

Bilder einfügen

 Sprache: **Deutsch** · English

Bilder einfügen

Bilder in Reports/Formulare einbinden – als Ressource oder dynamisch aus der Datenbank per InputStream.

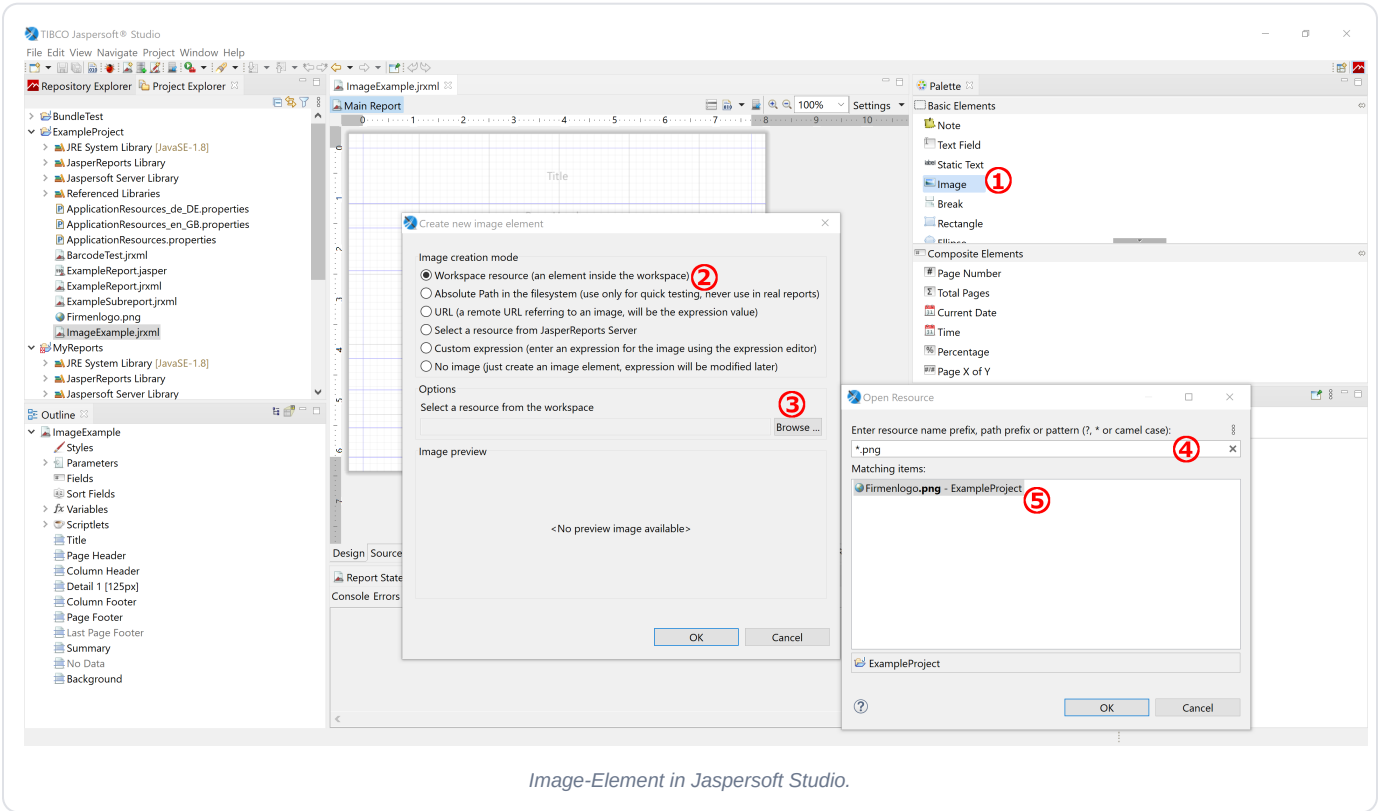
- HOW-TO
- ANWENDUNGSENTWICKLER
- STAND: JUL 2026
- GILT FUER NUCLOS 4.2026.X

Auf dieser Seite

- [Bild ins Formular einfügen](#)
- [Wichtige Eigenschaften](#)
- [Bild in Nuclos bereitstellen](#)
- [Bilder aus den Reportdaten \(dynamisch\)](#)
- [Verwandte Seiten](#)

Bild ins Formular einfügen

JasperReports akzeptiert **GIF, PNG und JPEG** sowie Vektorgrafiken als **SVG** (ohne eingebettete Rastergrafiken). Ziehe das *Image*-Werkzeug per Drag & Drop auf die Fläche und wähle die Datei (Filter ggf. auf „Alle Dateien“ stellen).



Wichtige Eigenschaften

Property	Werte
Scale Image	<i>Retain Shape</i> (Seitenverhältnis, Standard) · <i>Fill Frame</i> (füllt Fläche, ggf. verzerrt).
On Error Type	<i>Error</i> (Fehler werfen, Standard) · <i>Blank</i> (nichts) · <i>Icon</i> (Platzhalter).

Rotation	None · Left (90°) · Right (90°) · UpsideDown (180°).
Image Alignment	horizontale/vertikale Ausrichtung.

Bild in Nuclos bereitstellen

Nuclos bindet das Bild zur Laufzeit anhand des Datei-/Imagennamens ein. Lade es unter **Konfiguration** **Sonstige** **Ressource** hoch.



Nicht vergessen

Die neue Ressource muss dem **Nuclet zugewiesen** werden – sonst meldet der Report, dass das Bild nicht gefunden wird.

Bilder aus den Reportdaten (dynamisch)

Um Bilder aus der Datenbank anzuzeigen:

1. Feld anlegen, das auf das Bild verweist (z.B. User_Photo).
2. Feld-**Class** auf `java.io.InputStream` setzen.
3. Im **Image**-Tab den Expression Editor öffnen und das Feld als **Image Expression** eintragen.

Expression Editor (java)

Expression Editor

Edit the JasperReports expression that returns the expected type.

Parameters

Fields

Variables

Resources

Built-in Functions

Date & Time
Logical
Message Bundles
Numeric / Math
Report
Text
User Defined Expressions
Recent Expressions

type filter text

User_Photo Field InputStream

<

>

type filter text

available() int

equals(Object) boolean

getClass() Class

hashCode() int

markSupported() boolean

nullInputStream() java.io.InputStream

read(byte[]) int

read(byte[], int, int) int

read() int

readAllBytes() byte[]

readNBytes(byte[], int, int) int

readNBytes(int) byte[]

skip(long) long

toString() String

Value Class (backward compatibility only)

?

Finish

Cancel

Verwandte Seiten

Formulare mit Jaspersoft Studio erstellen

Formulare.

[Öffnen](#)

Ressource

Ressourcen.

[Öffnen](#)

QRCode

Barcodes.

[Öffnen](#)