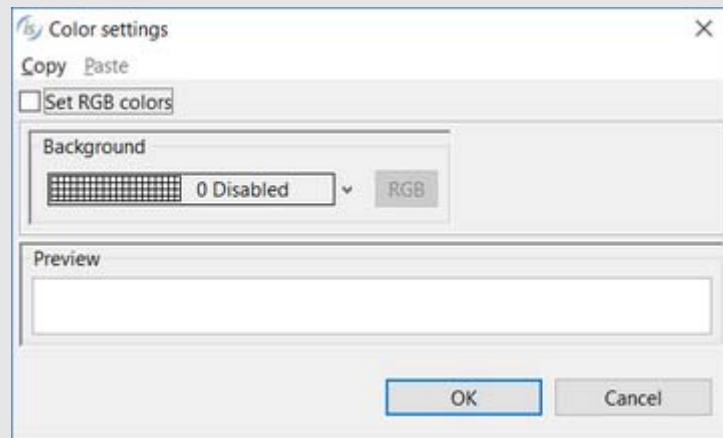
visible variable	Numeric variable that hosts the visible state.
------------------	--

DETAIL

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn.
color	Opens a dialog that allows the user to choose the color.  A screenshot of the 'Color settings' dialog box. At the top, there are 'Copy' and 'Paste' buttons and a checkbox labeled 'Set RGB colors'. Below this are two color selection boxes: 'Foreground' (set to '1 Black') and 'Background' (set to '8 Gray'). A preview window shows a gray bar labeled 'Sample'. At the bottom are 'OK' and 'Cancel' buttons.

font	Opens a dialog that allows the user to choose the font.
	
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
size	Specifies the width of the report item.
skip after print	TRUE...Prints a blank page after this report section FALSE...Doesn't print a blank page after this report section
visible	TRUE... The report item is visible FALSE... The report item is hidden

zebra Opens a dialog that allows to choose an additional color in order to produce a zebra effect.



Events

No Events available.

Exceptions

No Exceptions available.

Procedures

AfterPrint Allows the user to create a paragraph that is performed after the report item has been printed.

BeforePrint Allows the user to create a paragraph that is performed before printing the report item.

Variables

color variable Numeric variable that hosts the color value.

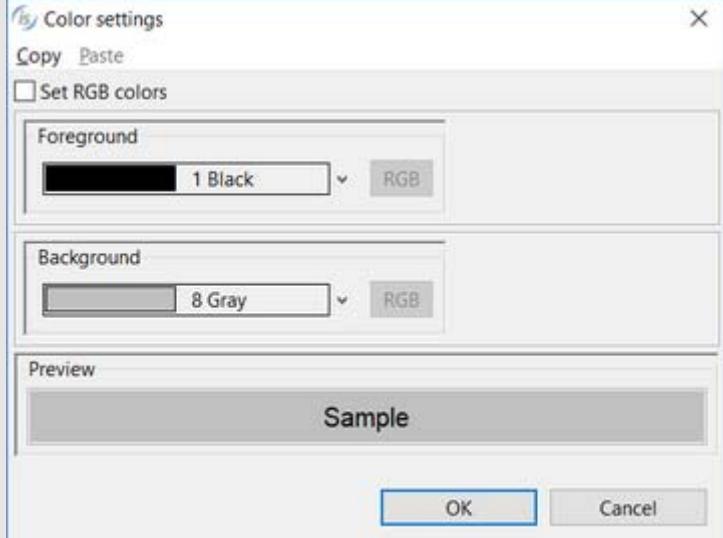
visible variable Numeric variable that hosts the visible state.

PAGE-Footer

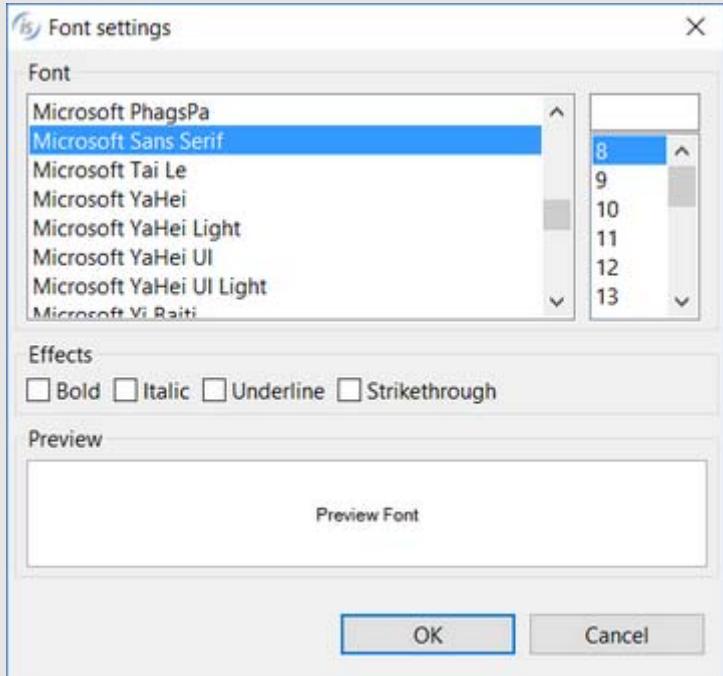
Properties

(name) Specifies the control name. This property is set automatically when the control is drawn.

color Opens a dialog that allows the user to choose the color.



The screenshot shows the 'Color settings' dialog box. It has sections for 'Foreground' and 'Background'. Under 'Foreground', a color swatch is black with the label '1 Black' and an 'RGB' button. Under 'Background', a color swatch is gray with the label '8 Gray' and an 'RGB' button. Below these is a 'Preview' section with a 'Sample' label. At the bottom are 'OK' and 'Cancel' buttons. The dialog has a standard Windows-style title bar with a close button.

font	Opens a dialog that allows the user to choose the font.
	
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
size	Specifies the width of the report item.
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	No Events available.
Exceptions	No Exceptions available.
Procedures	
AfterPrint	Allows the user to create a paragraph that is performed after the report item has been printed.

BeforePrint	Allows the user to create a paragraph that is performed before printing the report item.
-------------	--

Variables

color variable	Numeric variable that hosts the color value.
----------------	--

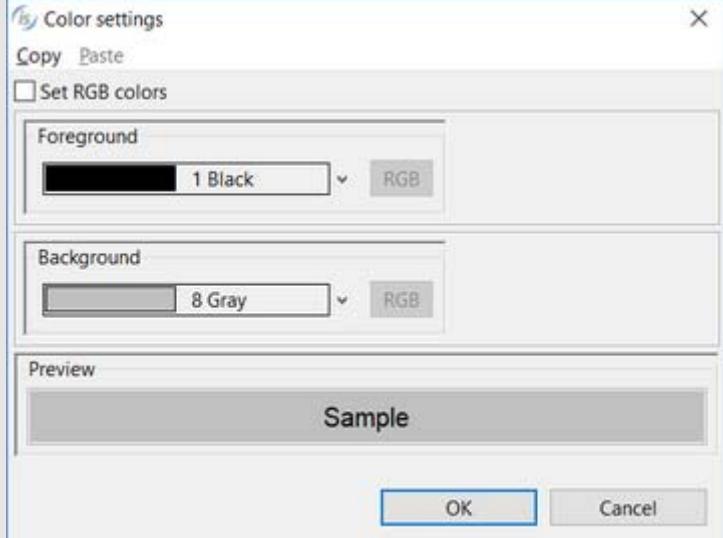
visible variable	Numeric variable that hosts the visible state.
------------------	--

FOOTER

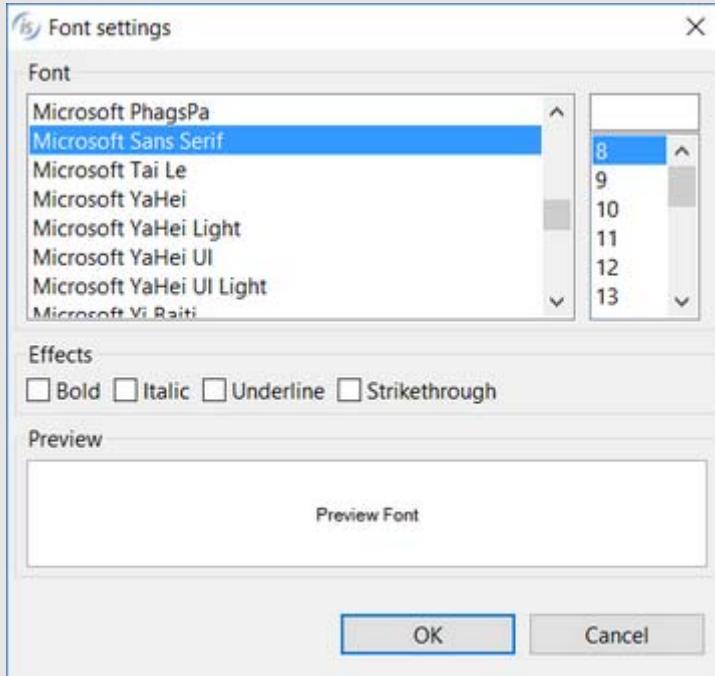
Properties

(name) Specifies the control name. This property is set automatically when the control is drawn.

color Opens a dialog that allows the user to choose the color.

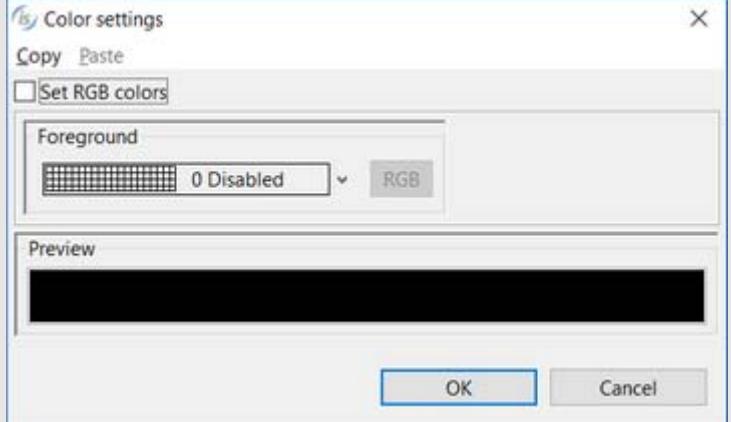
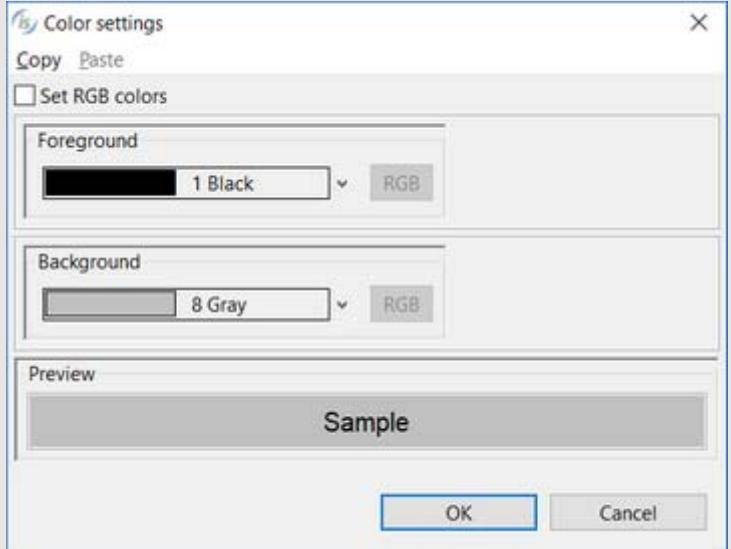


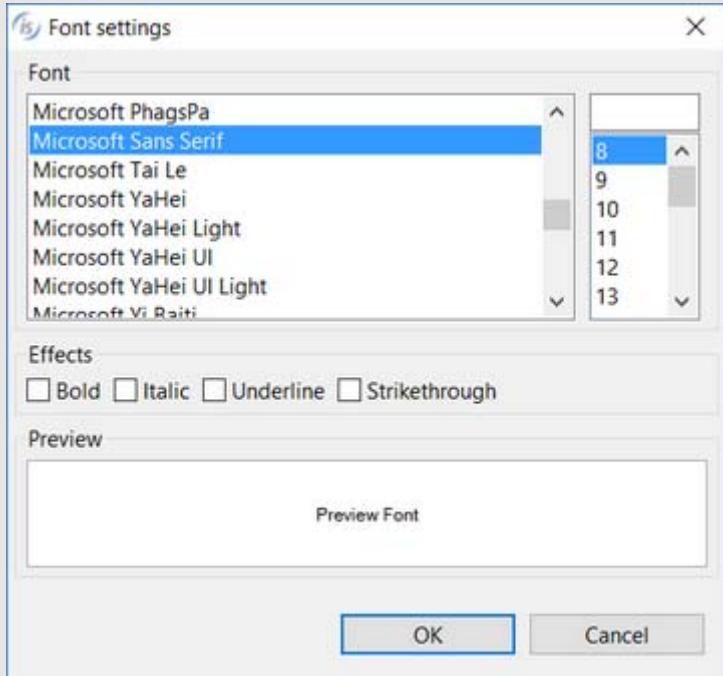
The image shows a 'Color settings' dialog box. At the top, there are 'Copy' and 'Paste' buttons and a checkbox for 'Set RGB colors'. Below this, the 'Foreground' section shows a color swatch set to 'Black' and an 'RGB' button. The 'Background' section shows a color swatch set to 'Gray' and an 'RGB' button. At the bottom, there is a 'Preview' section with a 'Sample' label and a small preview window. At the very bottom are 'OK' and 'Cancel' buttons.

font	Opens a dialog that allows the user to choose the font.
	
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print after page footer	TRUE...The report header prints after the page footer on the last page of the report. FALSE...The report header prints before the page footer on the last page of the report.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
size	Specifies the width of the report item.
skip after print	TRUE...Prints a blank page after this report section FALSE...Doesn't print a blank page after this report section
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	No Events available.
Exceptions	No Exceptions available.

Procedures	
AfterPrint	Allows the user to create a paragraph that is performed after the report item has been printed.
BeforePrint	Allows the user to create a paragraph that is performed before printing the report item.
Variables	
color variable	Numeric variable that hosts the color value.
visible variable	Numeric variable that hosts the visible state.

BOX

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn.
border color	Opens a dialog that allows the user to choose the border color. 
border style	BOXED...The border is shown NO-BOX...The border is not shown
border width	Specifies the width of the border.
color	Opens a dialog that allows the user to choose the color. 
column	Specifies the X coordinate of the report item.

font	Opens a dialog that allows the user to choose the font.
	
line	Specifies the Y coordinate of the report item.
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
size	Specifies the width of the report item.
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	
No Events available.	
Exceptions	
No Exceptions available.	
Procedures	
AfterPrint	Allows the user to create a paragraph that is performed after the report item has been printed.

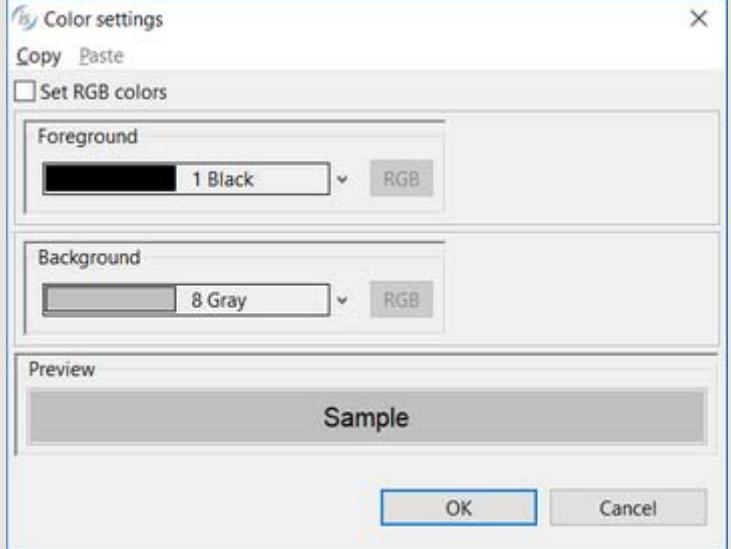
BeforePrint	Allows the user to create a paragraph that is performed before printing the report item.
-------------	--

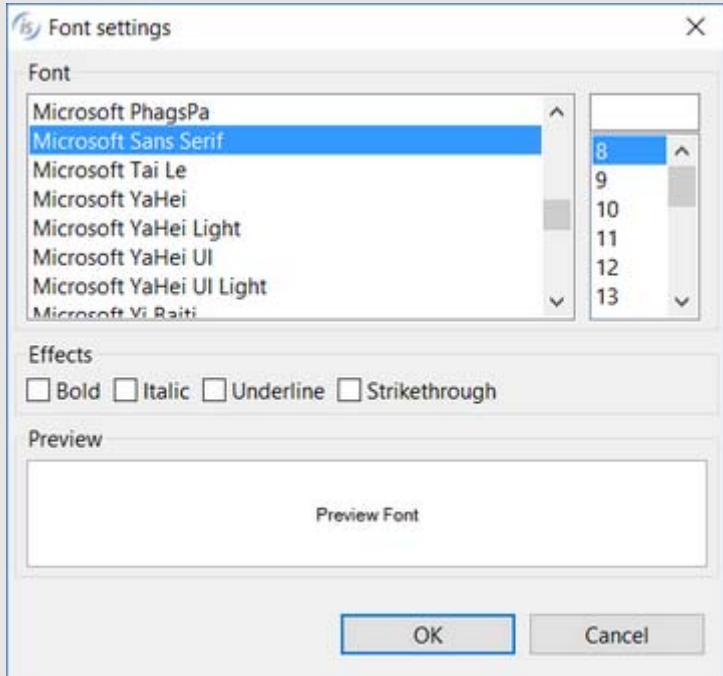
Variables

color variable	Numeric variable that hosts the color value.
----------------	--

visible variable	Numeric variable that hosts the visible state.
------------------	--

CHECK BOX

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn.
color	Opens a dialog that allows the user to choose the color.  A screenshot of the 'Color settings' dialog box. It has tabs for 'Foreground' and 'Background'. Under 'Foreground', a color swatch shows 'Black' and a button says 'RGB'. Under 'Background', a color swatch shows 'Gray' and a button says 'RGB'. Below these are 'Copy' and 'Paste' buttons and a checkbox 'Set RGB colors'. A preview window at the bottom shows a gray bar labeled 'Sample'. At the bottom right are 'OK' and 'Cancel' buttons.
column	Specifies the X coordinate of the report item.

font	Opens a dialog that allows the user to choose the font.
	
line	Specifies the Y coordinate of the report item.
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
size	Specifies the width of the report item.
title	Specifies the text shown on the report item.
value	0...no check mark visible >0... check mark visible
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	No Events available.
Exceptions	No Exceptions available.

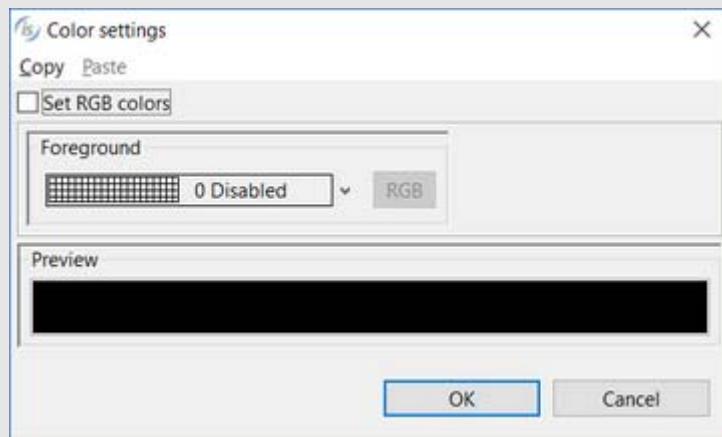
Procedures	
AfterPrint	Allows the user to create a paragraph that is performed after the report item has been printed.
BeforePrint	Allows the user to create a paragraph that is performed before printing the report item.
Variables	
color variable	Numeric variable that hosts the color value.
title variable	Alphanumeric variable that hosts the title.
value variable	Numeric variable that hosts the value.
visible variable	Numeric variable that hosts the visible state.

DATE TIME

Properties

(name) Specifies the control name. This property is set automatically when the control is drawn.

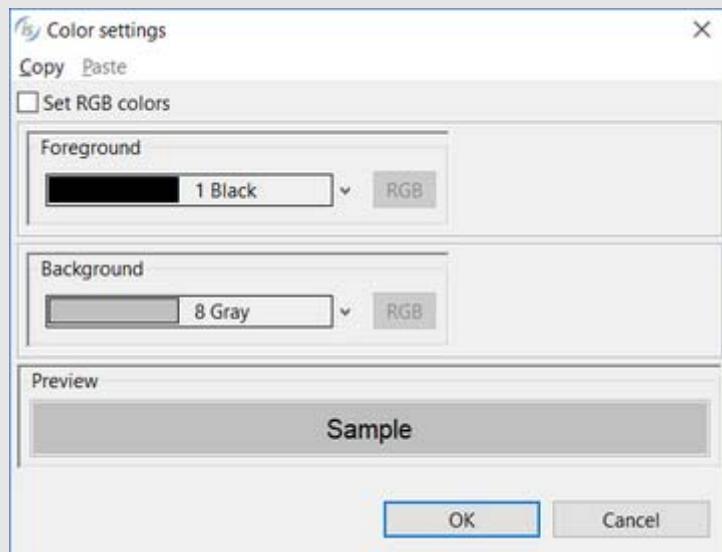
border color Opens a dialog that allows the user to choose the border color.



border style BOXED...The border is shown
NO-BOX...The border is not shown

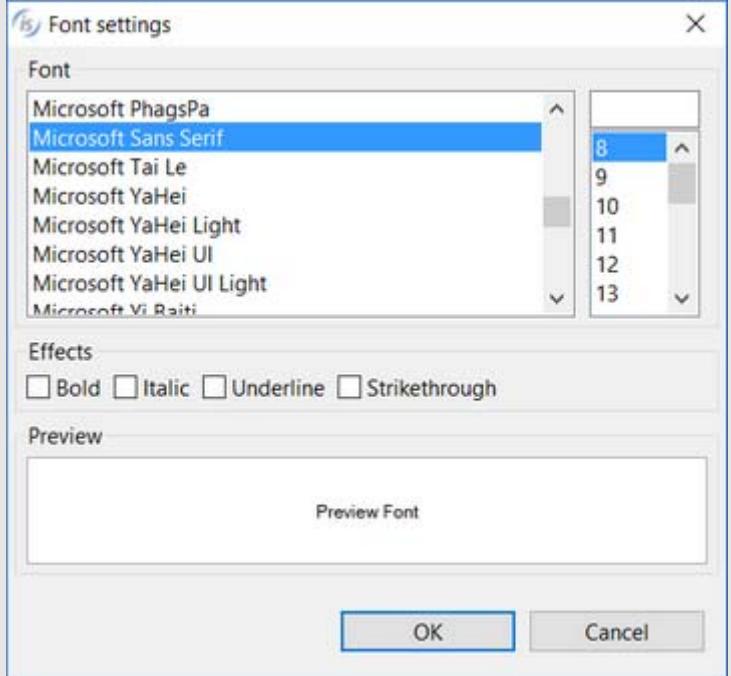
border width Specifies the width of the border.

color Opens a dialog that allows the user to choose the color.



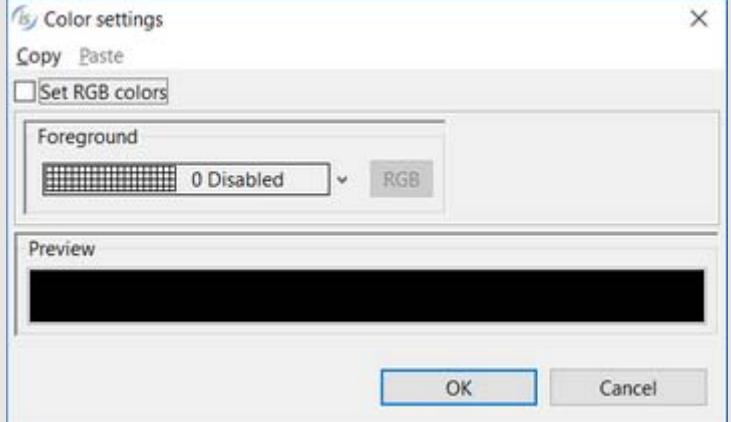
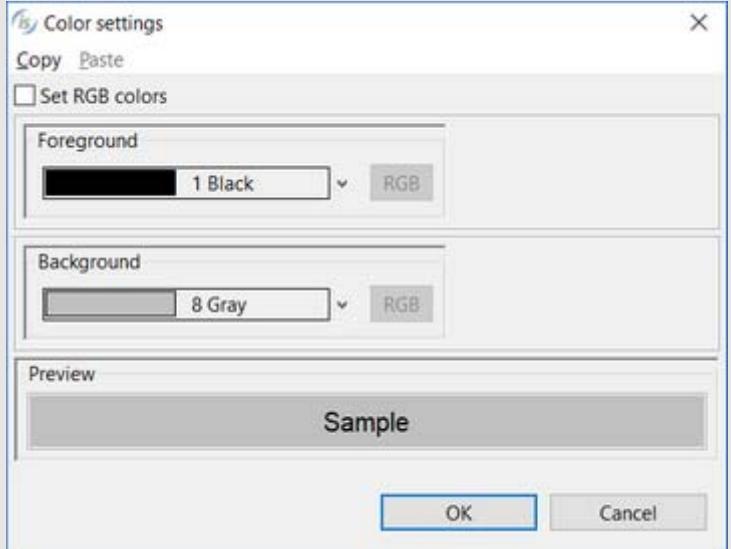
column Specifies the X coordinate of the report item.

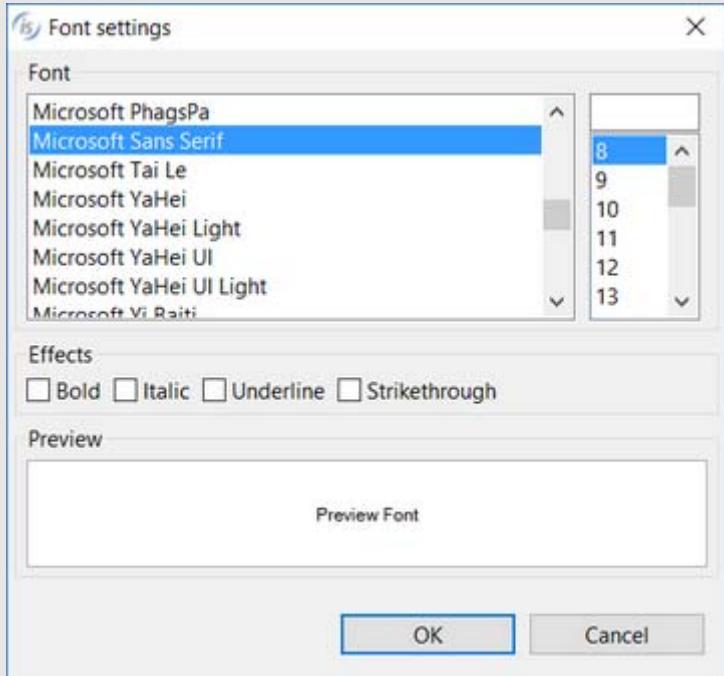
date format Specifies how the date is shown on the Report

date picture format	Specifies how the date is stored in the program Data Division
font	Opens a dialog that allows the user to choose the font.
	
hyperlink	Specifies a URL to navigate if the control is clicked when displayed in a web browser.
justification	Center Left Right Unaligned
line	Specifies the Y coordinate of the report item.
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
print if repeat	TRUE...When consecutive records contain the same data values, both data values print. FALSE...When consecutive records contain the same data values, the second (and subsequent) same data values do not print.
size	Specifies the width of the report item.
time format	Specifies how the time is shown on the Report

time picture format	Specifies how the time is stored in the program Data Division
value picture	Specifies the picture for the value variable
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	
No Events available.	
Exceptions	
No Exceptions available.	
Procedures	
AfterPrint	Allows the user to create a paragraph that is performed after the report item has been printed.
BeforePrint	Allows the user to create a paragraph that is performed before printing the report item.
Variables	
color variable	Numeric variable that hosts the color value.
hyperlink variable	Alphanumeric variable that hosts the hyperlink.
value variable	Numeric variable that hosts the value.
visible variable	Numeric variable that hosts the visible state.

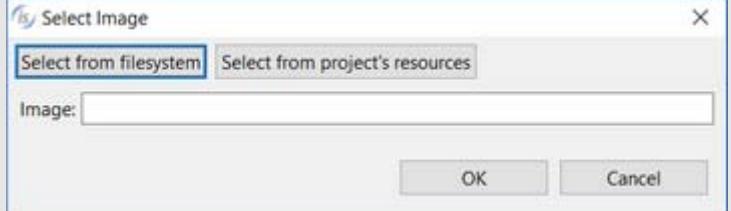
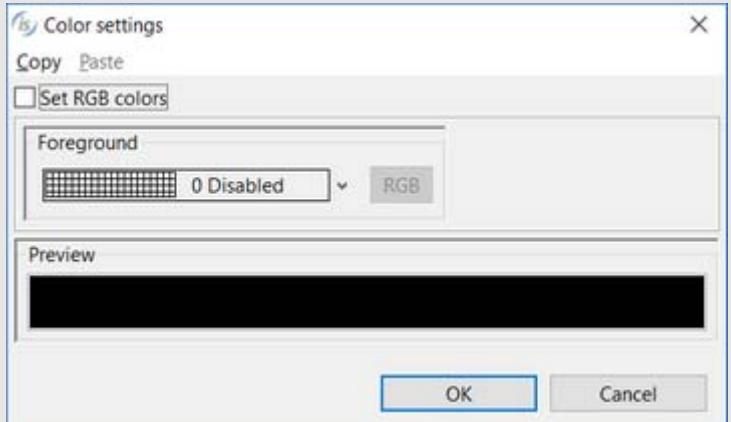
ENTRY FIELD

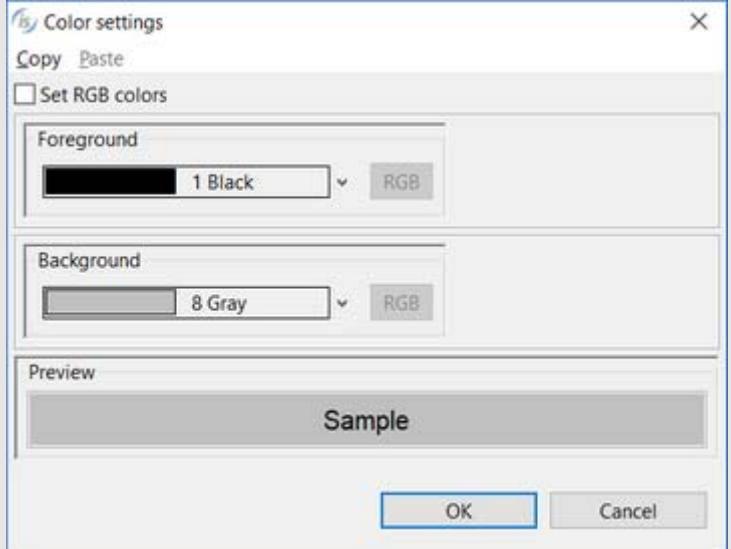
Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn.
border color	Opens a dialog that allows the user to choose the border color. 
border style	BOXED...The border is shown NO-BOX...The border is not shown
border width	Specifies the width of the border.
color	Opens a dialog that allows the user to choose the color. 
column	Specifies the X coordinate of the report item.

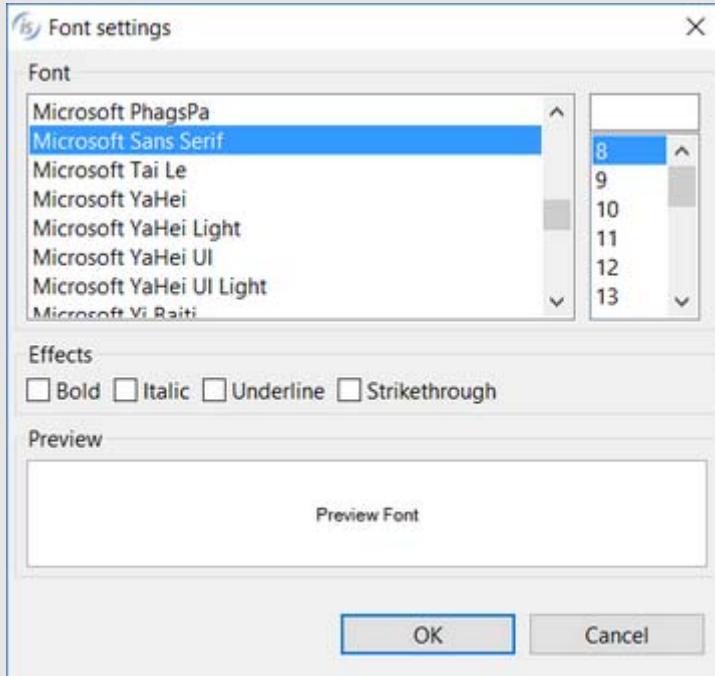
display type	COLLAPSE...only one space between two strings allowed KEEP SPACES....any amount of space between two strings is kept. PREFORMATTED...any format settings are kept. <PRE> and </PRE> HTML tags are generated to maintain the text format.
font	Opens a dialog that allows the user to choose the font. 
horizontal spacing	Specifies the space between characters printed inside the control.
hyperlink	Specifies a URL to navigate if the control is clicked when displayed in a web browser.
justification	Center Left Right Unaligned
line	Specifies the Y coordinate of the report item.
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
print if repeat	TRUE...When consecutive records contain the same data values, both data values print. FALSE...When consecutive records contain the same data values, the second (and subsequent) same data values do not print.

size	Specifies the width of the report item.
value picture	Specifies the picture for the value variable
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	
No Events available.	
Exceptions	
No Exceptions available.	
Procedures	
AfterPrint	Allows the user to create a paragraph that is performed after the report item has been printed.
BeforePrint	Allows the user to create a paragraph that is performed before printing the report item.
Variables	
color variable	Numeric variable that hosts the color value.
hyperlink variable	Alphanumeric variable that hosts the hyperlink.
value variable	Numeric variable that hosts the value.
visible variable	Numeric variable that hosts the visible state.

IMAGE

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn.
bitmap	Opens a dialog that allows to choose the bitmap.  A screenshot of the "Select Image" dialog box. It has tabs for "Select from filesystem" and "Select from project's resources". Below them is a "Image:" input field. At the bottom are "OK" and "Cancel" buttons.
bitmap path	FULL PATH...The browser uses the full path of the bitmap to locate the bitmap file. DYNAMIC FULL PATH...A call to the C\$FULLNAME library routine is used to derive the full path of the bitmap file. The bitmap can be stored in any of the FILE-PREFIX directories named. The browser uses the full path of the bitmap to locate the bitmap file. USER DEFINED...The browser searches for the bitmap in the same directory as the HTML file.
bitmap position	Specifies the position of the bitmap within the control area: -Center -Left Top -Left Bottom -Right Top -Right Bottom
bitmap style	RATIO...The bitmap's height-to-width ratio is preserved inside the space allocated for the bitmap. STRETCH...The bitmap height-to-width ratio is altered if necessary to fit the bitmap into the space allocated for it.
border color	Opens a dialog that allows the user to choose the border color.  A screenshot of the "Color settings" dialog box. It includes a "Copy" and "Paste" button, a checkbox for "Set RGB colors", and a "Foreground" color picker showing "0 Disabled" and an "RGB" button. Below is a "Preview" window showing a black rectangle. At the bottom are "OK" and "Cancel" buttons.

border style	BOXED...The border is shown NO-BOX...The border is not shown
border width	Specifies the width of the border.
color	Opens a dialog that allows the user to choose the color.
	 <p>The screenshot shows the 'Color settings' dialog box. It has sections for 'Foreground' and 'Background'. Under 'Foreground', the color is set to '1 Black'. Under 'Background', the color is set to '8 Gray'. There is a 'Preview' section at the bottom with a 'Sample' button. At the bottom right are 'OK' and 'Cancel' buttons.</p>
column	Specifies the X coordinate of the report item.

font	Opens a dialog that allows the user to choose the font.
	
hyperlink	Specifies a URL to navigate if the control is clicked when displayed in a web browser.
line	Specifies the Y coordinate of the report item.
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
print if repeat	TRUE...When consecutive records contain the same data values, both data values print. FALSE...When consecutive records contain the same data values, the second (and subsequent) same data values do not print.
size	Specifies the width of the report item.
value	Specifies the control value
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	No Events available.

Exceptions

No Exceptions available.

Procedures

AfterPrint Allows the user to create a paragraph that is performed after the report item has been printed.

BeforePrint Allows the user to create a paragraph that is performed before printing the report item.

Variables

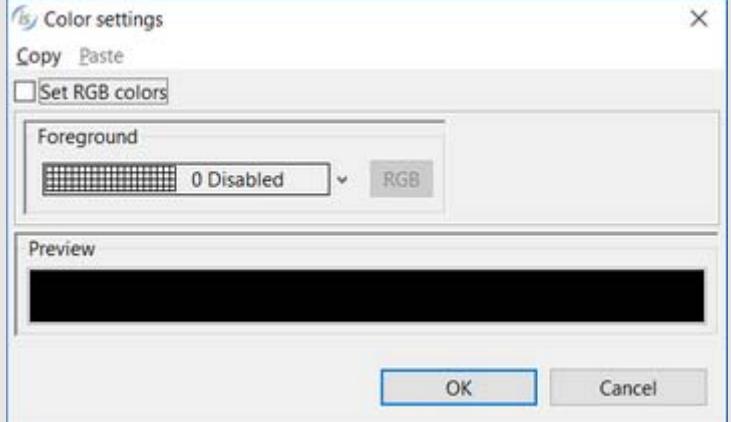
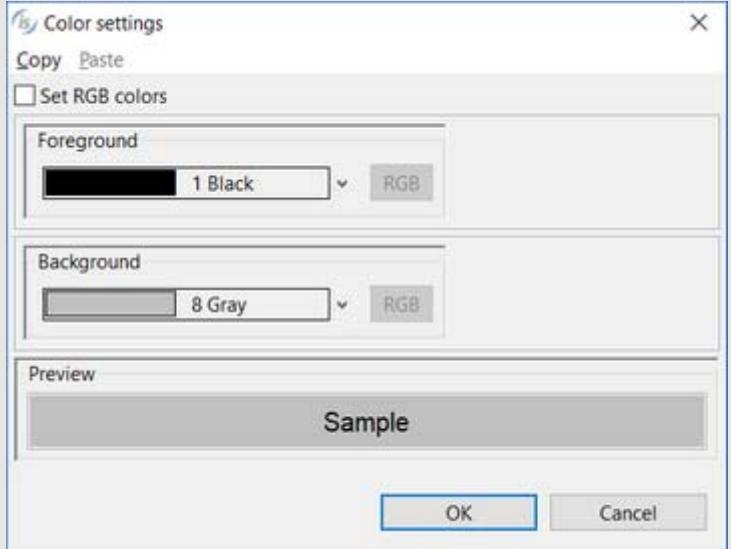
color variable Numeric variable that hosts the color value.

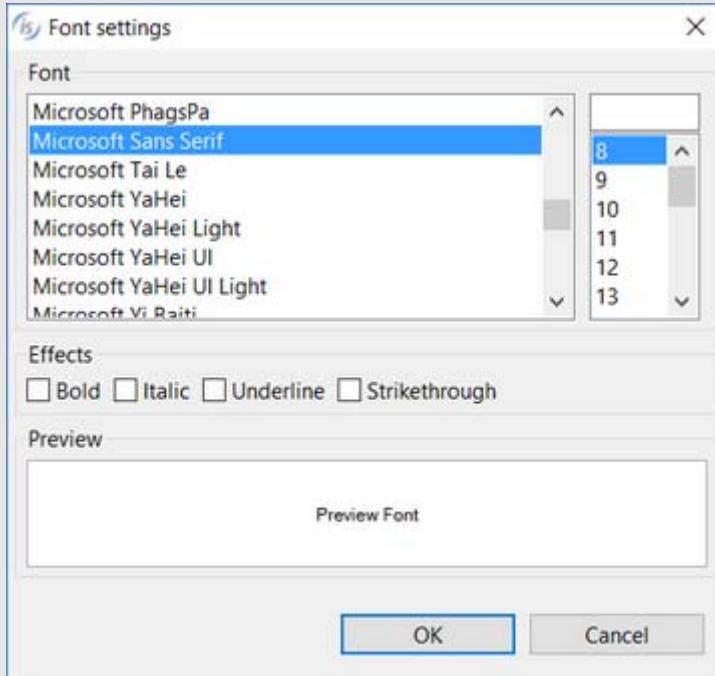
hyperlink variable Alphanumeric variable that hosts the hyperlink.

value variable Numeric variable that hosts the value.

visible variable Numeric variable that hosts the visible state.

LABEL

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn.
border color	Opens a dialog that allows the user to choose the border color. 
border style	BOXED...The border is shown NO-BOX...The border is not shown
border width	Specifies the width of the border.
color	Opens a dialog that allows the user to choose the color. 
column	Specifies the X coordinate of the report item.

font	Opens a dialog that allows the user to choose the font.
	
hyperlink	Specifies a URL to navigate if the control is clicked when displayed in a web browser.
justification	Center Left Right Unaligned
line	Specifies the Y coordinate of the report item.
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
size	Specifies the width of the report item.
title	Specifies the text shown on the report item
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	No Events available.

Exceptions

No Exceptions available.

Procedures

AfterPrint Allows the user to create a paragraph that is performed after the report item has been printed.

BeforePrint Allows the user to create a paragraph that is performed before printing the report item.

Variables

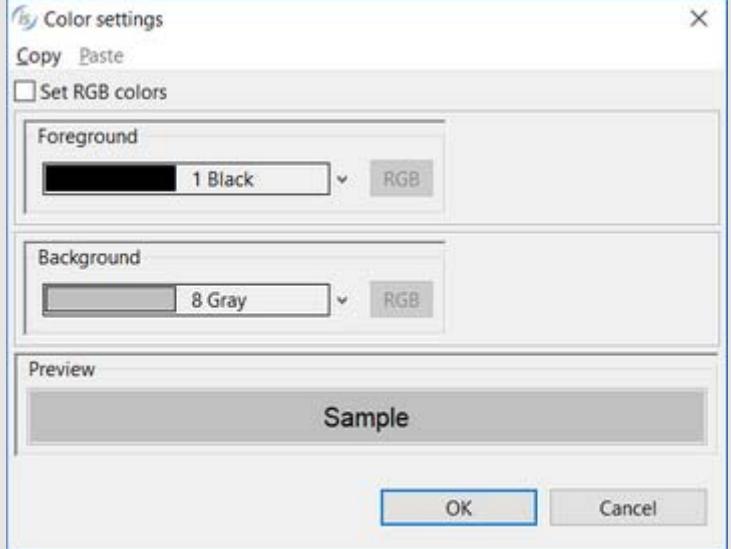
color variable Numeric variable that hosts the color value.

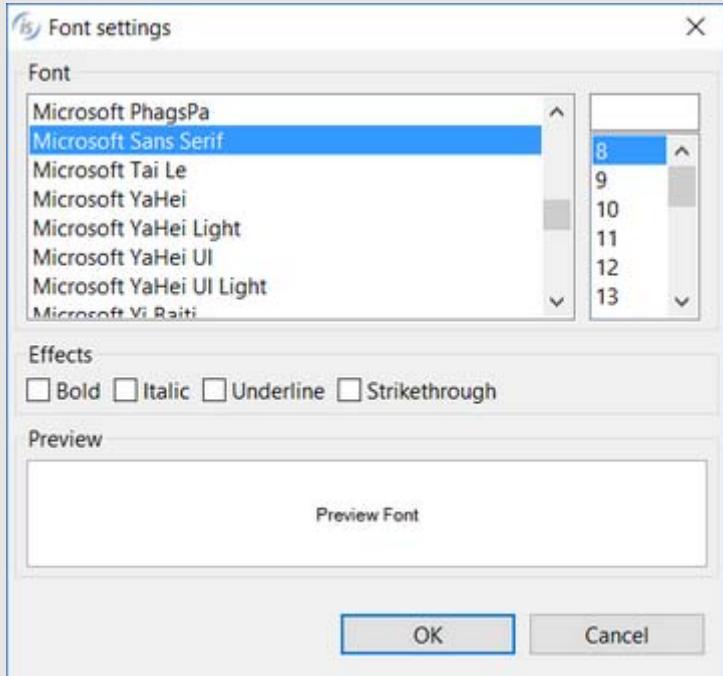
hyperlink variable Alphanumeric variable that hosts the hyperlink.

title variable Numeric variable that hosts the title.

visible variable Numeric variable that hosts the visible state.

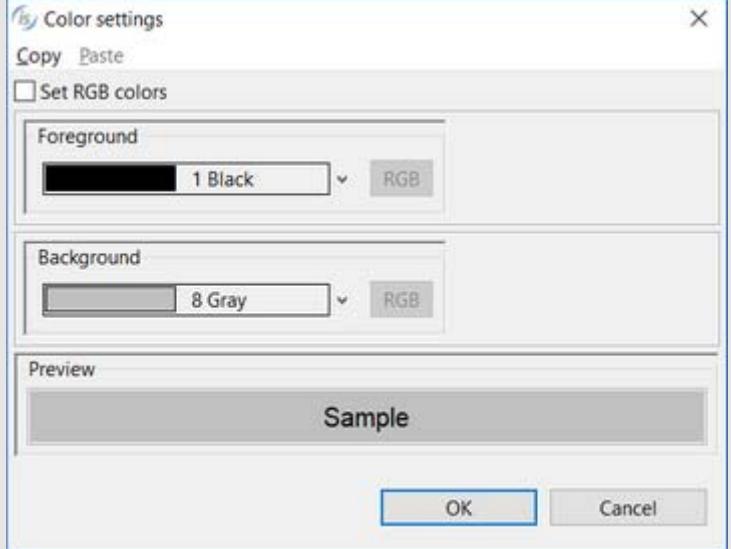
LINE

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn.
color	Opens a dialog that allows the user to choose the color.  A screenshot of the 'Color settings' dialog box. It has sections for 'Foreground' and 'Background'. Under 'Foreground', a color swatch shows 'Black' and a button says 'RGB'. Under 'Background', a color swatch shows 'Gray' and a button says 'RGB'. Below these is a 'Preview' section with a 'Sample' button. At the bottom are 'OK' and 'Cancel' buttons.
column	Specifies the X coordinate of the report item.

font	Opens a dialog that allows the user to choose the font.
	
kind	Horizontal Vertical
line	Specifies the Y coordinate of the report item.
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.
print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
size	Specifies the width of the report item.
visible	TRUE... The report item is visible FALSE... The report item is hidden
width	Specifies the thickness
Events	No Events available.
Exceptions	No Exceptions available.

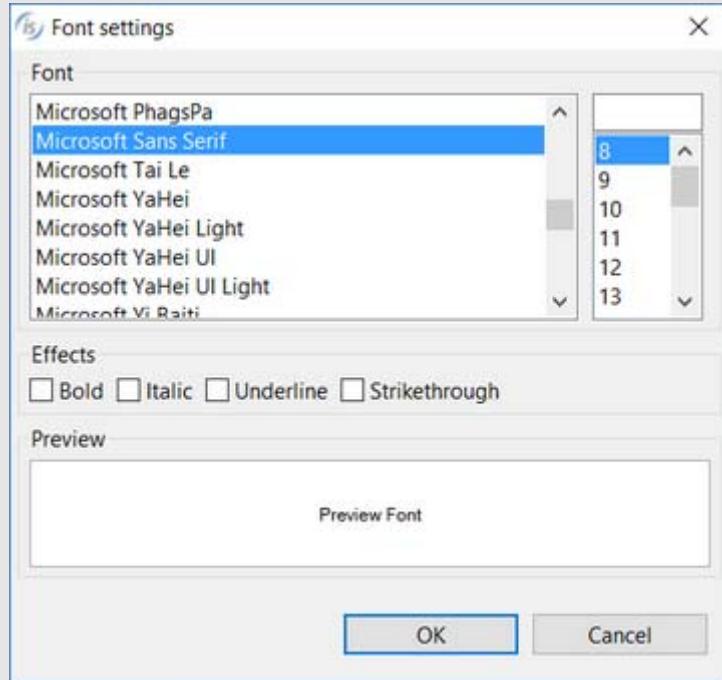
Procedures	
AfterPrint	Allows the user to create a paragraph that is performed after the report item has been printed.
BeforePrint	Allows the user to create a paragraph that is performed before printing the report item.
Variables	
color variable	Numeric variable that hosts the color value.
visible variable	Numeric variable that hosts the visible state.

RADIO BUTTON

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn.
color	Opens a dialog that allows the user to choose the color.  A screenshot of the 'Color settings' dialog box. It has sections for 'Foreground' and 'Background'. Under 'Foreground', a color swatch shows 'Black' and a button labeled 'RGB'. Under 'Background', a color swatch shows 'Gray' and a button labeled 'RGB'. Below these is a 'Preview' section with a 'Sample' button. At the bottom are 'OK' and 'Cancel' buttons.
column	Specifies the X coordinate of the report item.

font

Opens a dialog that allows the user to choose the font.



group

This property has not effect, it's included for user documentation purposes.

line

Specifies the Y coordinate of the report item.

lines

Specifies the width of the report item.

lock

TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse.
FALSE...You can move the control on the Report Designer by dragging it with the mouse.

print condition

Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.

size

Specifies the width of the report item.

title

Specifies the text shown on the report item

value

0...no check mark visible
>0... check mark visible

value picture

Specifies the picture for the value variable.

visible

TRUE... The report item is visible
FALSE... The report item is hidden

Events

No Events available.

Exceptions

No Exceptions available.

Procedures

AfterPrint Allows the user to create a paragraph that is performed after the report item has been printed.

BeforePrint Allows the user to create a paragraph that is performed before printing the report item.

Variables

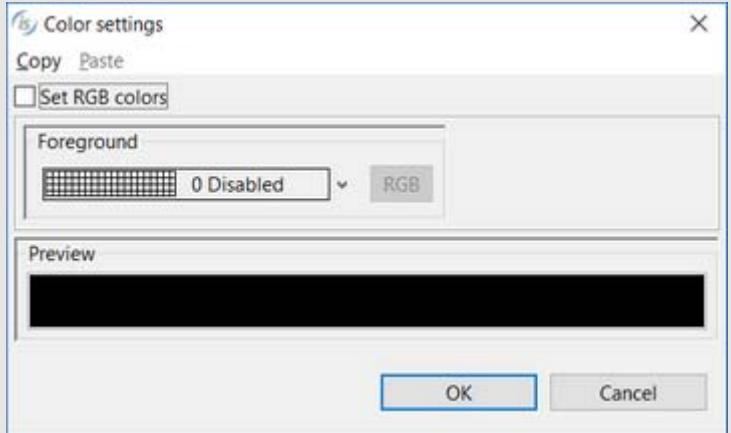
color variable Numeric variable that hosts the color value.

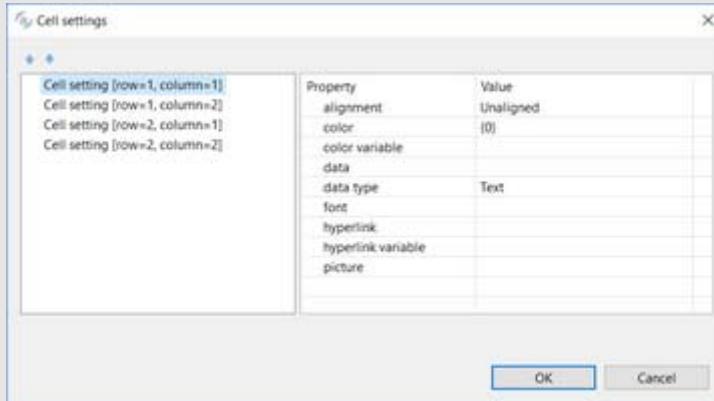
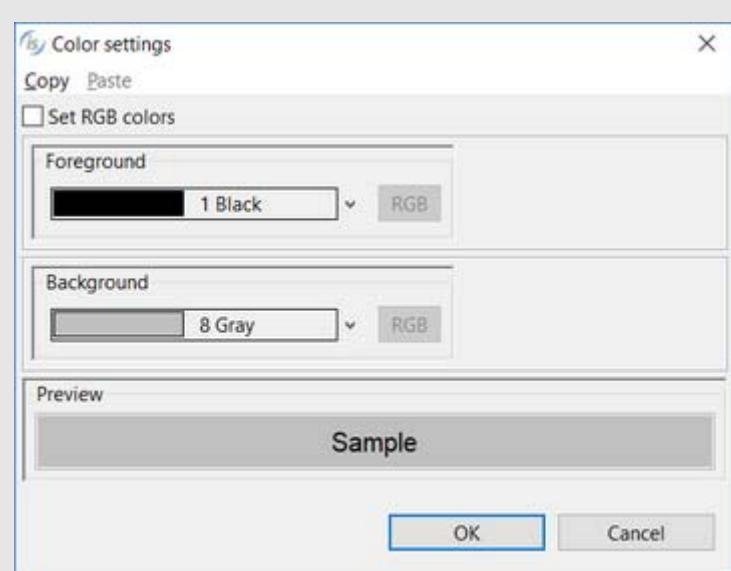
title variable Numeric variable that hosts the title.

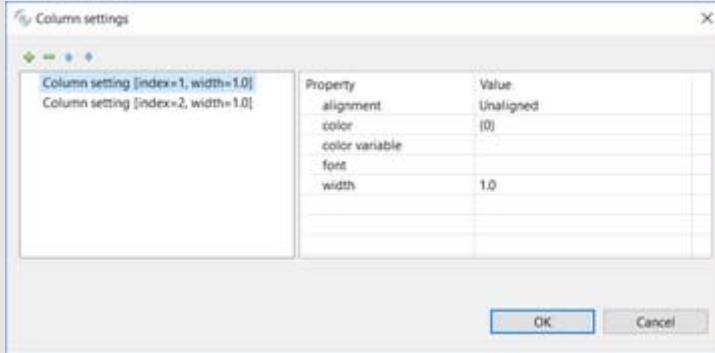
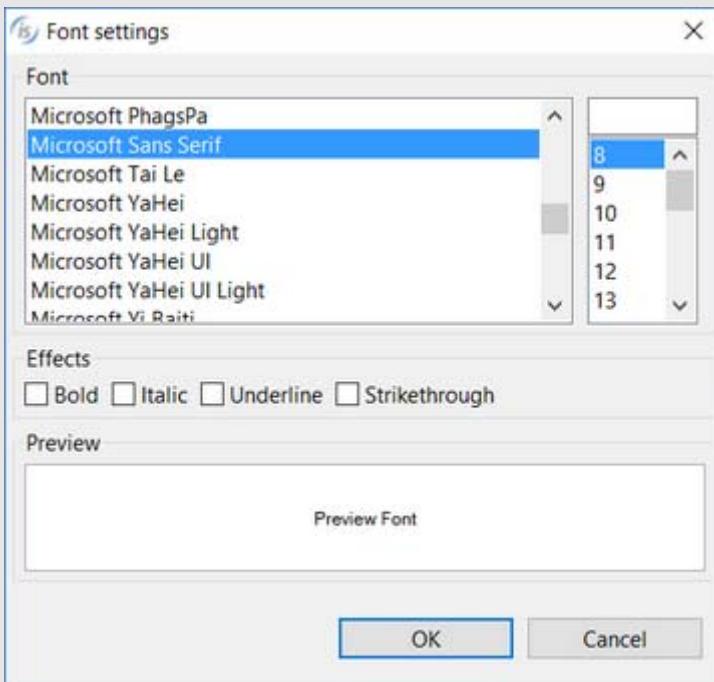
value variable Numeric variable that hosts the value.

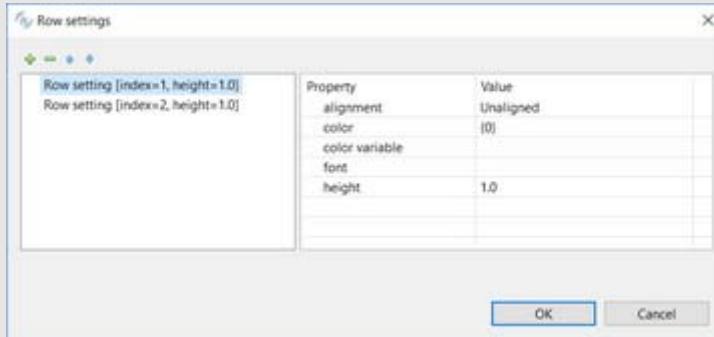
visible variable Numeric variable that hosts the visible state.

TABLE

Properties	
(name)	Specifies the control name. This property is set automatically when the control is drawn.
bitmap path	FULL PATH...The browser uses the full path of the bitmap to locate the bitmap file. DYNAMIC FULL PATH...A call to the C\$FULLNAME library routine is used to derive the full path of the bitmap file. The bitmap can be stored in any of the FILE-PREFIX directories named. The browser uses the full path of the bitmap to locate the bitmap file. USER DEFINED...The browser searches for the bitmap in the same directory as the HTML file. It applies to all the images shown in the table.
border color	Opens a dialog that allows the user to choose the border color.  A screenshot of the 'Color settings' dialog box. It has tabs for 'Copy' and 'Paste'. A checkbox labeled 'Set RGB colors' is checked. Under 'Foreground', there is a color picker with a grid preview, a text input '0 Disabled', and an 'RGB' button. Below it is a 'Preview' section with a large black rectangular area. At the bottom are 'OK' and 'Cancel' buttons.
border style	BOXED...The border is shown NO-BOX...The border is not shown
border width	Specifies the width of the border.
cell padding	Specifies the amount of blank space in pixel between the text in a cell and the horizontal grid lines above and below that text.

cell settings	Opens a dialog that allows to configure cell settings
	 <p>The screenshot shows the 'Cell settings' dialog box. On the left, there is a list of items: 'Cell setting [row=1, column=1]', 'Cell setting [row=1, column=2]', 'Cell setting [row=2, column=1]', and 'Cell setting [row=2, column=2]'. To the right of each item is a table with two columns: 'Property' and 'Value'. The properties listed are alignment, color, color variable, data, data type, font, hyperlink, hyperlink variable, and picture. The corresponding values are Unaligned, (0), , Text, , , , and . At the bottom right of the dialog are 'OK' and 'Cancel' buttons.</p>
cell spacing	Specifies the amount of blank space in pixels between the vertical grid lines and horizontal grid lines in a table cell.
color	Opens a dialog that allows the user to choose the color.
	 <p>The screenshot shows the 'Color settings' dialog box. It includes sections for 'Foreground' (set to Black) and 'Background' (set to Gray). Below these are 'Copy' and 'Paste' buttons and a checkbox for 'Set RGB colors'. A 'Preview' section at the bottom shows a 'Sample' color bar. At the bottom right are 'OK' and 'Cancel' buttons.</p>
column	Specifies the X coordinate of the report item.

column settings	Opens a dialog that allows to configure column settings.
	
font	Opens a dialog that allows the user to choose the font.
	
group	This property has not effect, it's included for user documentation purposes.
line	Specifies the Y coordinate of the report item.
lines	Specifies the width of the report item.
lock	TRUE...Locks the control on the Report Designer so that you cannot move it anymore by dragging it with the mouse. FALSE...You can move the control on the Report Designer by dragging it with the mouse.

print condition	Specifies a condition (e.g. WRK-USER="Admin") that avoids the Report item to be printed when false.
row settings	Opens a dialog that allows to configure row settings. 
show grid line	TRUE...Cells dividers are shown FALSE...Cells dividers are not shown
size	Specifies the width of the report item.
visible	TRUE... The report item is visible FALSE... The report item is hidden
Events	
No Events available.	
Exceptions	
No Exceptions available.	
Procedures	
AfterPrint	Allows the user to create a paragraph that is performed after the report item has been printed.
BeforePrint	Allows the user to create a paragraph that is performed before printing the report item.
Variables	
color variable	Numeric variable that hosts the color value.
visible variable	Numeric variable that hosts the visible state.

How to print Reports

Retrieving data for the Report

The way you retrieve data for the Report depends if you rely on Tagged Areas or not in your code.

For programs that don't use Tagged Areas (default), follow these steps:

- write COBOL statements that allow you to retrieve the first data record in the BeforeDoPrint procedure of your Report.

Code example:

```
report-1-bef-do-print.  
    move low-value to prd-key  
    start product key not < prd-key  
        invalid  
            move 0 to report-1-doprintrtn-loop  
        not invalid  
            move 1 to report-1-doprintrtn-loop  
            read product next no lock  
                at end  
                    move 0 to report-1-doprintrtn-loop  
                end-read  
            end-start  
.  
.
```

<ReportName>-doprintrtn-loop is a variable declared automatically by the IDE. When you set this variable to 0 the print loop is interrupted, when you set it to 1 the print loop continues.

- write COBOL statements that allow you to retrieve the next record for the Report in the AfterDoPrint procedure of your Report.

Code example:

```
report-1-aft-do-print.  
    move 1 to report-1-doprintrtn-loop  
    read product next no lock  
        at end  
            move 0 to report-1-doprintrtn-loop  
        end-read  
.  
.
```

For programs that use Tagged Areas, follow these steps:

- write COBOL statements that allow you to retrieve data for the Report in the paragraph "is-<reportname>-master-print-loop", that is always generated when a Report is present. Write your statements outside of the tagged area in that paragraph

Code example:

```
move low-value to prd-key  
start product key not < prd-key  
    invalid  
        continue  
    not invalid  
        perform until 1 = 2  
            read product next  
            at end  
                exit perform  
        end-read  
        perform IS-REPORT-1-DO-PRINT-RTN  
    end-perform  
end-start.  
.
```

Note that for each record read in the above loop, the paragraph IS-<ReportName>-DO-PRINT-RTN is performed. Such paragraph is responsible for printing the data.

Printing the Report

The following paragraphs are always generated for programs with reports. You can perform them in your program logic. Choose the proper one according to the desired printer output:

Paragraph Name	Action
is-report-name-preview	<p>Opens the Report in a Print Preview</p> <p>The preview dialog is very similar to the one shown by the SpoolPrinter class (com.iscobol.rts.print.SpoolPrinter) and the Print Preview, but the preview dialog generated by the IDE allows also to export the report to xls and xlsx files.</p>  <p>See Exporting to Excel file below for details.</p>
is-report-name-print	Prints the Report on the active printer
is-report-name-print-pdf	<p>Prints the Report to a PDF file. The name and location of the PDF file are calculated according to the output file name property. If the report name specified by that property didn't have any extension, a ".pdf" extension is appended to it. If the name specified by that property had an extension, such extension is replaced with ".pdf".</p> <p>Use the "@[display]:" prefix followed by a client side path (e.g. "@[display]:C:\Temp\myReport.pdf") to generate the report on the client machine when the program runs in a thin client environment. The "@[display]:" prefix is automatically ignored when running out of a thin client environment.</p>
is-report-name-print-xls	<p>Prints the Report to an Excel workbook in XLS format. The name and location of the XLS file are calculated according to the output file name property. If the report name specified by that property didn't have any extension, a ".xls" extension is appended to it. If the name specified by that property had an extension, such extension is replaced with ".xls".</p> <p>Use the "@[display]:" prefix followed by a client side path (e.g. "@[display]:C:\Temp\myReport.xls") to generate the report on the client machine when the program runs in a thin client environment. The "@[display]:" prefix is automatically ignored when running out of a thin client environment.</p>
is-report-name-print-xlsx	<p>Prints the Report to an Excel workbook in XLSX format. The name and location of the XLSX file are calculated according to the output file name property. If the report name specified by that property didn't have any extension, a ".xlsx" extension is appended to it. If the name specified by that property had an extension, such extension is replaced with ".xlsx".</p> <p>Use the "@[display]:" prefix followed by a client side path (e.g. "@[display]:C:\Temp\myReport.xlsx") to generate the report on the client machine when the program runs in a thin client environment. The "@[display]:" prefix is automatically ignored when running out of a thin client environment.</p>
is-report-name-print-tofile	Prints the Report to an HTML file. The name and location of the HTML file are controlled by the output file name property.

Paragraph Name	Action
is-report-name-setup-print	Shows the Choose Printer dialog, then print the Report on the chosen printer

Exporting to Excel file

The *Export* menu in the print preview dialog allows to export the current report to either a XLS or XLSX file.

The export process can be configured through specific properties. See [Export to Excel feature](#) for details.

There are few limitations if compared with the HTML report:

- Header and footer information like the page number are skipped and don't appear in the Excel file.
- If the report includes more lines than the maximum number of lines allowed by the XLS format, then the report is automatically split into multiple spreadsheets.
- If the report includes more columns than the maximum number of columns allowed by the XLS format, then the export fails, but no error is shown.

Adding a new Report to the program

A program can contain more than one report. To add new reports:

1. Right click on the program name.
2. Select *New / Report* from the pop-up menu.

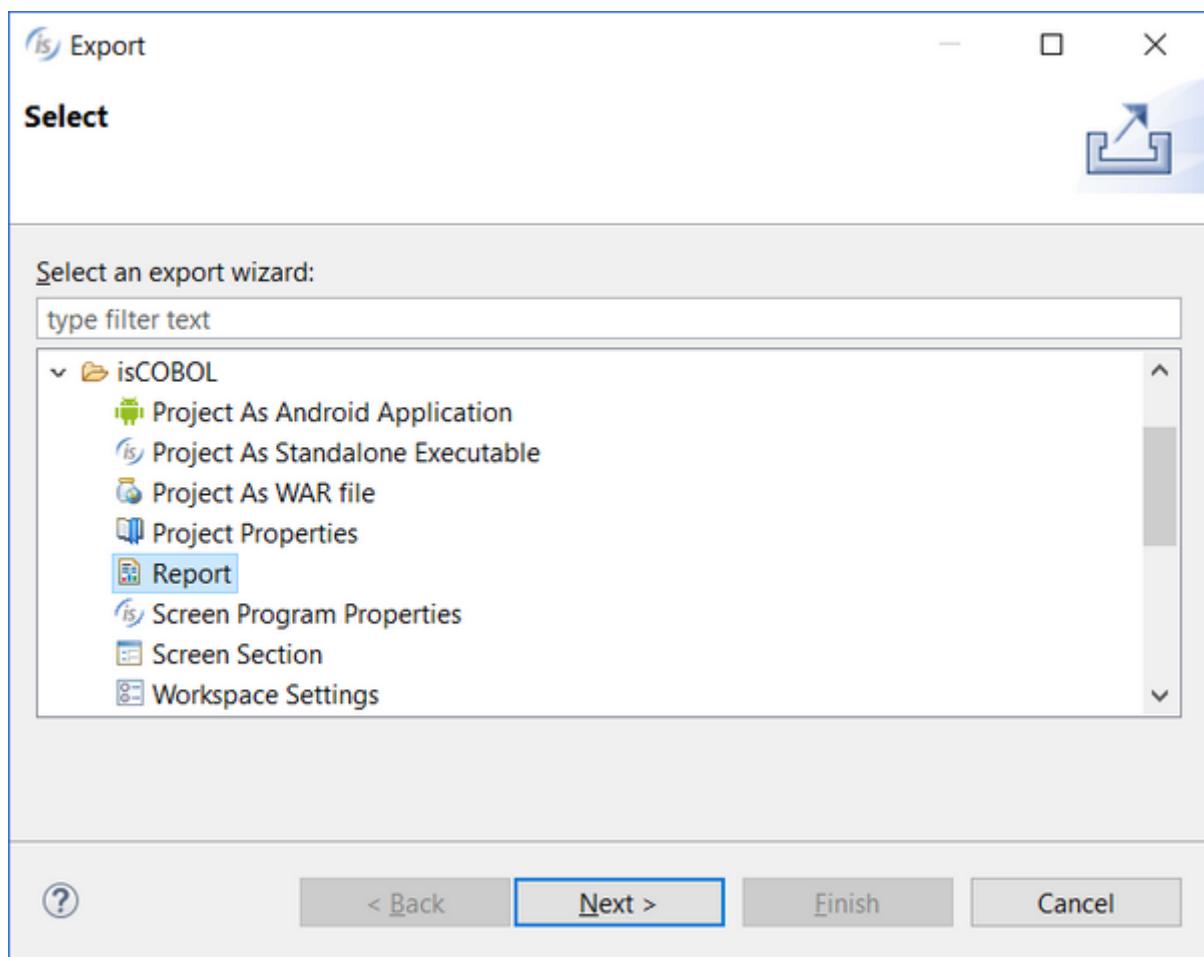
The wizard procedure allows you to choose between creating a blank report or starting from a template. See [Loading Report Templates](#) for more details about templates.

Import / Export of Reports

Reports can be exported to external files and imported in other projects.

To export one or more screens:

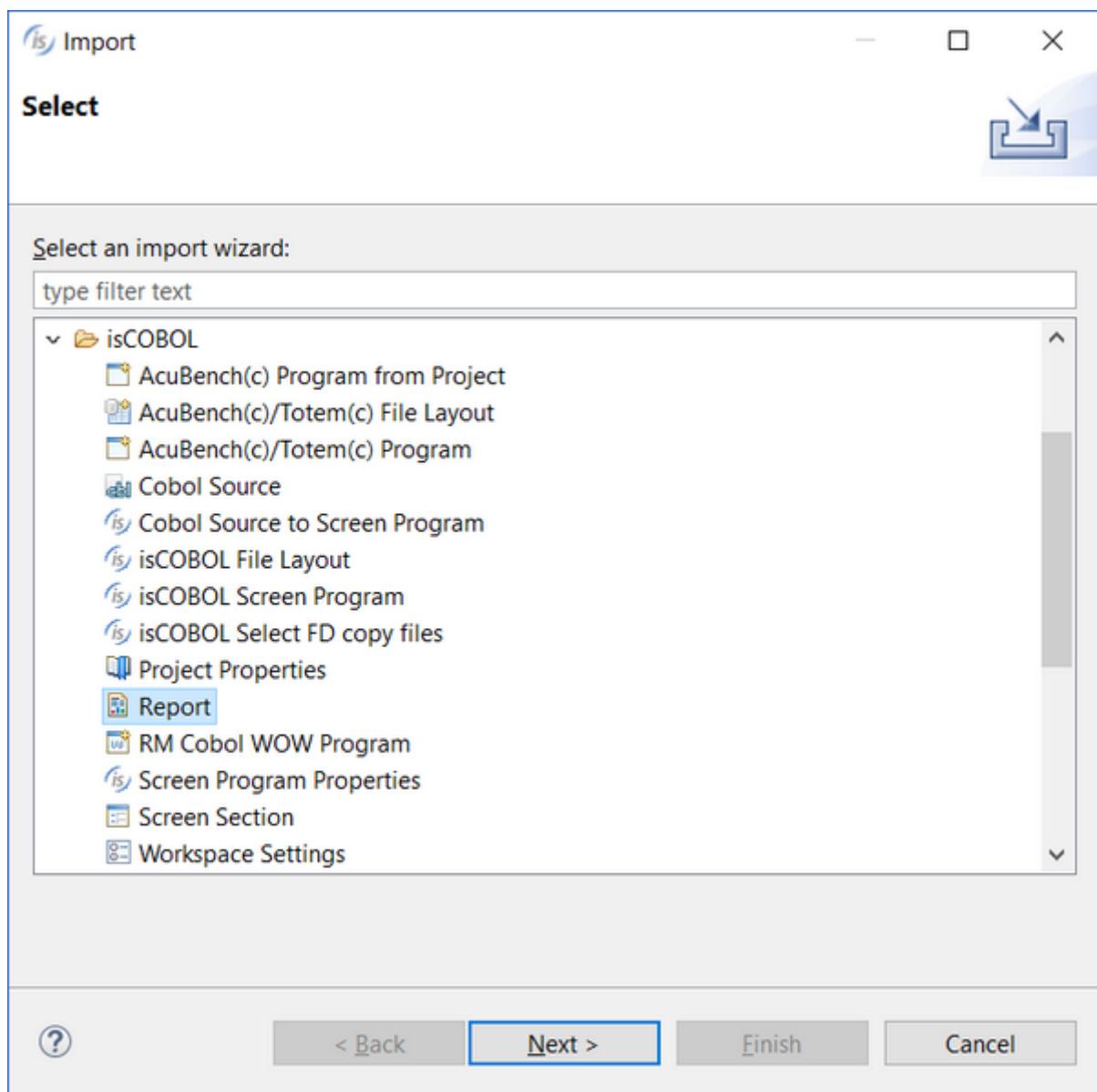
1. right click on the program name in the isCOBOL Explorer
2. choose *Export*
3. expand *isCOBOL*
4. select *Report*



5. click *Next*
6. select the screens you want to export
7. click *Next*
8. choose the destination file (the file must have .irl extension)
9. click *Finish*

To import a irl file and have its report added to your program:

1. right click on the program name in the isCOBOL Explorer
2. choose *Import*
3. expand *isCOBOL*
4. select *Report*



5. click *Next*
6. browse for the directory where .irl files are
7. select the screens you wish to import
8. click *Finish*

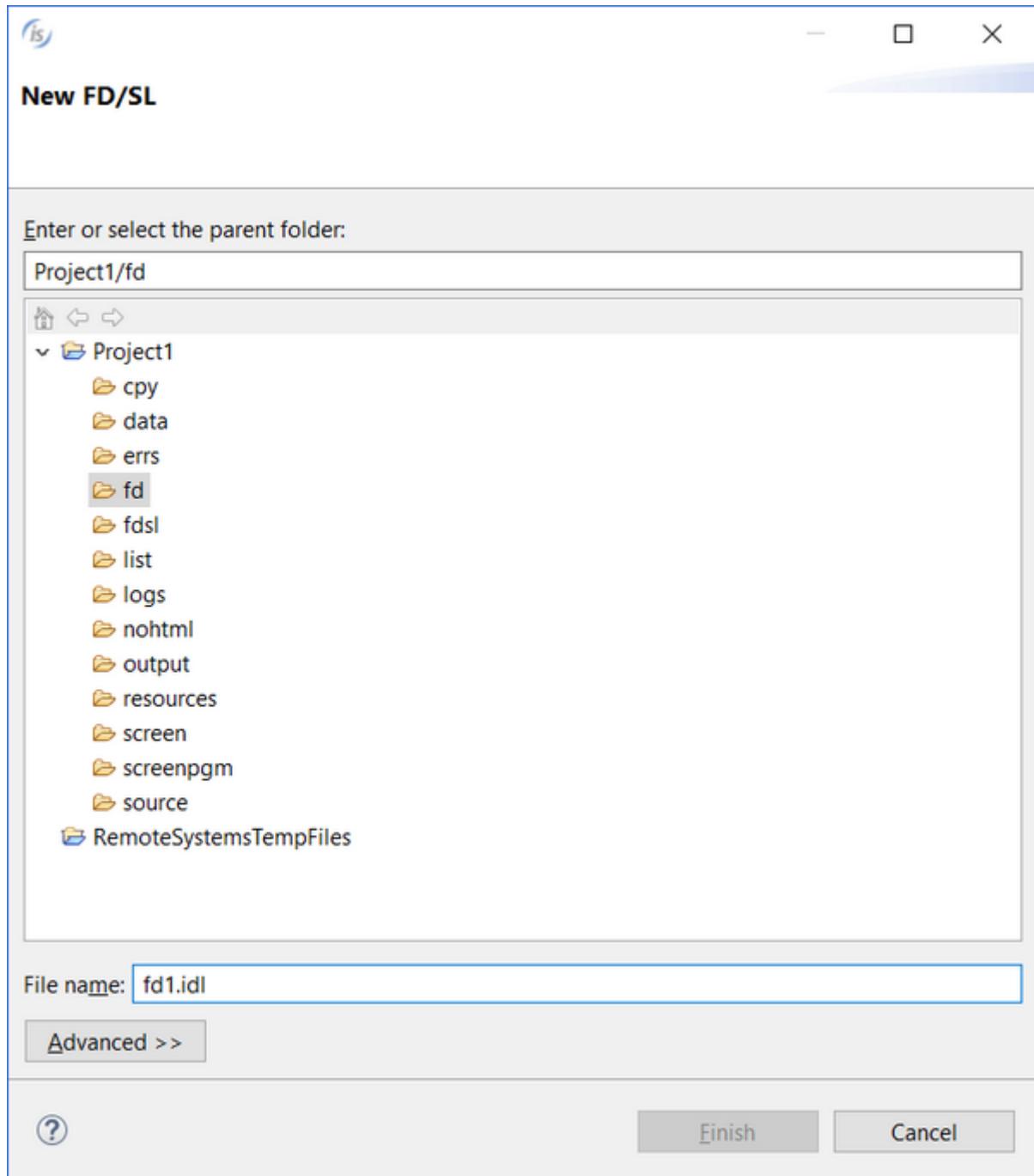
The irl file can also be used as a template for new screens. See [Loading Report Templates](#) for more information.

File Layouts

File Layouts allow to maintain data file structures in the IDE graphical editor.

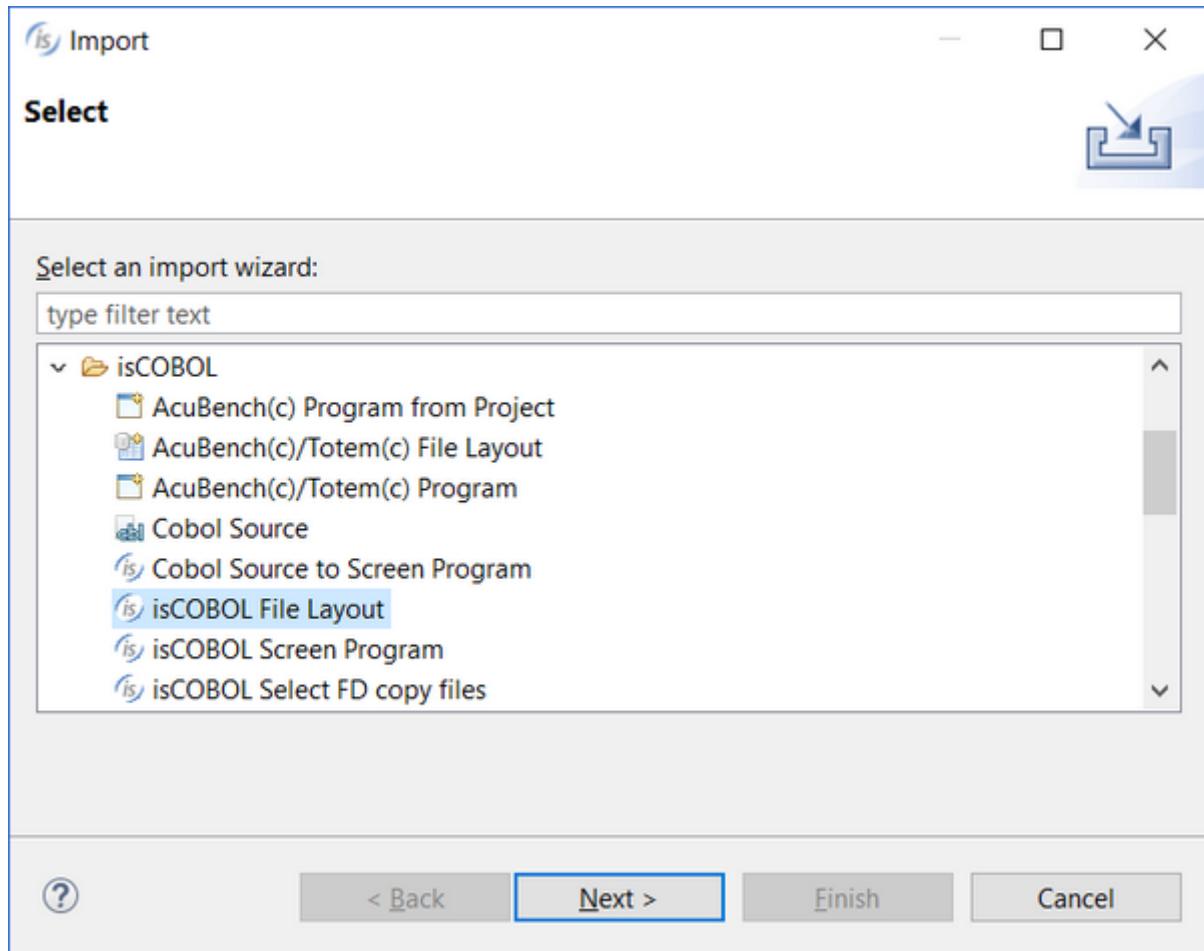
Creating a new File Layout

To create a new File Layout, right click on the project name in the isCOBOL Explorer area and choose *New / FD/SL* from the pop-up menu. The IDE will ask for fd name. Ensure that the name terminates with the .idl extension and that it's placed in the *fd* folder of the project.



Adding an Existing File Layout to the Current Project

isCOBOL IDE can also import an existing File Layout that is part of another project or workspace into the current project. To import an existing File Layout, right click on the project name in the isCOBOL Explorer area and choose *Import* from the pop-up menu. Then select *isCOBOL / isCOBOL File Layout* from the tree.



A wizard procedure will guide you in finding existing idl files to be imported into the project.

File Layout Structure



File Definition: contains the description of the file and the record.

Key: contains the description of the keys.

I/O Handling: configures which paragraphs must be generated for the I/O.

EFD: sets EFD directives for record fields.

Event Paragraph: contains user defined paragraphs for I/O handling.

To generate FD/SL copybooks right click on the File Layout name and select "Generate FD/SL" from the pop-up menu.

To view the generated copybooks right click on the File Layout and select "View *fdname.fd*", "View *fdname.sl*" or "View *fdname.prc*" from the pop-up menu.

File and Record Definition

File Definition

File Name: fd1 Optional

File Descriptor Settings

Advanced

Assign to

Device: RANDOM
Name: "FD1"

File Format

Format: Indexed
Access Mode: DYNAMIC
File Status: Status-FD1
Key Name:
Lock Mode: NONE

Comment:

Class

Name:
Variable:
Comment:

Record Definition

File Definition Key IO Handling EFD Event paragraph

File Name: Logical name of the file. This cannot be changed once the File Layout has been created.

Optional: Specifies whether the OPTIONAL clause should be generated in the SELECT statement for the file.

Advanced: Optional advanced file settings.

Device: Device of the file, choose between INPUT, OUTPUT, I/O, RANDOM, DISC, CONSOLE.

Name: Physical name of the file. If included between quotes is treated as constant string, otherwise its treated as a variable.

Format: Organization of the file. Choose between SEQUENTIAL, INDEXED or RELATIVE.

Access mode: Defines the access mode of the file. Choose between SEQUENTIAL, DYNAMIC or RANDOM .

File Status: Name of the file status variable. If omitted, no file status will be associated with the file.

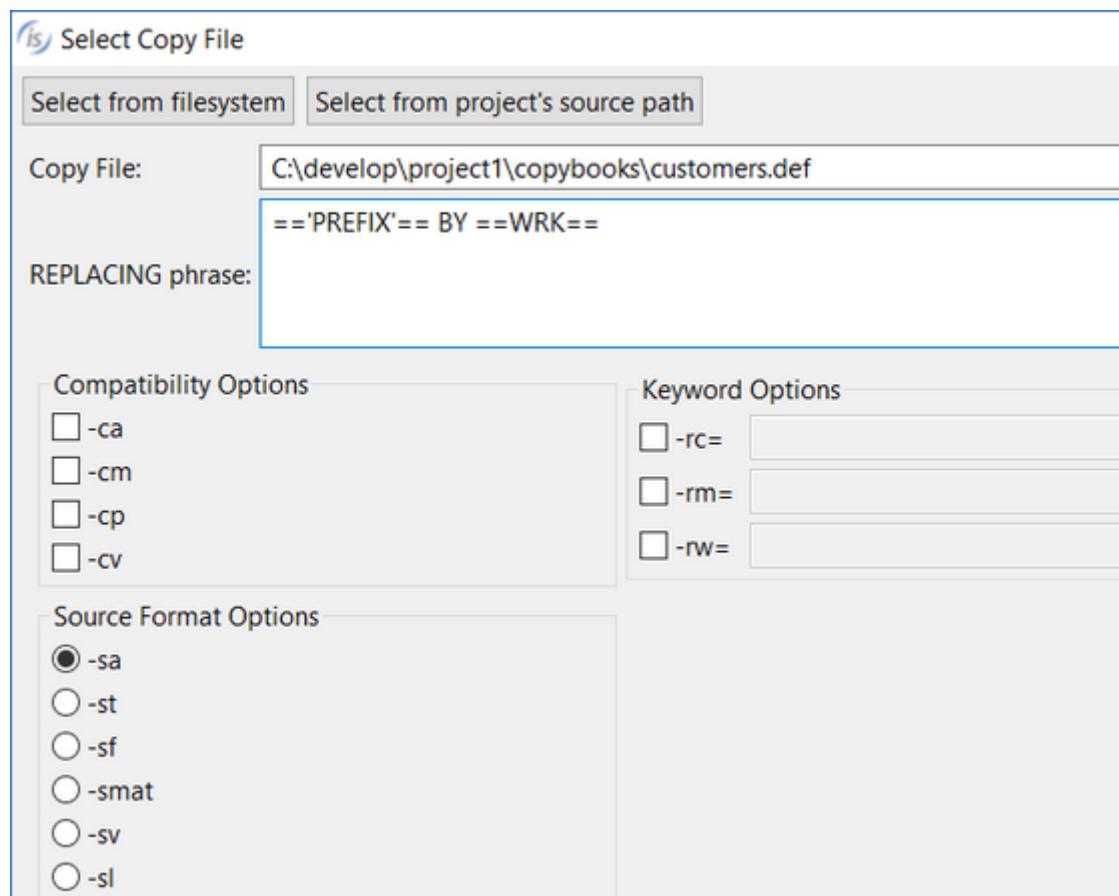
Record Definition

Property		Value
blank when zero		false
className		
comment		
copy-entry		
copy-file		
copy-path		
external		false
identified by		
is special names		None
justified		false
level		01
name		wrk-item
occurs		
picture	x(10)	
redefines		
same as		
sign is		None
synchronized		false
usage		
value		

- To create a new group variable, click on the *Add item* button or choose *01-49 Level* from the *Add Item* menu.
- To create a sub-item into a group variable, click on the *Insert a sub-item* button.
- To create a stand-alone variable, choose *77 Level* from the *Add item* menu.
- To create a constant, choose *78 Level* from the *Add item* menu.
- To create a boolean constant, choose *88 Level* from the *Add item* menu.
- To create a renames, choose *66 Level* from the *Add item* menu.
- To link a copybook, choose *Link Copy File* from the *Add item* menu. Linked copy files cannot be edited from inside the IDE, they're just referenced.
- To import all of the variables of a copybook, choose *Import Copy File* from the *Add item* menu. Once imported, the variables can be maintained from inside the IDE as well as if as they were created through it.

In the dialog shown by *Link Copy File* and *Import Copy File* it's possible to specify a **REPLACING** phrase as well as compatibility flags that the IDE should apply in order to parse the copybook content correctly.

Select from filesystem allows you to browse for a copybook in the whole file system; the full path of the copy book is used in this case. *Select from project's source path* allows you to browse for a copybook in the project folders; only the copybook name is used in this case.



Key Definition

The screenshot shows the 'Key' definition dialog box. At the top left is a 'Key' icon. Below it is a 'Key List' table with one row:

Name	Type	Fields
fdKey	Primary key unique	fd1-key

To the right of the table are four buttons: 'Modify Key', 'Remove Key', 'Up', and 'Down'. Below the table is a 'Fields Description' section. It contains a table with two columns: 'Field' and 'Name'. Under 'Field', there are three checkboxes: '01 fd1-item' (unchecked), '03 fd1-key' (checked), and '03 fd1-data' (unchecked). To the right of this table are three buttons: 'Add Key', 'Select all', and 'Deselect all'. At the bottom of the dialog are several tabs: 'File Definition', 'Key' (which is selected and highlighted in blue), 'IO Handling', 'EFD', and 'Event paragraph'.

To add a new key to the file:

1. select the fields that will be part of the key from the Fields Description. Double click in the Name column in order to display possible field name extensions.
2. click on the "Add Key" button
3. Edit the columns of the Key List to set key name and type
4. When more than one key has been defined, use "Up" and "Down" buttons to change the keys definition order

I/O Handling

The screenshot shows the 'IO Handling' tab in the isCOBOL IDE. At the top, there are three radio button options:

- Use Default Code Generation
- Use User-defined IO Paragraphs
- Do not generate any IO Paragraphs

Below these options is a section titled 'User Defined IO Paragraphs' containing a table:

Item	Value
Declaratives	
Open File	
Read File Key Is	
Read File Next	
Read File Previous	
Write	
Rewrite	
Delete File Record	
Delete File	
Close File	
Commit	
Rollback	

At the bottom of the tab, there is a navigation bar with the following tabs:

- File Definition
- Key
- IO Handling**
- EFD
- Event paragraph

Three options are available in the I/O Handling tab:

- Use Default Code Generation: The IDE will automatically generate paragraphs for I/O handling (default behavior)
- Use User-Defined I/O Paragraphs: The IDE will generate paragraphs written by the user in the Event paragraph section of the File Layout.
- Do not generate any I/O Paragraphs: no Procedure Division code will be generated for the file

EFD Directives

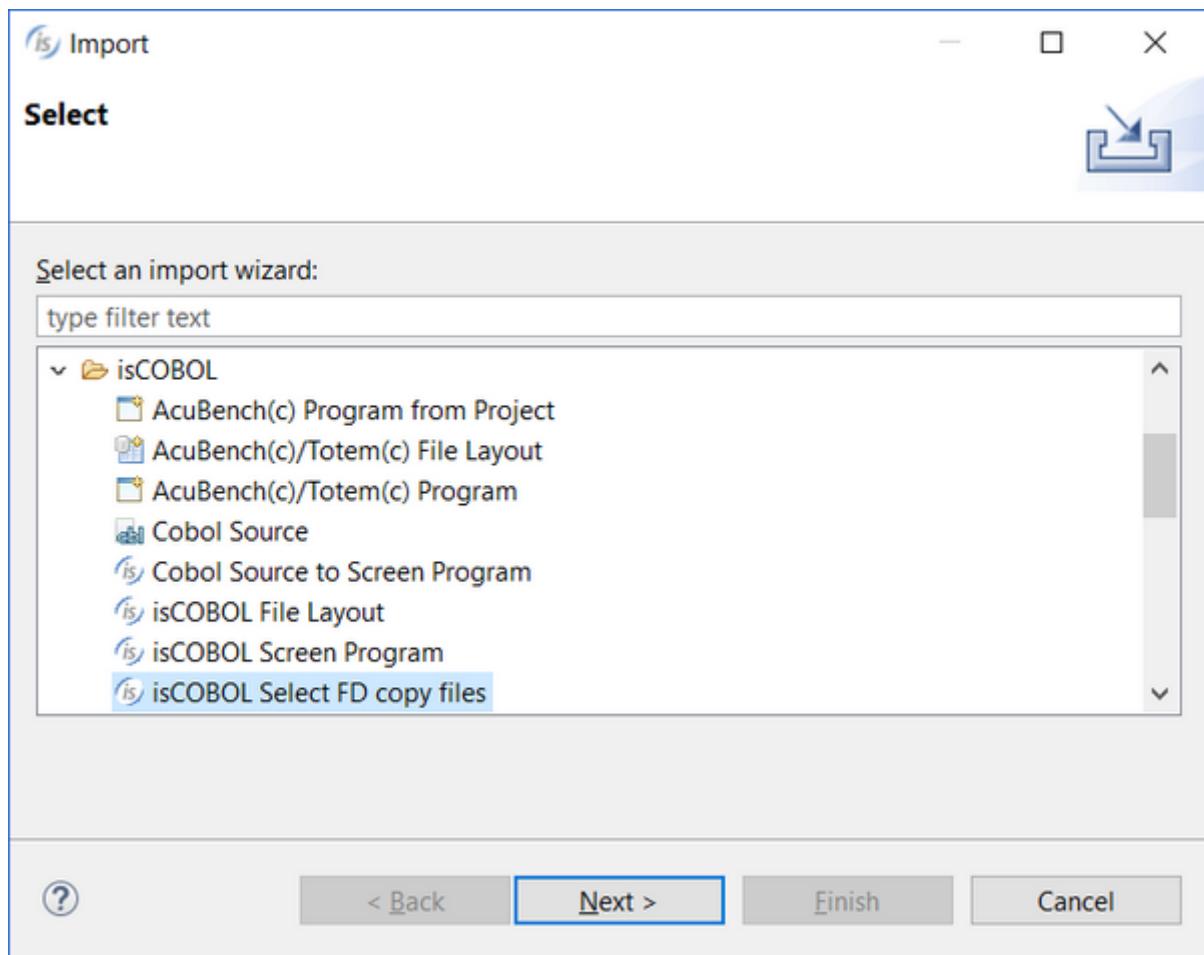
The EFD tab allows the user to define EFD directives that should be generated for the record fields. This kind of setting is optional.

Generating File Layouts from Existing Copybooks

isCOBOL IDE is able to generate a File Layout by importing existing fd and sl copybooks.

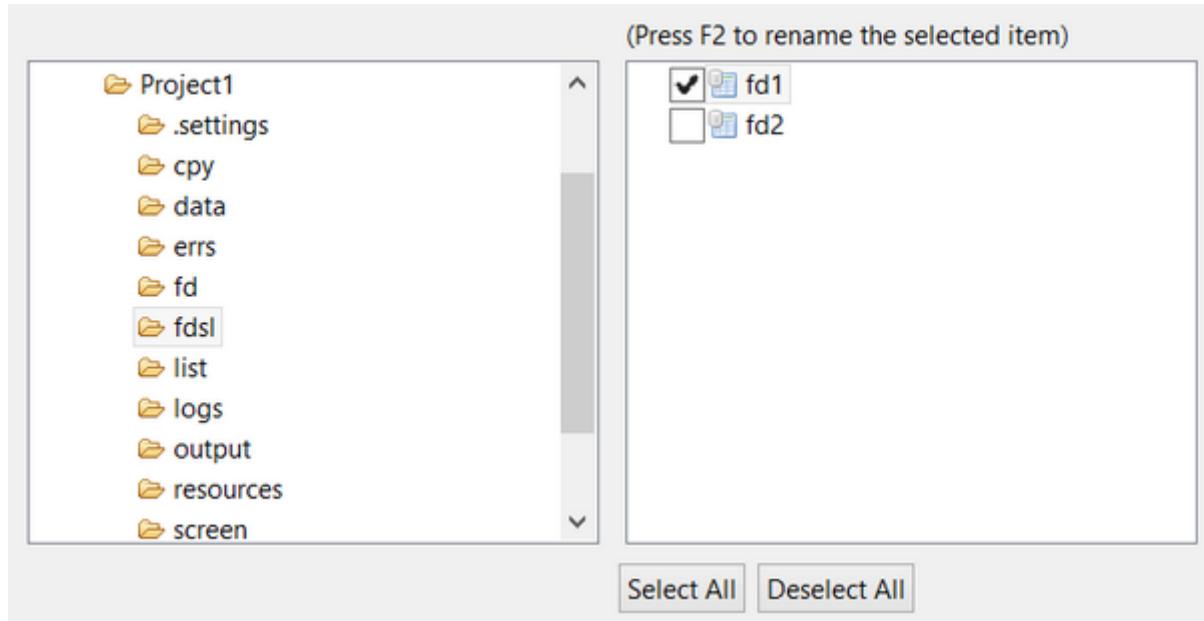
1. Right click on the project name in the isCOBOL Explorer area.
2. Choose *Import* from the pop-up menu.

3. Choose *isCOBOL / isCOBOL Select FD copy files* from the tree.



4. Browse to find the folder containing the copybooks, check the ones that you wish to import and click

Finish.



5. The files will appear between your FDs.

Datasets

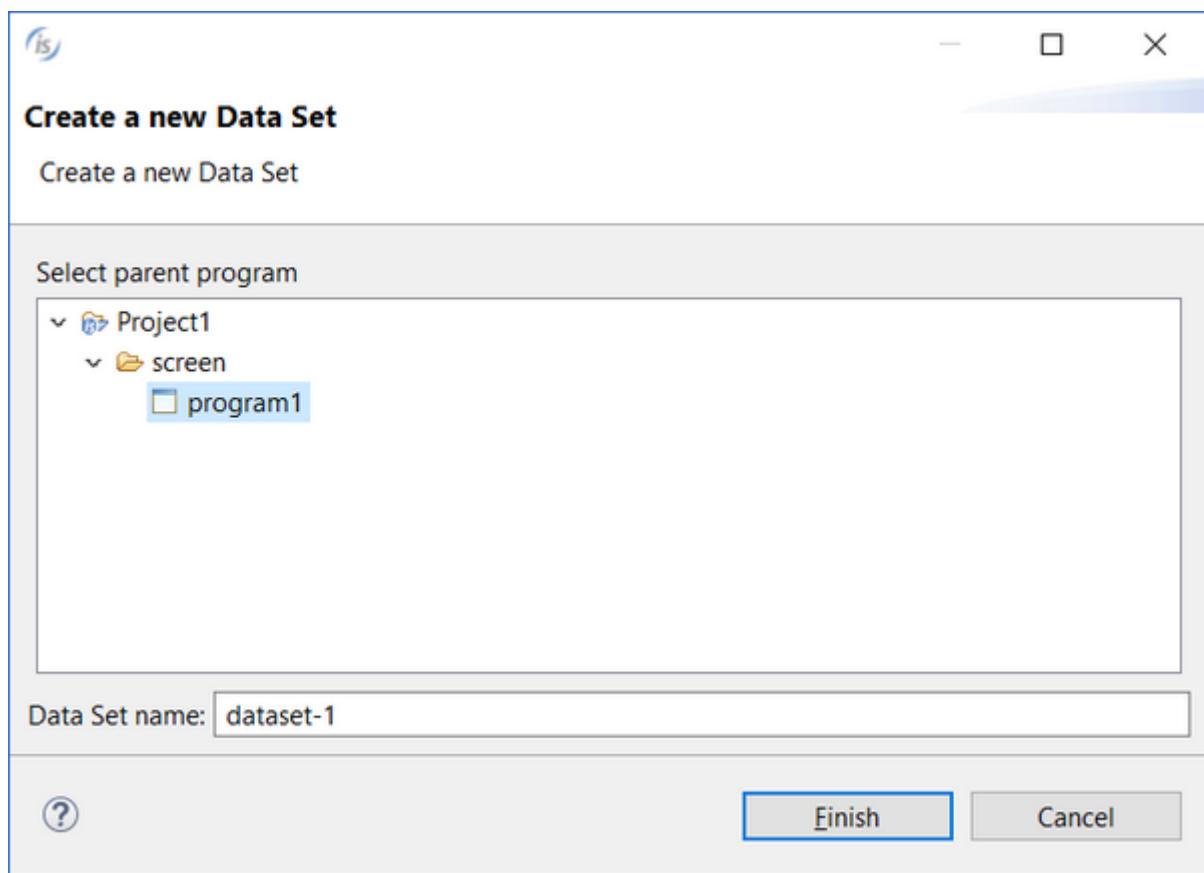
File Layouts are associated with the Screen Programs through Datasets.

A Dataset is a list of FD/SL that must be included in the program.

Creating a new Dataset

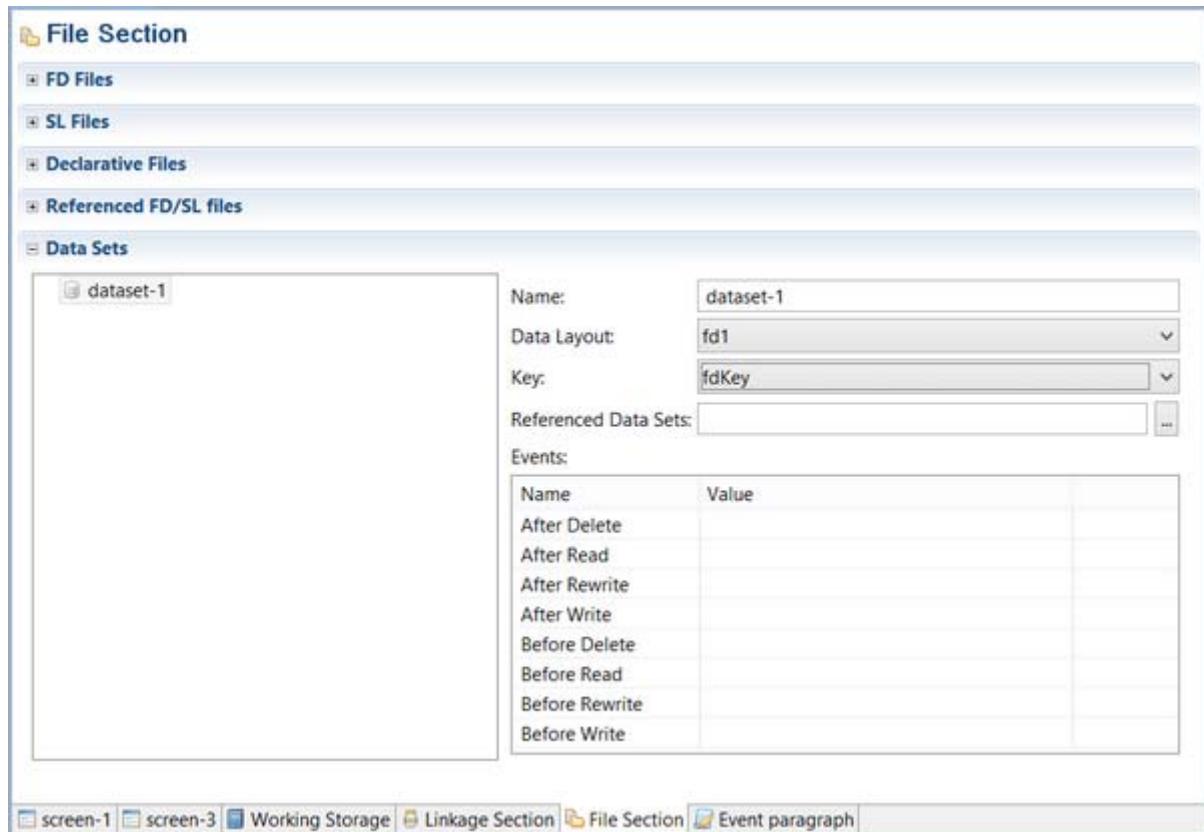
To create a new Dataset for the program, right click on the program name in the isCOBOL Explorer, select *New / Data Set* from the pop-up menu.

A panel will prompt you for the name of the Dataset and the program that will host it.



Once the Dataset has been added to the program, you can choose the following in the File Section or the Screen Program:

- Which fd is associated with the Dataset.
- Which key will be used by START and READ statements generated by the IDE.
- Which paragraphs must be executed before and after the standard I/O operations (optional).



External Paragraphs and Variables

isCOBOL IDE sets some paragraphs and variables as external. This means that even if they're declared in the Screen Program sections, the IDE will not generate them, assuming that they're available in the copybooks added by the user.

In order to set some variables and paragraphs as external

1. Right click on the program name in the isCOBOL Explorer area.
2. Choose *External Settings* from the pop-up menu.

It's also possible to define external paragraphs and variables in [Screen Program properties](#).

Code generation

Once the program has been designed, you can ask the IDE to generate the source code.

In order to generate FD/SL copybooks, in the *Data* page of the isCOBOL Explorer:

1. Right click on the fd name.
2. Choose *Generate FD/SL* from the pop-up menu.

In order to generate the code for the whole program, in the *Structural* page of the isCOBOL Explorer:

1. Click on the program name to select it.

2. Click the *Generate Program* button in the tool-bar.



or, alternatively

1. Right click on the program name..
2. Choose *Generate Program* from the pop-up menu

At this point, the quickest way to reach the generated source is:

1. Right click on the program name.
2. Choose *View %Program%.cbl* from the pop-up menu.

The IDE will automatically show the Code Editor with the generated source code inside.

Warning: any modification done inside the source code tagged areas (the blocks of code delimited by *begin and *end comments) will be lost during the next generation.

In order to configure which parts of the code must be generated in addition to the output format and other code generation options:

1. Click on the *Window* menu.
2. Choose *Preferences*.
3. Choose *isCOBOL / Code Generator* from the tree.

The *Code Generator* options also include *Program Tag*. In this panel its possible to choose one or more copybooks that must always be generated in the program's Working Storage and Procedure Division.

When the code generation is completed, generated files are available in the source, screen and cpy folders of your project and they can be compiled, run and debugged as the standard COBOL programs. Refer to [Compiling](#) and [Run and Debug](#) for information about compiling and debugging programs.

Do not modify generated files directly. When you need to modify a Screen Program, always do it using designers and paragraph editors and then regenerate the code, otherwise your modification will be overwritten during the next code generation.

Consistency Check

isCOBOL IDE offers the ability to check the project's consistency. To perform the consistency check right click on the project name in the isCOBOL Explorer and choose *Consistency Check* from the pop-up menu.

The following tests are performed:

- Check whether a variable referenced in a control property exists.
- Check whether a paragraph referenced in a control property exists.
- Check the compatibility of a variable's value with its picture.
- Check whether a data-layout referenced in a data-set exists.
- Check whether a fd-key referenced in a data-set exists.
- Check whether a copybook referenced in the 'FD Files'/'SL Files'/'Declarative Files' sections of the 'File Section' page exists.
- Check whether a data-layout contains at least one record description entry.

Any problem that is found is shown in the Consistency Check View.

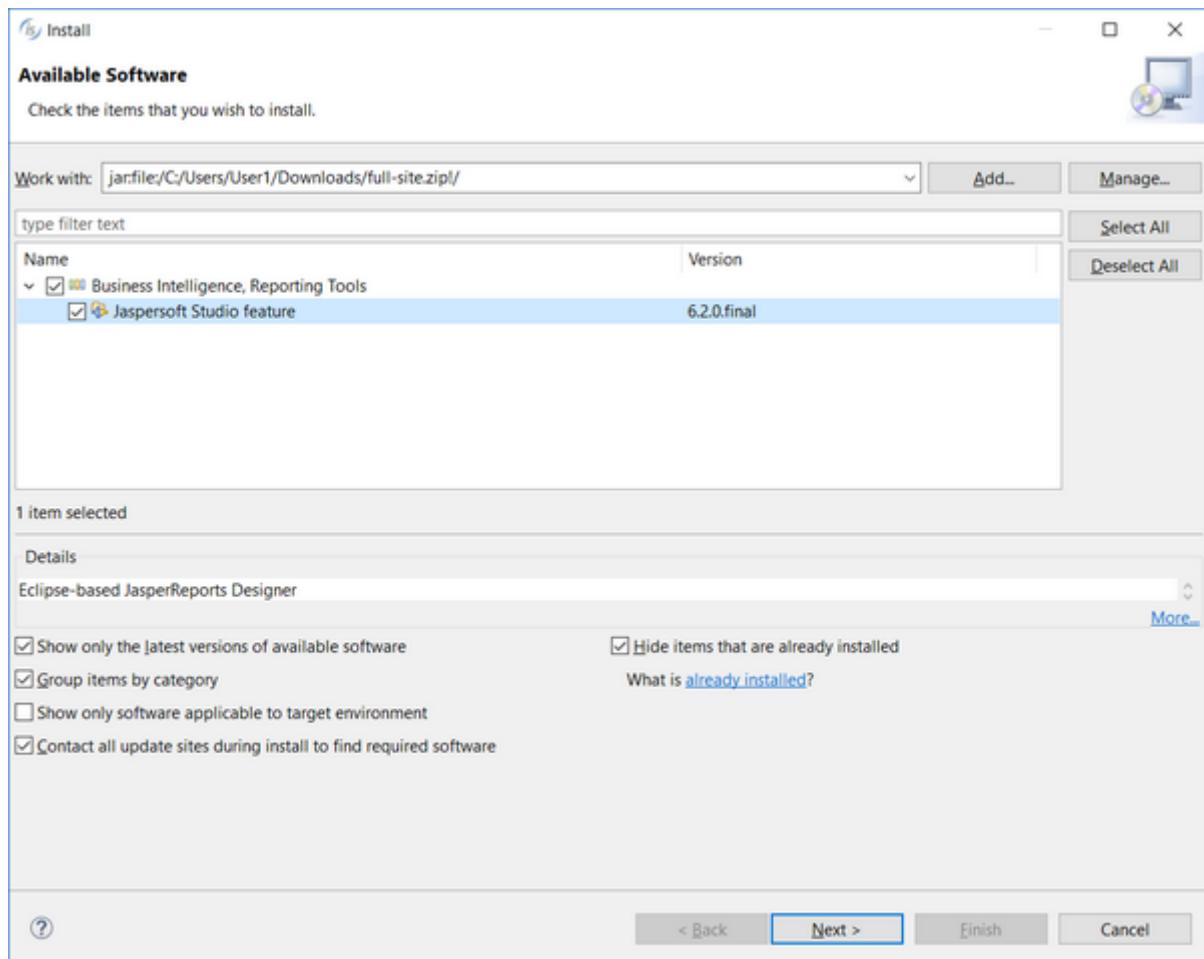
Jaspersoft Studio Integration

Jaspersoft Studio can be installed as an additional plugin in the isCOBOL IDE.

Version 6.2.0 is suggested as it includes all the necessary libraries for development and runtime.

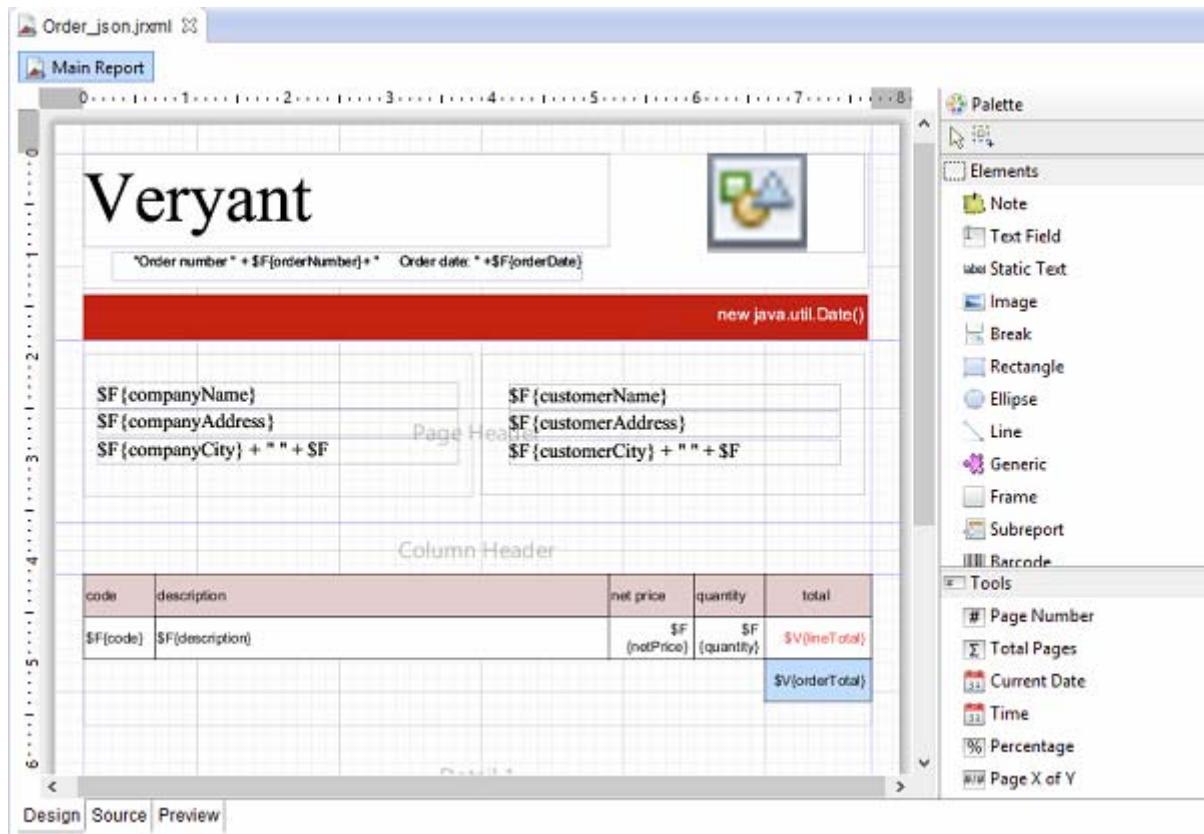
In order to install Jaspersoft Studio 6.2.0

1. Download the archive *full-size.zip* from <https://sourceforge.net/projects/jasperstudio/files/updatesite/6.2.0/>.
2. In the IDE, click on *Help* in the menu bar and choose *Install New Software...*
3. Click on the *Add...* button
4. Click on the *Archive...* button
5. Browse for the file *full-size.zip* previously downloaded
6. Click on the *Add* button



7. mark "Jaspersoft Studio feature" (as shown in the picture above) and go ahead to the end of the wizard. Accept license agreement and be OK to install software with unsigned content when required.

The Jaspersoft Studio Designer allows isCOBOL users to create sophisticated layouts containing charts, images, sub-reports, crosstabs and much more. Moreover it enables you to access your data through JDBC, TableModels, JavaBeans, XML, Hibernate, CSV, and custom sources. Then to publish your reports as PDF, RTF, XML, XLS, CSV, HTML, XHTML, text, DOCX, or OpenOffice.



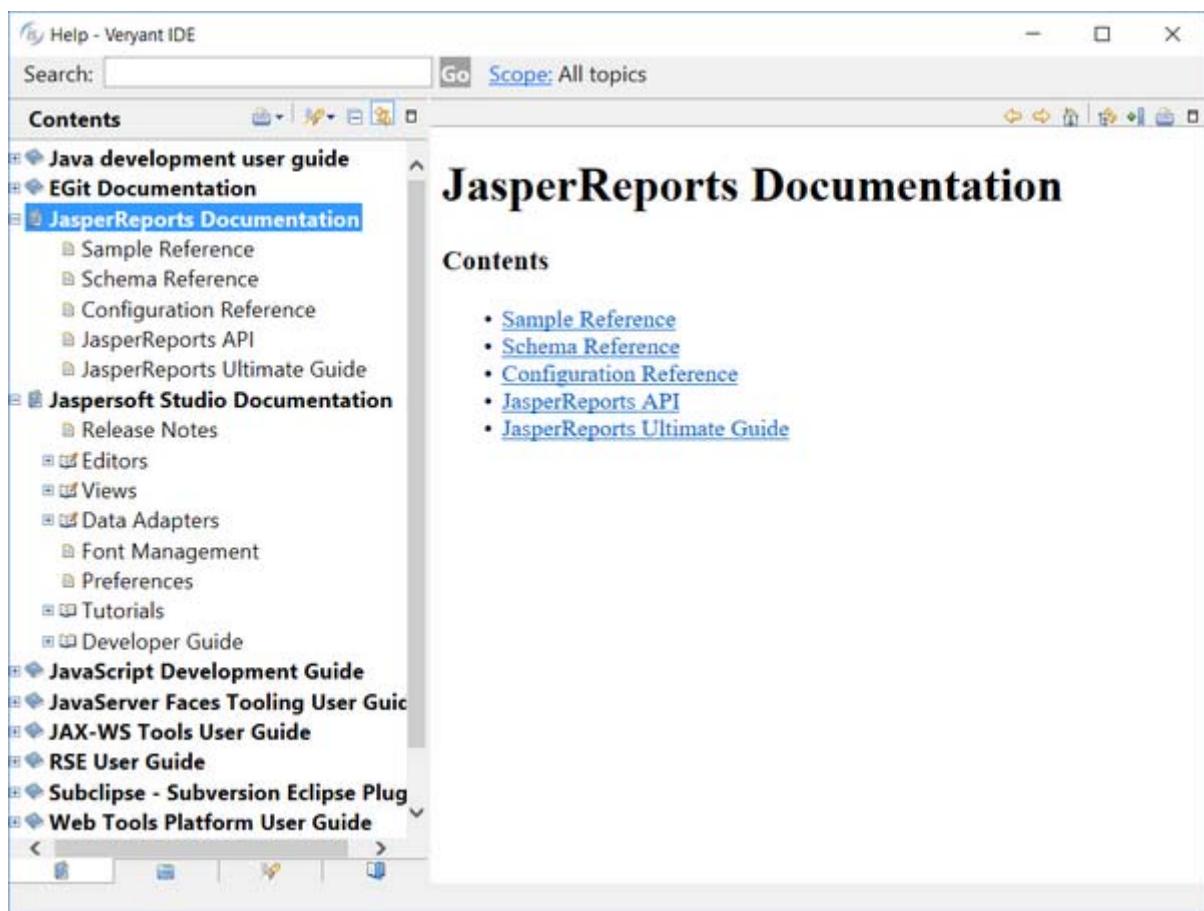
Jasper Usage Examples

Along with isCOBOL IDE are installed some examples of isCOBOL Projects that print reports using Jasper. The examples are located in the `sample\projects\jasper` subfolder, under the IDE installation folder. Follow the instructions fo the README file to add these sample projects to your workspace and try them out.

Jasper Documentation

Once the Jaspersoft Studio plugin has been installed, the corresponding documentation is available in the IDE.

In order to find it, click *Help* in the menu bar and select *Help Contents*.



The Jaspersoft Studio alternative

If you wish to avoid adding new plugins to your IDE, consider to download and install Jaspersoft Studio as a separate product. Setups are available at <https://community.jaspersoft.com/project/jaspersoft-studio/releases>.

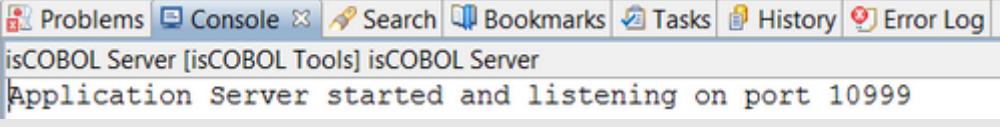
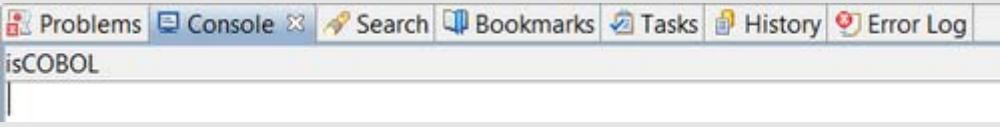
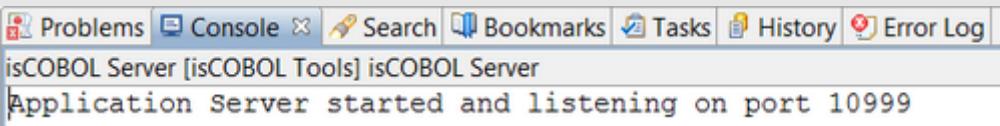
isCOBOL Tools

isCOBOL IDE allows you to manage the isCOBOL server-side components and to launch utilities. These features are accessible from the isCOBOL Tools menu in the menu bar.

Managing isCOBOL Server

1. Click on "isCOBOL Tools" in the menu bar.
2. Select "isCOBOL Server".

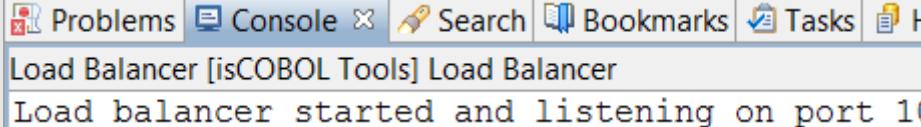
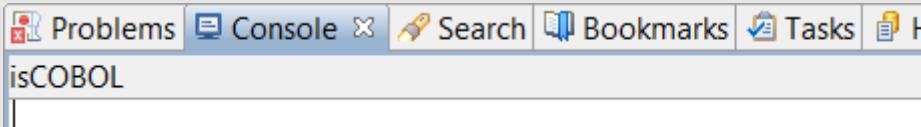
3. Select the desired action:

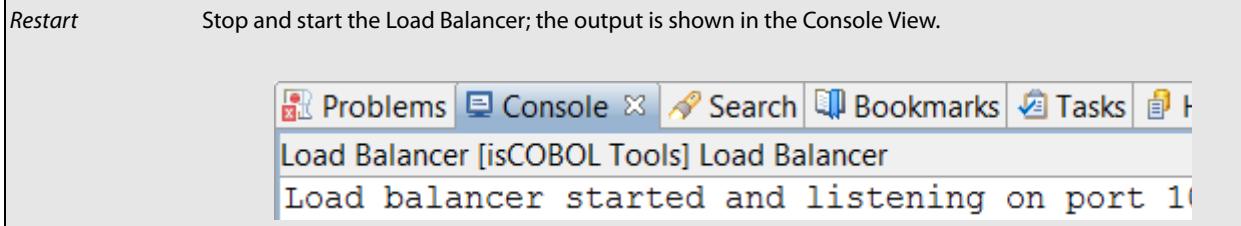
Start	Start the isCOBOL Server; the output is shown in the Console View.
	
Stop	Stop the isCOBOL Server; the Console View is cleaned.
	
Restart	Stop and start the isCOBOL Server; the output is shown in the Console View.
	

isCOBOL Server settings can be configured in the *Preferences* menu. See [isCOBOL Server Settings](#) for more information.

Managing Load Balancer

1. Click on “isCOBOL Tools” in the menu bar.
2. Select “Load Balancer”.
3. Select the desired action:

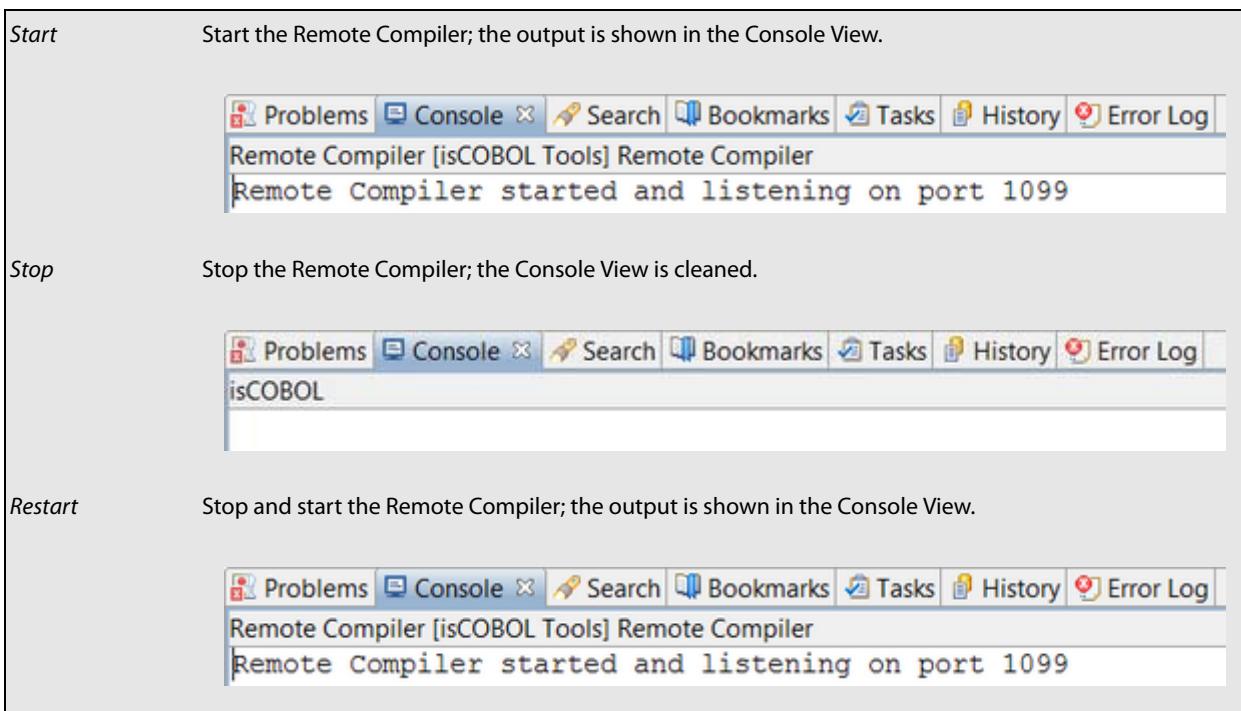
Start	Start the Load Balancer; the output is shown in the Console View.
	
Stop	Stop the Load Balancer; the Console View is cleaned.
	



Load Balancer settings can be configured in the *Preferences* menu. See [Load Balancer Settings](#) for more information.

Managing Remote Compiler server

1. Click on "isCOBOL Tools" in the menu bar.
2. Select "Remote Compiler".
3. Select the desired action:



The Remote Compiler can be configured in the *Preferences* menu. See [Remote Compiler Settings](#) for more information.

Note: With this operation you start a Remote Compiler server on the local machine. This is useful only for test purposes, since usually the Remote Compiler server runs on a server machine with precompilers installed.

Launching isCOBOL Utilities

1. Click on "isCOBOL Tools" in the menu bar.
2. Select "Utilities".

3. Select one of the following utilities:

- Index and Relative File Editor
- Index File Migration
- Jdbc2FD
- Color Picker
- Stream2Wrk
- isCOBOL Launcher
- JavaBean Copy Generator
- COBFILEIO
- JUTIL

When either JUTIL or Stream2Wrk is selected, since they're command line utilities, the IDE shows a graphical dialog in which you can configure the command. The output of the command is shown in the [Console](#) view.

Importing programs from AcuBench

AcuBench(TM) is the integrated development environment member of the extend(TM) family of Acucorp solutions.

The following files can be imported:

- psf: All items described in AcuBench program files are imported.
- dlt: Data layouts can be imported as well.

The following files cannot be imported:

- stf: The screen is imported along with the program (psf). It cannot be imported separately.
- pjt: AcuBench project files are not recognized.
- wtf: The report is imported along with the program (psf). It cannot be imported separately.

Note - AcuBench generated paragraphs are maintained in the source code, but their content is moved to the corresponding IDE generated paragraph and they just jump to it.

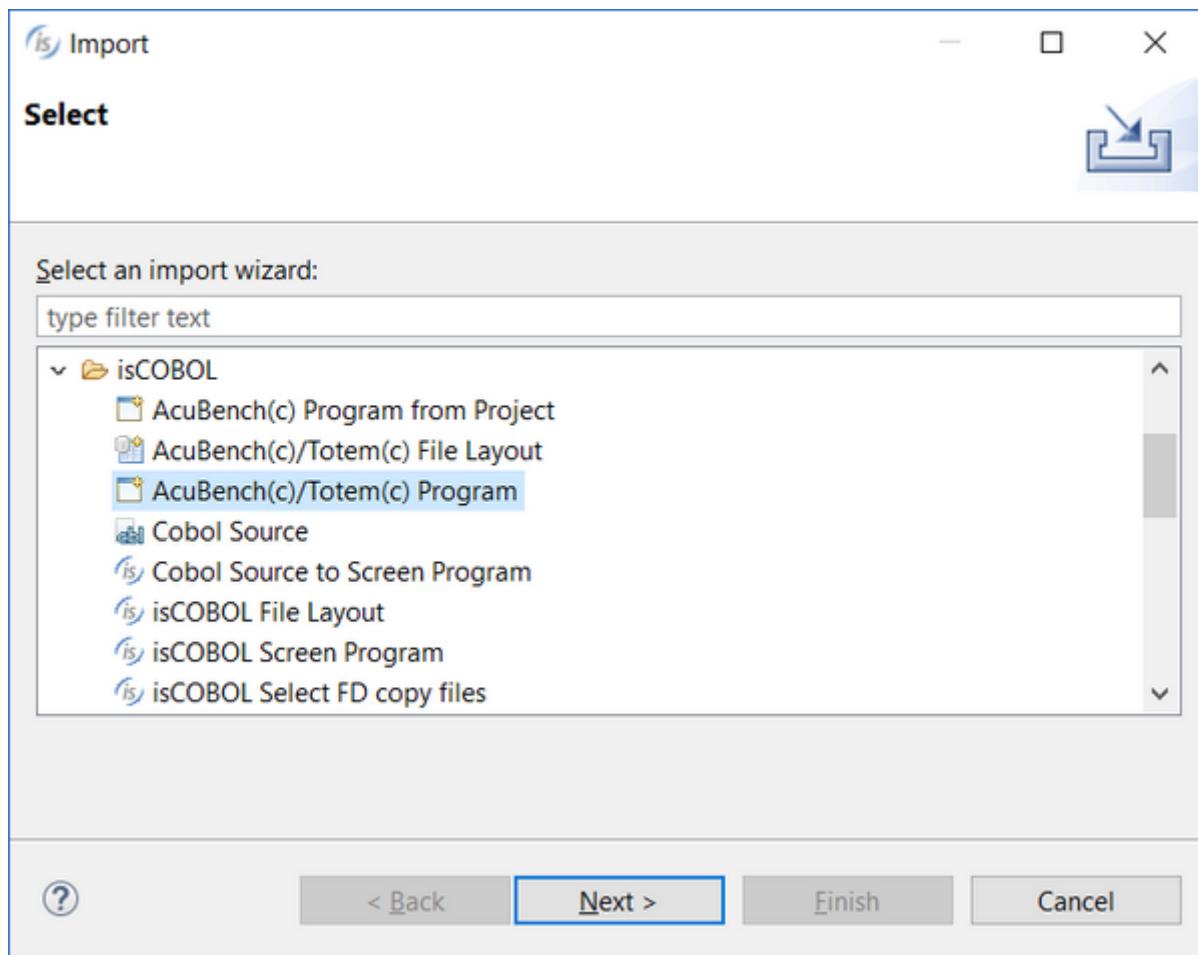
The best practice for importing AcuBench programs and their items consists in the following steps, in order:

1. link custom copy books, if any
2. import file layouts
3. import programs

To import an AcuBench program:

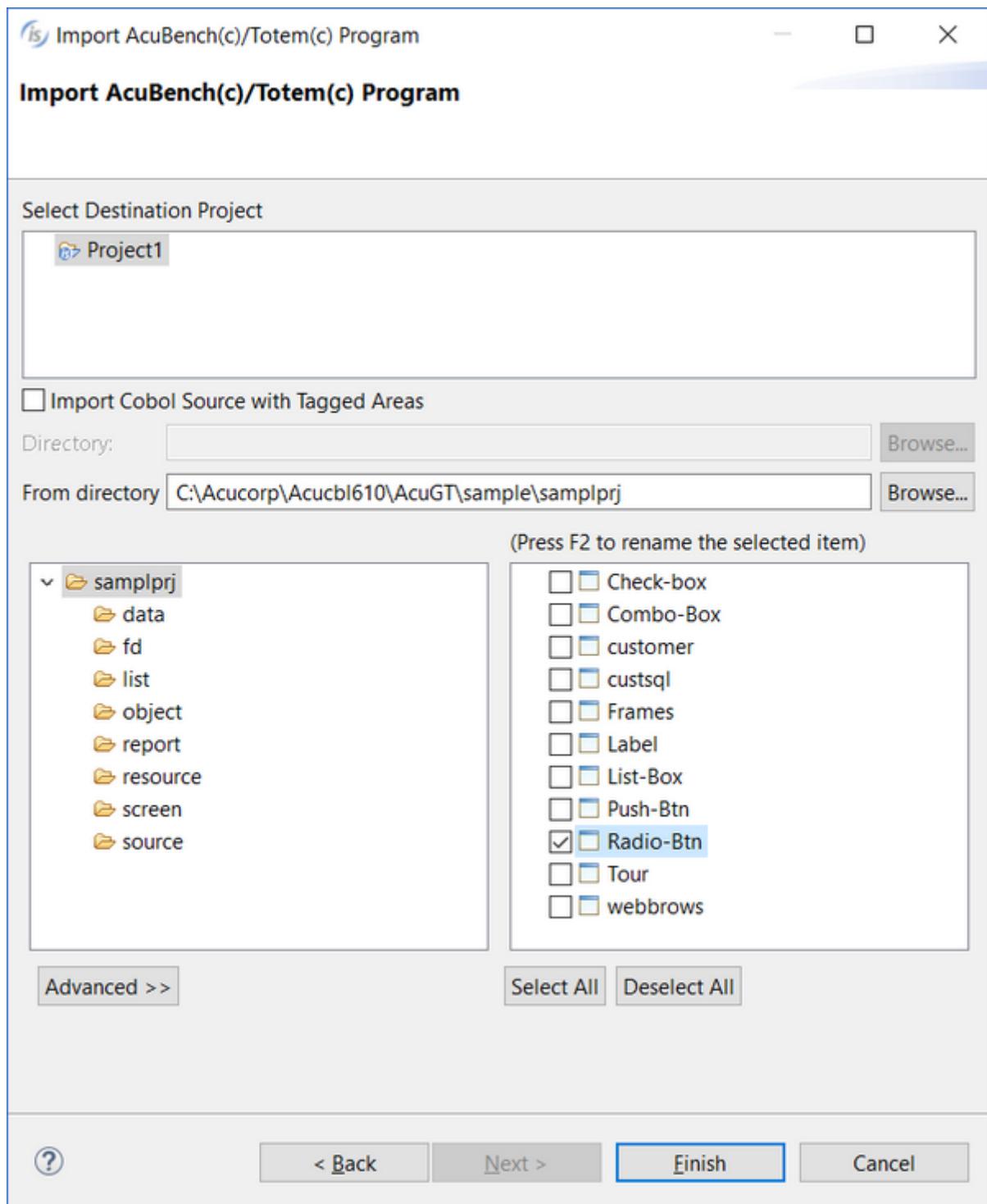
1. Right click on the project name in the isCOBOL Explorer area.
2. Choose *Import* from the pop-up menu.

3. Choose *isCOBOL / AcuBench(c)/Totem(c) Program* from the tree.



Note - the *AcuBench(c)/Totem(c) Program* option reads psf files from disk. This is useful for standard AcuBench project that use default settings. If your AcuBench project include some customization made at project level (e.g. different file extensions for the generated source files), then you might prefer the option *AcuBench(c) Program from Project*, that retrieves the list of psf files and their settings by reading the pjt file.

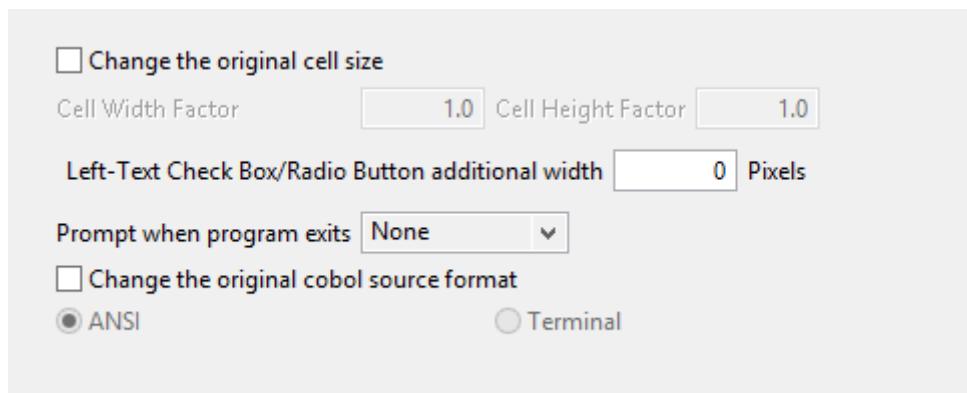
4. Browse to find the psf files, check the ones that you wish to import.



If the programs included custom code written outside of the tagged areas, check the option *Import Cobol Source with Tagget Areas* and provide the directory where the cbl files are stored.

5. Before clicking on the *Finish* button you can optionally set one of the advanced options, such as changing the source format, set the action to be performed at program exit or altering the original cell size. Click on the

Advanced >> button to show this panel:

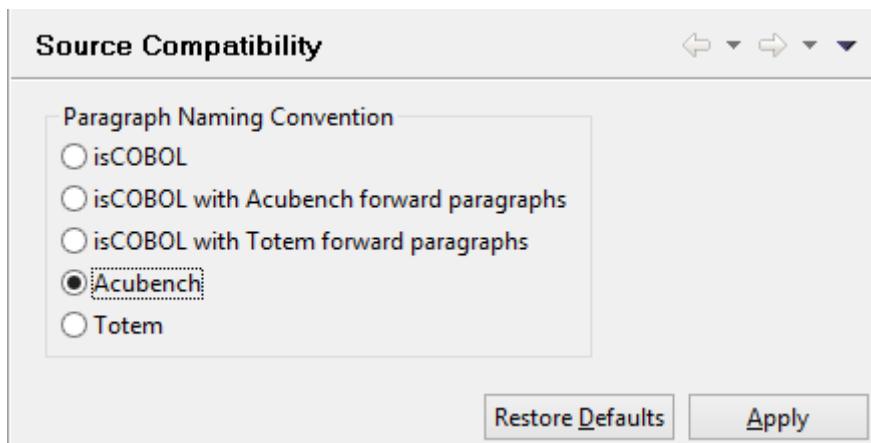


6. After clicking on *Finish*, the program will appear between your Screen Programs.

Changing the paragraph naming convention

Internal paragraphs such as "Acu-Initial-Routine" are maintained as they are. If you prefer to have isCOBOL names such as "is-initial-routine" for them, or if you prefer to have both names with Acu's paragraphs redirecting to isCOBOL's paragraphs (like it happened with previous versions of the IDE)

1. right click on the program name in the Structural View
2. choose *Properties*
3. choose *Screen Program / Source Compatibility*
4. select the paragraph naming convention that you prefer



AcuBenchPrint

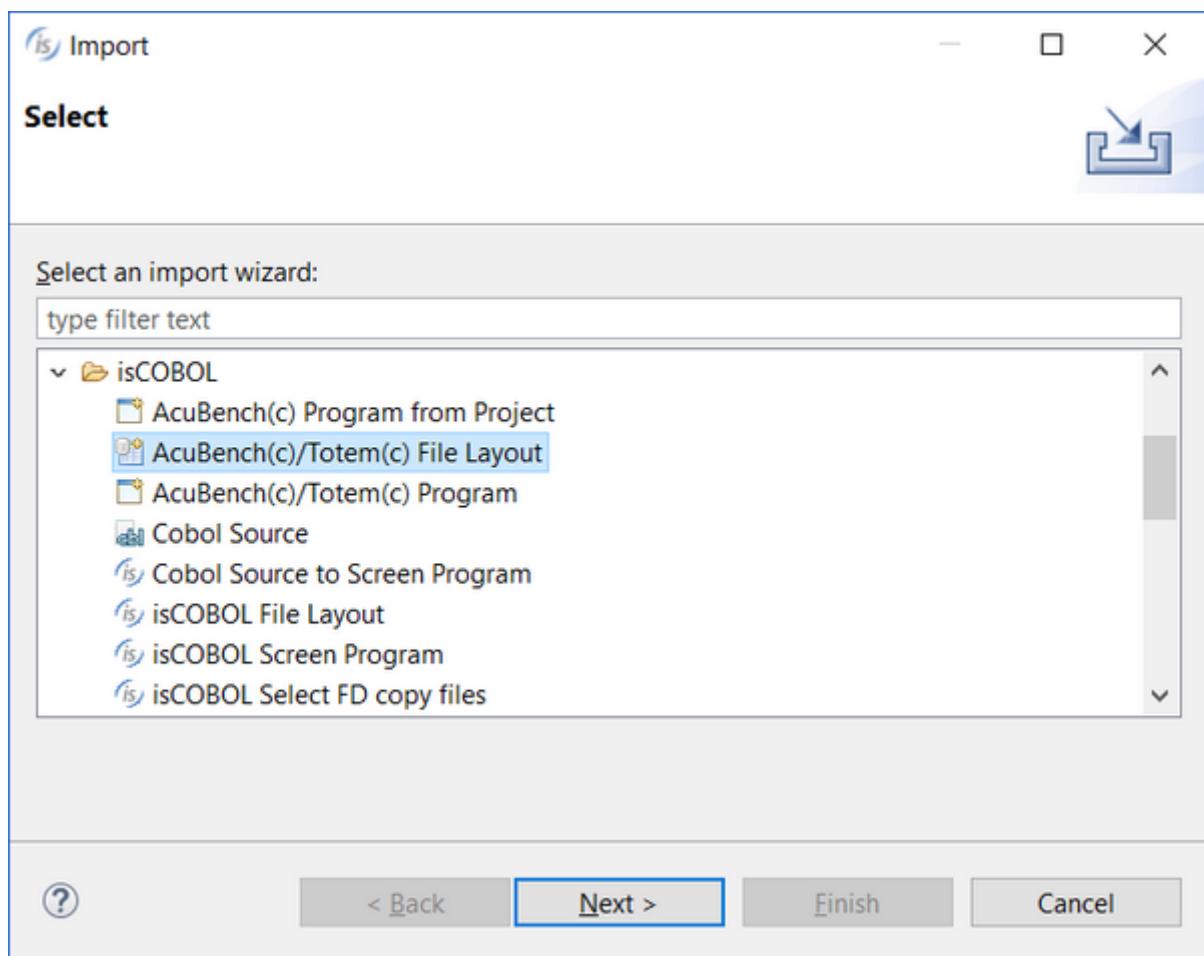
While the AcuBench reports are executed as HTML files to be printed via IE Active/X using the AcuBenchPrint.dll wrapper, the isCOBOL support for AcuBench reports includes an HTML rendering tool that is able to print and preview generated HTML without any IE Active/X dependences. In practice, there's no more need of AcuBenchPrint.dll in order to print or preview reports imported from AcuBench under isCOBOL.

Importing a Data Layout from AcuBench

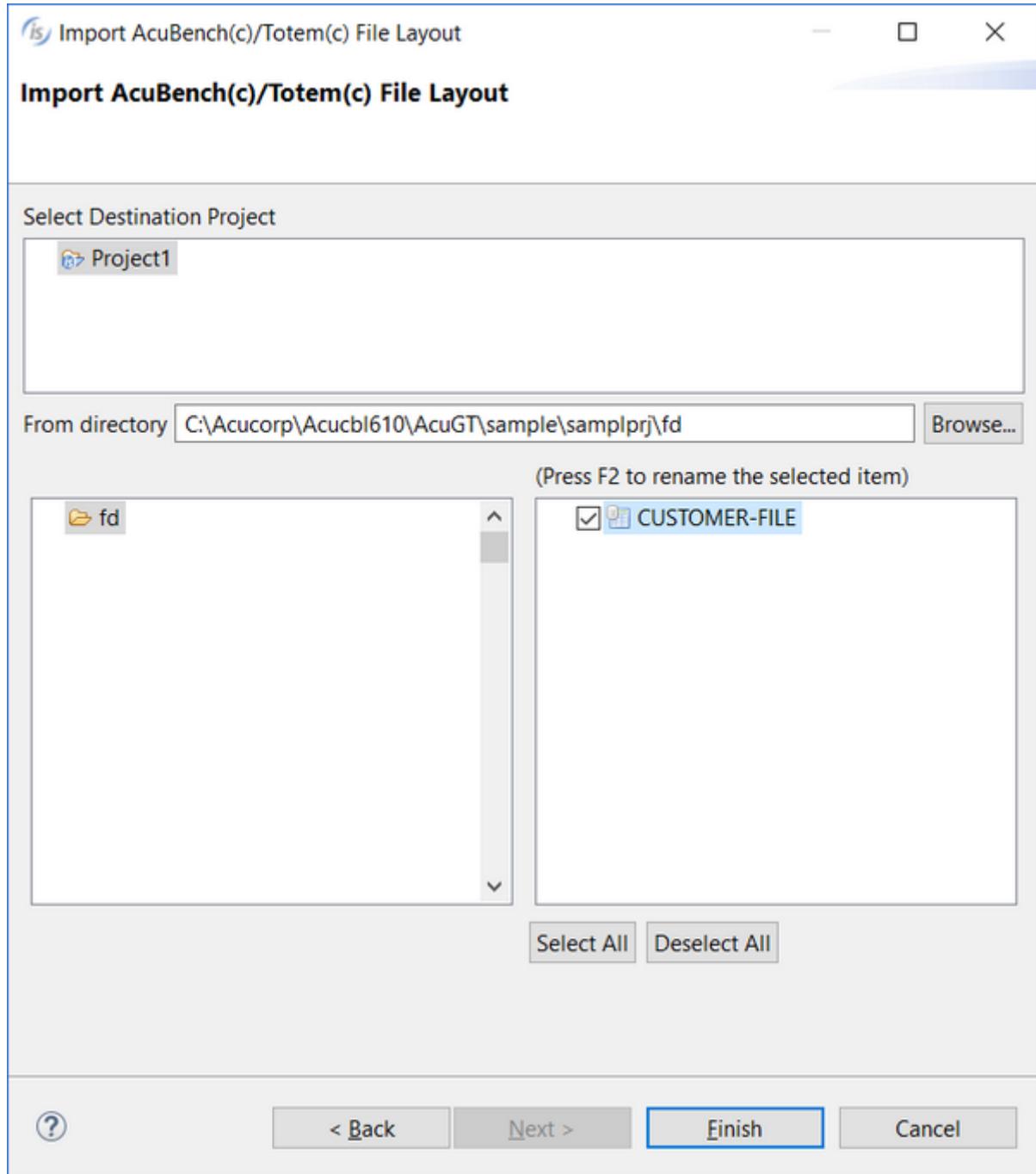
To import an AcuBench Data Layout:

1. Right click on the project name in the isCOBOL Explorer area.

2. Choose *Import* from the pop-up menu.
3. Choose *isCOBOL / AcuBench(c)/Totem(c) File Layout* from the tree.



4. Browse to find the dlt files, check the ones that you wish to import and click *Finish*.



5. The file will appear between your FDs.

Importing programs from Totem

Totem(TM) is the integrated development environment supplied by CASEMaker.

The following files can be imported:

- cpf: all items described in Totem program files are imported except for reports. (Reports are currently not supported)
- clf: Data layouts can be imported as well.

The following files cannot be imported:

- stf: The screen is imported along with the program (cpf). It cannot be imported separately.
- cwf: Totem project files are not recognized.
- wtf: The report is imported along with the program (cpf). It cannot be imported separately.
- dmd: Data modeling diagrams are not recognized.
- skl: Skeleton files for code generation are not recognized by isCOBOL IDE. isCOBOL IDE has its own generation rules.

During the import process:

- Totem Data Controls are translated into standard GUI controls, since Data Controls are not supported by isCOBOL IDE.
- The above behavior also applies to Custom Controls, which will not be maintainable in the IDE Screen Designer.

Note - Totem generated paragraphs are maintained in the source code, but their content is moved to the corresponding IDE generated paragraph and they just jump to it.

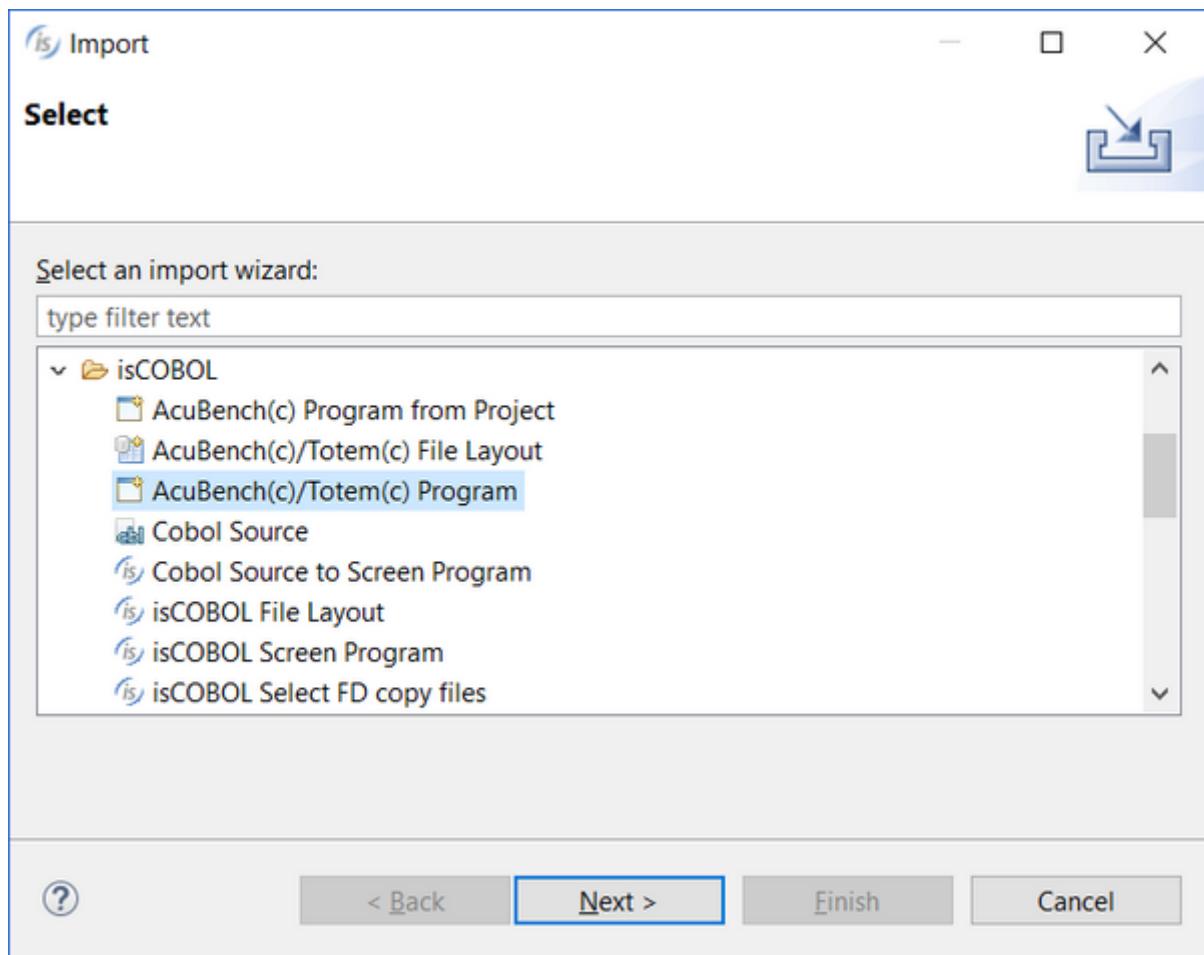
The best practice for importing Totem programs and their items consists in the following steps, in order:

1. link custom copy books, if any
2. import file layouts
3. import programs

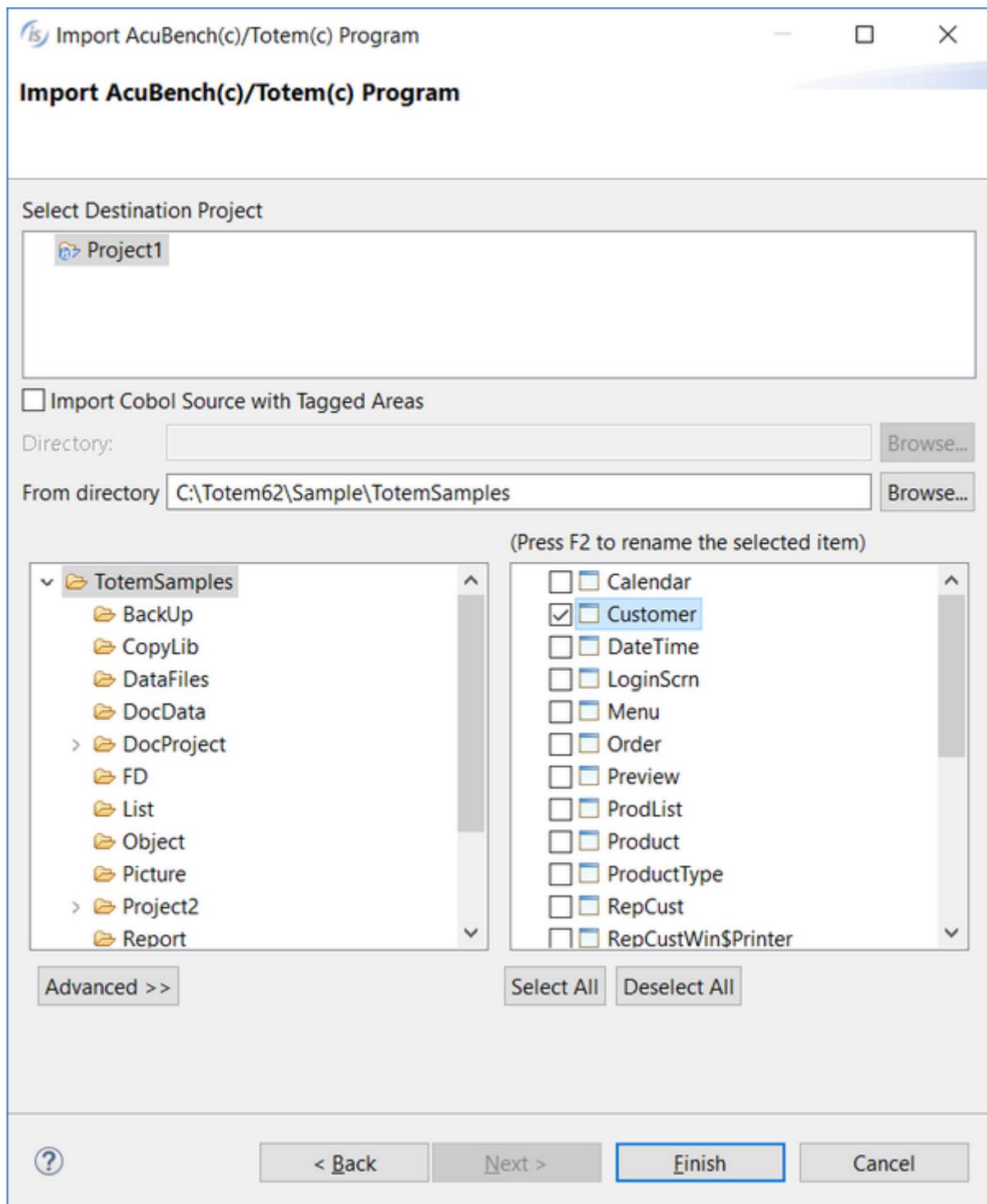
To import a Totem program:

1. Right click on the project name in the isCOBOL Explorer area.
2. Choose *Import* from the pop-up menu.

3. Choose *isCOBOL / AcuBench(c)/Totem(c) Program* from the tree.



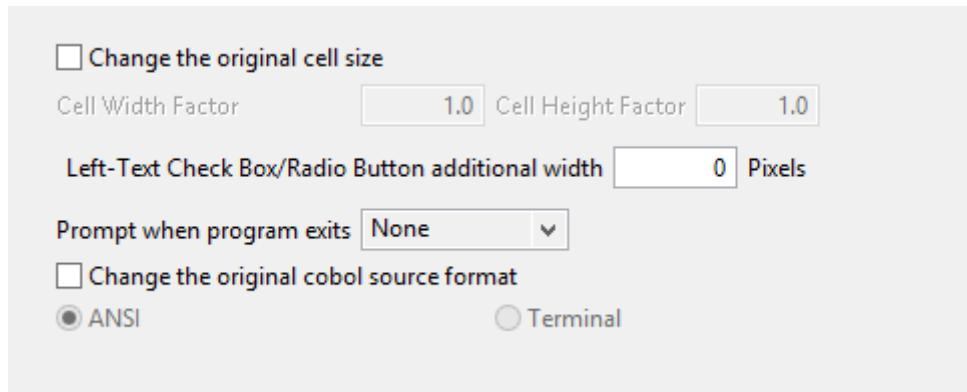
4. Browse to find the cpf files, check the ones that you wish to import



If the programs included custom code written outside of the tagged areas, check the option *Import Cobol Source with Tagget Areas* and provide the directory where the cbl files are stored.

5. Before clicking on the *Finish* button you can optionally set one of the advanced options, such as changing the source format, set the action to be performed at program exit or altering the original cell size. Click on the

Advanced >> button to show this panel:

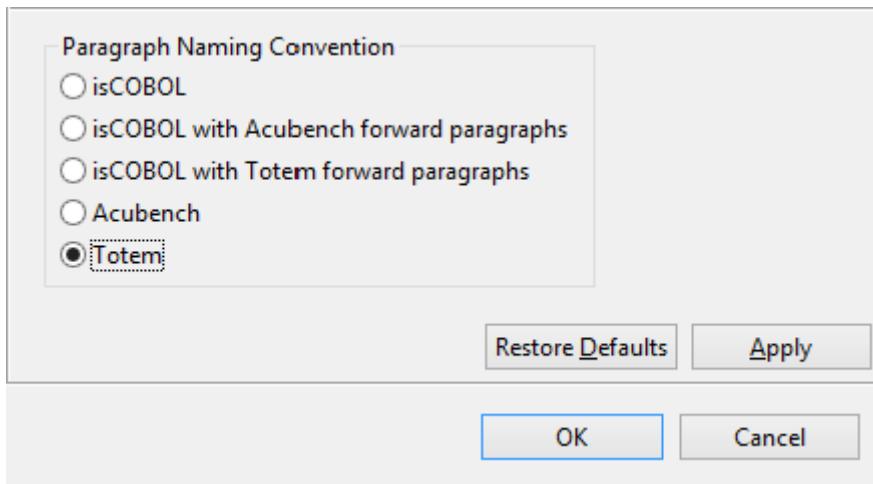


6. The program will appear between your Screen Programs.

Changing the paragraph naming convention

Internal paragraphs such as "Initialize-Routine" are maintained as they are. If you prefer to have isCOBOL names such as "is-initial-routine" for them, or if you prefer to have both names with Totem's paragraphs redirecting to isCOBOL's paragraphs (like it happened with previous versions of the IDE)

1. right click on the program name in the Structural View
2. choose *Properties*
3. choose *Screen Program / Source Compatibility*
4. select the paragraph naming convention that you prefer

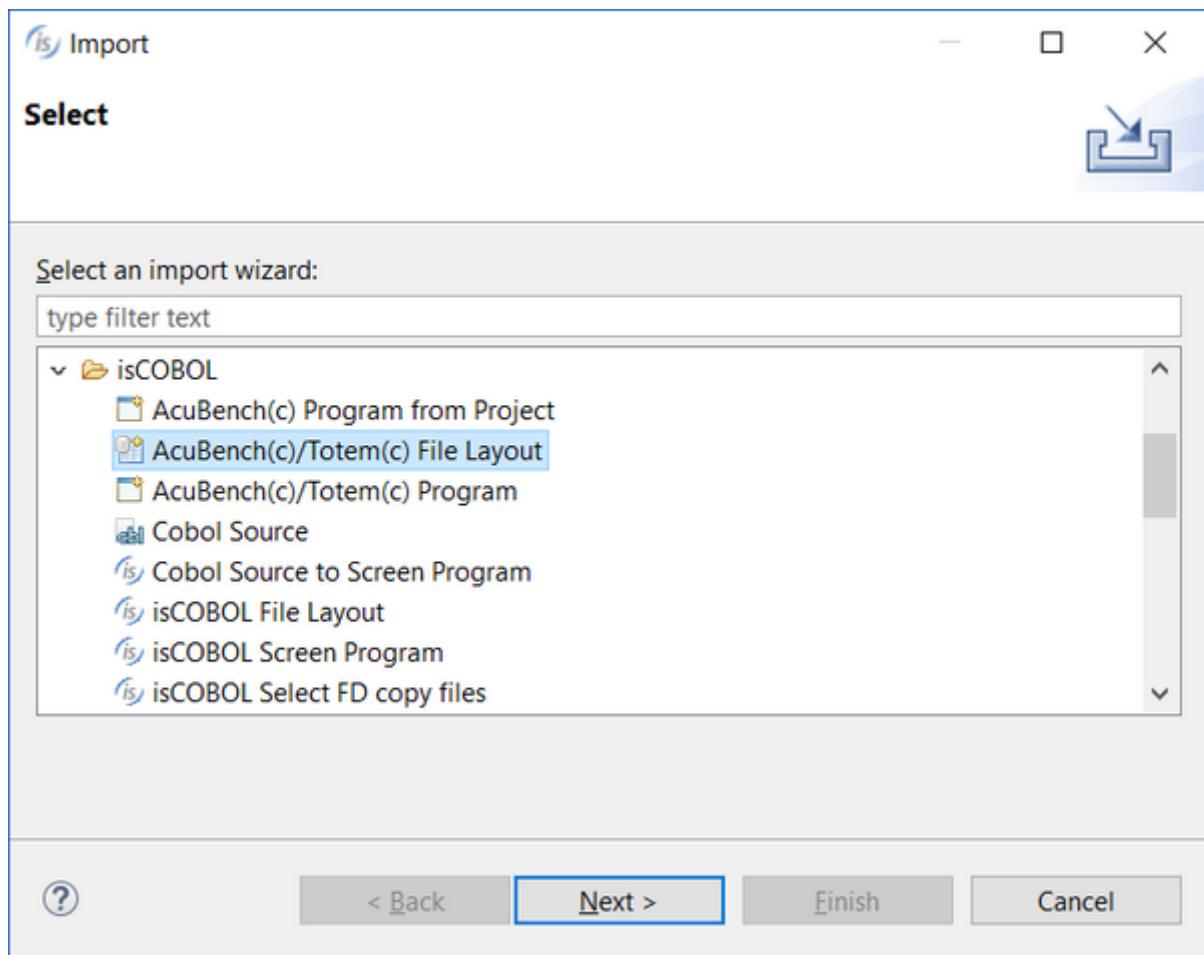


Importing a Data Layout from Totem

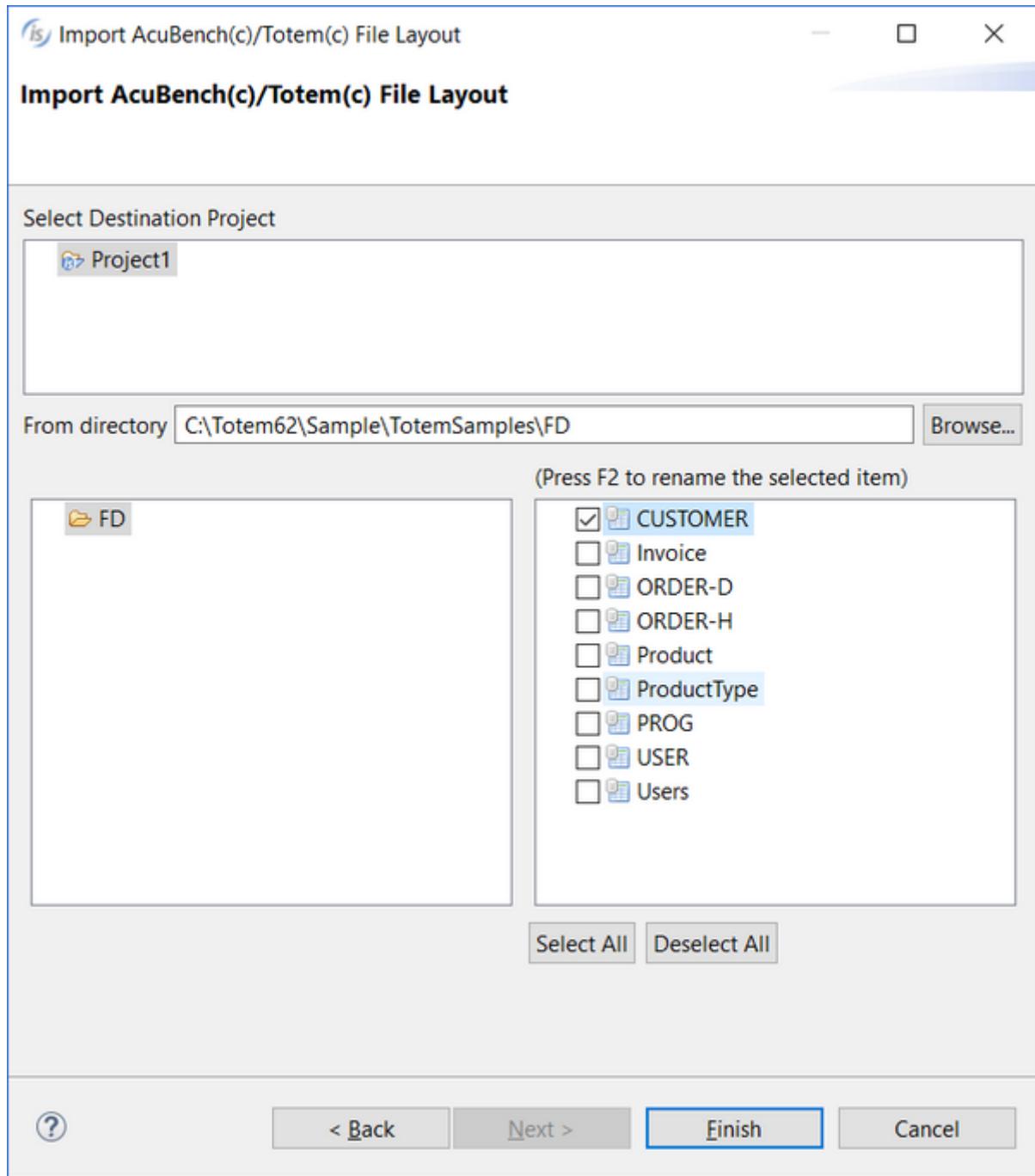
To import a Totem Data Layout:

1. Right click on the project name in the isCOBOL Explorer area.
2. Choose *Import* from the pop-up menu.

3. Choose *isCOBOL / AcuBench(c)/Totem(c) File Layout* from the tree.



4. Browse to find the clf files, check the ones that you wish to import and click *Finish*.



5. The file will appear between your FDs.

Importing programs from Cobol WOW

This argument is discussed in the Transitioning from RM/COBOL guide.

See [Importing projects from Cobol-WOW in isCOBOL IDE](#) for more information.

Importing a Program with Screen Section

isCOBOL IDE allows to import programs with Screen Section in order to maintain their screen through the Screen Designer. Programs source files are read from disk and transformed into isCOBOL Screen Programs.

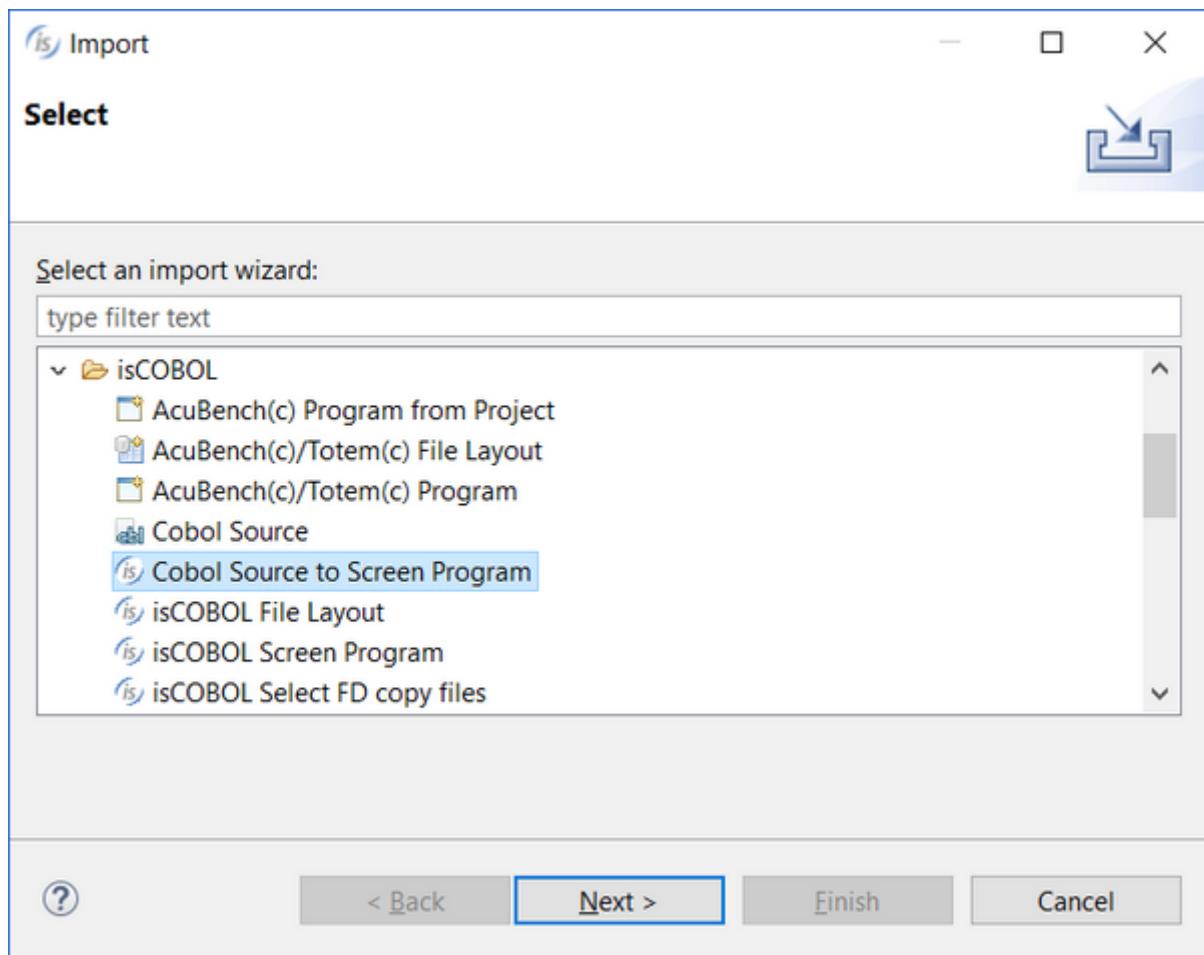
Before performing this kind of import, the following steps have to be followed:

1. Ensure that the IDE is set to generate code in the same format of the program you're going to import. This setting is available by choosing *Window* in the menu bar, then *Preferences* and eventually selecting *isCOBOL > Code Generator* from the tree. The "Source Format" setting must be "ANSI" or "Terminal" according to the format of the source file(s) you're going to import.
2. In the *Properties* of the project that will host your program, change compiler options by activating all the flags that are necessary for the compilation of the program you're going to import. For example, if the source code contains the syntax RECORDING MODE IS V, activate the -cv option. Also ensure that the -sp option contains all the necessary paths to find the copybooks declared in the program; these paths must be physical paths selected from the file system, Eclipse aliases are currently not considered by the import logic.
3. If the program you're going to import uses files, ensure that these files are described in separate copybooks (sl and fd copybooks) and import these copybooks as described in [Generating File Layouts from Existing Copybooks](#).

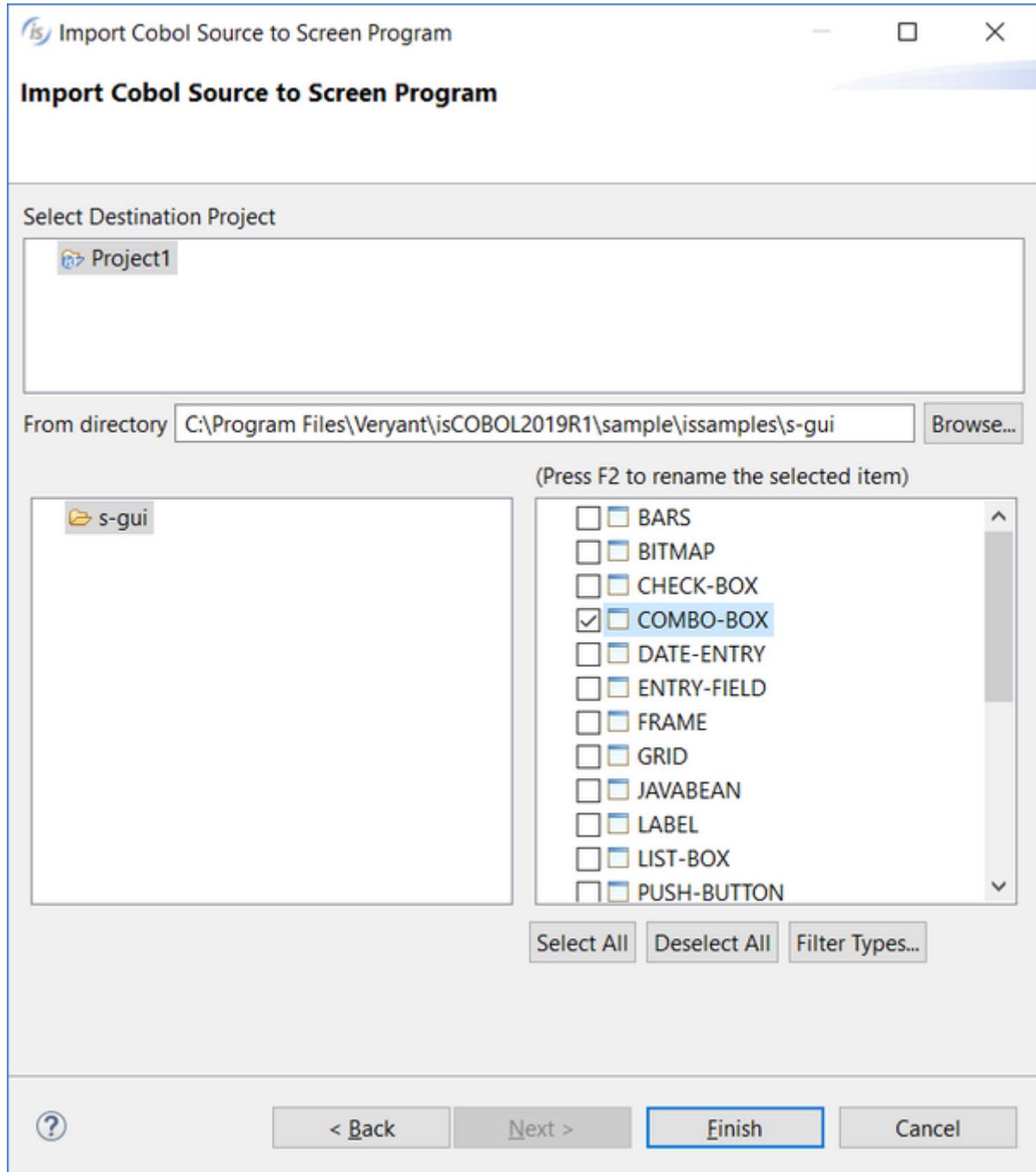
At this point you can proceed with the import of the program with Screen Section.

1. Right click on the project name in the isCOBOL Explorer area.
2. Choose *Import* from the pop-up menu.

3. Choose *Cobol Source to Screen Program* from the tree



4. Browse to find the source files, check the ones that you wish to import and click *Finish*



By default, only files with ".cbl" extension are considered. If your source files have different extensions, click on the *Filter Types...* button to configure it.

5. The program will appear between your Screen Programs.

This kind of Import is subjected to the following limitations:

1. The IDE doesn't know window dimensions, so in the Screen Designer, the properties Lines and Size might not be set correctly. If so, you can edit them to correct window dimensions.
2. Coordinates and size properties that are set with variables in the imported program are set to their

default in the Screen Designer. In this case the layout of Screen in the Designer might not match the layout you see at runtime.

3. Bitmaps and fonts referenced in the imported program are not shown in the Screen Designer, despite they appear correctly when you run the program.
4. Constant data items (level 78) are replaced by their values in controls properties.

Command-line usage

isCOBOL IDE implements some command-line switches that allow you to perform tasks in background mode, without user interaction.

The following tasks can be performed:

[Generate and build projects](#)

[Import AcuBench/Totem programs \(PSF and CPF\)](#)

[Import AcuBench/Totem file layout \(DLT and CLF\)](#)

[Import SL/FD copy files](#)

[Import COBOL program](#)

Note - Despite for your convenience only few important options are shown in this documentation, all the Eclipse command line options can be used. Refer the the Eclipse documentation for further details.

Generate and build projects

Command:

```
isIDE -data workspaceLocation -nosplash --launcher.suppressErrors -application  
com.iscobol.plugins.screenpainter.IscobolScreenPainter.builderApplication [clean]  
[build] [generate] [logfile logfilename]
```

- *workspaceLocation* is the workspace directory.
- **-nosplash** is suggested to avoid the splash screen.
- **--launcher.suppressErrors** is suggested to avoid interactive message boxes on error. If this option is used, then errors are saved to the file ".log" in the workspace ".metadata" folder.
- **clean**, if specified, performs the same operation of *Project -> Clean* for each project in the workspace.
- **generate**, if specified, performs the same operation of *Project -> Generate Project* for each project in the workspace.
- **build**, if specified, performs the same operation of *Project -> Build Project* for each project in the workspace.
- **logfile** allows to choose the name of the file where to print the logs. If not specified, the IDE will create a file named "iscobol_builder<yyyyMMddHHmmss>.log" in the working directory.

Note - The 'build' option implies 'generate', so if you specify both 'build' and 'generate', 'generate' will be ignored.

Import AcuBench/Totem programs (PSF and CPF)

Command:

```
isIDE -data workspaceLocation -nosplash --launcher.suppressErrors -application  
com.iscobol.plugins.screenPainter.IscobolScreenPainter.importApplication [project  
projectName] [file programFile or folder folder] [cellWidthFact cell-width]  
[cellHeightFact cell-height] [leftText left-text-val] [logfile logfilename]
```

- *workspaceLocation* is the workspace directory.
- **-nosplash** is suggested to avoid the splash screen.
- **--launcher.suppressErrors** is suggested to avoid interactive message boxes on error. If this option is used, then errors are saved to the file ".log" in the workspace ".metadata" folder.
- *projectName* is the project where the imported program should be added. If the **project** option is omitted, the first project in the workspace is used.
- *programFile* is the name of a PSF or CPF file. Use **file** instead of **folder** if you wish to import a single program.
- *folder* is a directory where PSF and CPF files can be found. All PSF and CPF files will be imported. Use **folder** instead of **file** if you wish to import multiple programs at once.
- **cellWidthFact**, **cellHeightFact** and **leftText** allow you to set the corresponding advanced options that are available in the import wizard; see [Importing programs from AcuBench](#) and [Importing programs from Totem](#) for details.
- **logfile** allows to choose the name of the file where to print the logs. If not specified, the IDE will create a file named "iscobel_builder<yyyyMMddHHmmss>.log" in the working directory.

Import AcuBench/Totem file layout (DLT and CLF)

Command:

```
isIDE -data workspaceLocation -nosplash --launcher.suppressErrors -application  
com.iscobol.plugins.screenPainter.IscobolScreenPainter.importDltApplication  
[project projectName] [file dltFile or folder folder] [logfile logfilename]
```

- *workspaceLocation* is the workspace directory.
- **-nosplash** is suggested to avoid the splash screen.
- **--launcher.suppressErrors** is suggested to avoid interactive message boxes on error. If this option is used, then errors are saved to the file ".log" in the workspace ".metadata" folder.
- *projectName* is the project where the imported file layout should be added. If the **project** option is omitted, the first project in the workspace is used.
- *dltFile* is the name of a DLT or CLF file. Use **file** instead of **folder** if you wish to import a single file layout.
- *folder* is a directory where DLT and CLF files can be found. All DLT and CLF files will be imported. Use **folder** instead of **file** if you wish to import multiple files at once.
- **logfile** allows to choose the name of the file where to print the logs. If not specified, the IDE will create a file named "iscobel_builder<yyyyMMddHHmmss>.log" in the working directory.

Import SL/FD copy files

Command:

```
isIDE -data workspaceLocation -nosplash --launcher.suppressErrors -application  
com.iscobol.plugins.screenPainter.IscobolScreenPainter.importFDSDLApplication  
[project projectName] [file copyFile or folder folder] [fdsuffix suffix] [slsuffix  
suffix] [logfile logfilename]
```

- *workspaceLocation* is the workspace directory.
- **-nosplash** is suggested to avoid the splash screen.
- **--launcher.suppressErrors** is suggested to avoid interactive message boxes on error. If this option is used, then errors are saved to the file ".log" in the workspace ".metadata" folder.
- *projectName* is the project where the imported file layout should be added. If the **project** option is omitted, the first project in the workspace is used.
- *copyFile* is the name of a FD copy file. Use **file** instead of **folder** if you wish to import a single FD copy file.
- *folder* is a directory where FD and SL copy files can be found. All FD and SL copy files will be imported. Use **folder** instead of **file** if you wish to import multiple files at once.
- **fdsuffix** and **slsuffix** allow you to specify custom extensions. By default ".fd" and ".sl" are used.
- **logfile** allows to choose the name of the file where to print the logs. If not specified, the IDE will create a file named "iscobol_builder<yyyyMMddHHmmss>.log" in the working directory.

Import COBOL program

Command:

```
isIDE -data workspaceLocation -nosplash --launcher.suppressErrors -application  
com.iscobol.screenPainter.IscobolScreenPainter.importCobolProgramApplication  
[project projectName] [program programSource or folder folder] [suffix  
suffix] [logfile logfilename]
```

- *workspaceLocation* is the workspace directory.
- **-nosplash** is suggested to avoid the splash screen.
- **--launcher.suppressErrors** is suggested to avoid interactive message boxes on error. If this option is used, then errors are saved to the file ".log" in the workspace ".metadata" folder.
- *projectName* is the project where the imported file layout should be added. If the **project** option is omitted, the first project in the workspace is used.
- *programSource* is the name of a COBOL source file. Use **program** instead of **folder** if you wish to import a single source file.
- *folder* is a directory where COBOL source files can be found. All source files will be imported. Use **folder** instead of **program** if you wish to import multiple files at once.
- **suffix** allows to specify a custom extension. By default ".cbl" is used.
- **logfile** allows to choose the name of the file where to print the logs. If not specified, the IDE will create a file named "iscobol_builder<yyyyMMddHHmmss>.log" in the working directory.