



# XPress Tags 9.0 Benutzerhandbuch

# Contenido

<b>Zum Verständnis von XPress Tags.....</b>	<b>4</b>
Importieren von mit Tags versehenem Text.....	4
Erstellen von mit Tags versehenem Text in einer anderen Anwendung.....	4
Allgemeine Informationen zu XPress Tags.....	5
<b>Zeichen- und Absatzformatierung.....</b>	<b>6</b>
<b>Zeichenattribute.....</b>	<b>6</b>
Sprachen.....	7
Rubi Text.....	8
Zeichen gruppieren.....	9
Schriftengruppen.....	10
OpenType Attribute.....	10
<b>Absatzattribute.....</b>	<b>12</b>
<b>Andere Attribute.....</b>	<b>15</b>
<b>Ein-/Ausschalten von Gestaltungsmitteln.....</b>	<b>15</b>
<b>Zurückkehren zur Gestaltung mit Stilvorlagen.....</b>	<b>15</b>
<b>Stilvorlagen.....</b>	<b>16</b>
<b>Definieren von Stilvorlagen.....</b>	<b>16</b>
<b>Anwenden von Stilvorlagen.....</b>	<b>17</b>
Anwenden von Absatzstilvorlagen.....	17
Anwenden einer Zeichenstilvorlage.....	17
<b>Weitere XPress Tags für Stilvorlagen.....</b>	<b>18</b>
<b>Farben.....</b>	<b>19</b>
<b>Definieren einer individuellen Farbe.....</b>	<b>19</b>
Detaillierte Definition anwenderdefinierter Farben.....	19
<b>Anwenden einer individuellen Farbe.....</b>	<b>20</b>
<b>Sonderzeichen.....</b>	<b>21</b>
<b>Kodierung.....</b>	<b>21</b>
<b>Escaped-Zeichen.....</b>	<b>21</b>
<b>Andere Sonderzeichen.....</b>	<b>22</b>
<b>ASCII-Codes.....</b>	<b>22</b>
<b>Indexierung.....</b>	<b>24</b>
<b>Markieren von Text für einen Index.....</b>	<b>24</b>

**Definieren von Indextag-Daten.....24**

**Rechtliche Hinweise.....26**

# Zum Verständnis von XPress Tags

Mit der XPress Tags Filter XTensions® Software können Sie Plain-Text-Dateien verwenden, um Text aus einem QuarkXPress® Layout exportieren bzw. in das Layout importieren, einschließlich aller Absatz- und Zeichenattribute. Da das Plain-Text-Format an sich Attribute nicht unterstützt, wird das Format mit Codes angegeben, die der XPress Tags Filter umwandeln kann. Wenn die Optionen für Zeichen- und Absatzattribute von QuarkXPress erweitert werden, werden auch neue XPress Tags hinzugefügt, die diese Attribute unterstützen.

## Importieren von mit Tags versehenem Text

So importieren Sie eine Plain-Text-Datei, die eingebettete XPress Tags Codes enthält:

- 1 Wählen Sie **Datei > Importieren**.
- 2 Wählen Sie die gewünschte XPress Tags Datei aus.
- 3 Die Anwendung versucht, die Kodierung der XPress Tags Datei zu ermitteln. Wählen Sie zur Definition einer anderen Kodierung eine Option aus dem Dropdown-Menü **Kodierung**.
- 4 Aktivieren Sie in der Dialogbox **Text importieren** die Option **XPress Tags interpretieren**, um die XPress Tags Codes in Zeichenattribute und Absatzformate umzuwandeln. Falls Sie in der Dialogbox **Text importieren** die Option **XPress Tags interpretieren** vor dem Import von Text mit XPress Tags nicht aktivieren, können die Tags nicht von QuarkXPress umgewandelt werden. Statt dessen werden die Codes vom Programm als Textzeichen importiert.

## Erstellen von mit Tags versehenem Text in einer anderen Anwendung

Um XPress Tags Informationen in eine Textdatei aufzunehmen, die in einer anderen Anwendung erstellt wurde, stellen Sie dem Text die gewünschten Codes voran. Beginnen Sie die Textdatei mit einem Versions-Code und Kodierungsdaten (Beispiel: `<v8.00><e9>`). Weitere Informationen zum Kodieren finden Sie unter "[Kodierung](#)".

Sichern Sie den erstellten Text im Plain-Text-Dateiformat unter Verwendung der entsprechenden Kodierung.

## Allgemeine Informationen zu XPress Tags

Wenn Sie XPress Tag-Codes eingeben, beachten Sie bitte folgendes:

- Bei XPress Tags wird zwischen Groß- und Kleinschreibung unterschieden.
- XPress Tags für Zeichen- und Absatzattribute steht zwischen spitzen Klammern(< >). Das XPress Tag für Fettschrift lautet beispielsweise <B>.
- Für Codekombinationen geben Sie zunächst das „Kleiner-als“-Zeichen ein, dann die entsprechenden Codes und zum Schluss das „Größer-als“-Zeichen. Das XPress Tag für fett und kursiv lautet beispielsweise <BI>.
- XPress Tags für Zeichenattribute müssen unmittelbar vor den Zeichen stehen, auf die die Attribute angewendet werden sollen. Das mit dem XPress Tag Code aufgerufene Zeichenattribut bleibt solange wirksam, bis es deaktiviert wird oder bis durch die Eingabe anderer Codes die Stilvorlage geändert wird. Zum Deaktivieren eines Attributs definieren Sie den Code nach dem letzten Zeichen neu, für das das Attribut Gültigkeit haben soll (für einen Zeichenstil kann dies mit dem Code für Normaltext <P> geschehen).
- Bei XPress Tags, mit denen Sie mehr als einen Wert festlegen können (z. B. Absatzattribute), können Sie ein \$-Zeichen anstelle des tatsächlichen Werts eingeben. Wenn QuarkXPress auf einen \$-Code stößt, wird dieser durch den Wert ersetzt, der in der momentan angewendeten Stilvorlage festgelegt ist. (Wenn keine Stilvorlage aktiviert ist, wird ein Wert aus der Stilvorlage Normal verwendet.) Beispiel: Sie möchten, dass alle Formate der aktuell ausgewählten Stilvorlage mit Ausnahme des Zeilenabstands auf den Absatz angewendet werden. Dieser soll 18 Punkte und nicht den in der Stilvorlage angegebenen Wert betragen. Der Code sieht dann so aus:  
<\*p(\$,\$,\$,18,\$,\$,\$)>.
- Namen, die als XPress Tag Codes definiert werden, müssen von Zollzeichen " umschlossen sein. Soll z. B. die Schrift Palatino festgelegt werden, lautet der Code <f"Palatino">.
- Wenn eine Schrift mit XPress Tags definiert wird, kann der Schriftname im Code als Abkürzung erscheinen, wie z. B. helv für Helvetica. Wenn QuarkXPress einem importierten Text eine Schrift auf der Grundlage von XPress Tags zuweist, wählt das Programm die erste Schrift im Untermenü **Schrift**, die mit dem abgekürzten Schriftnamen beginnt.
- Einige Funktionen (wie Rubi Text, Gruppenzeichen und Betonungszeichen) sind nur in bestimmten Sprachversionen von QuarkXPress verfügbar. Sie können jedoch ein Projekt, in dem solche Funktionen verwendet werden, mit jeder Edition von QuarkXPress öffnen und Text, in dem solche Funktionen im XPress Tags Format verwendet werden, mit jeder Sprachenedition von QuarkXPress importieren und exportieren.

# Zeichen- und Absatzformatierung

Beim Definieren des Schriftstils mit Hilfe von XPress Tags wird durch den Code `<P>` stets der Schriftstil **Normal** zugeordnet. Beim Festlegen anderer Schriftstile (z. B. `<B>` für **Fett**) wird der jeweilige Stil aktiviert, wenn er noch nicht gesetzt wurde, und wieder deaktiviert, wenn er schon gesetzt wurde. Beispiel: Beim erstmaligen Festlegen von `<B>` wird dem folgenden Text der Schriftstil **Fett** zugewiesen. Beim nochmaligen Aufrufen des Codes `<B>` wird dem folgenden Text der Schriftstil **Fett** nicht mehr zugewiesen. Beim Aufrufen des Codes `<§>` wird dem folgenden Text der Schriftstil zugewiesen, der in der aktuellen Stilvorlage festgelegt ist. Ist aktuell keine Stilvorlage aktiv, wird die Stilvorlage **Normal** verwendet.

### Zeichenattribute

- Normal: `<P>`
- Fett: `<B>`
- Kursiv: `<I>`
- Konturiert: `<O>`
- Schattiert: `<S>`
- Unterstrichen: `<U>`
- Wort unterstrichen: `<W>`
- Durchgestrichen: `</>`
- Doppelte Durchstreichung: `<R>`
- Versalien: `<K>`
- Kapitälchen: `<H>`
- Hochgestellt: `<+>`
- Tiefgestellt: `<->` (Trennstrich)
- Index: `<V>`
- Ändern der Schrift\*: `<f "Schriftname">`
- Ändern der Schriftgröße\*: `<z###.##>` in Punkt

- Farbe ändern\*: <c"Farbname"> oder <cC, cM, cY, cK und cW>
- Ändern des Tonwerts\*: <s###. #> in Prozent des Tonwerts
- Schriftbreite\*: <h###. #> in Prozent der Schriftbreite
- Unterschneidung\*: <k###. ##> in 1/200 eines Geviertleerzeichens
- Spationieren\*: <t###. ##> in 1/200 eines Geviertleerzeichens
- Senden†: Wie Spationieren, jedoch mit einem Wert in Punkt am Ende
- Grundlinienversatz einstellen\*: <b###. ##> in Punkt
- Schrifthöhe\*: <y###. #> in Prozent der Schrifthöhe
- Ligaturen: <G1> zum Einschalten oder <G0> zum Ausschalten
- Deckkraft\*: <p###. #> in Prozent der Opazität
- OpenType: <o( "xxxx" )>, wobei "xxxx" = Code der OpenType®-Eigenschaft
- Sprache: <n##> (siehe "*Sprachen*")
- Zeichenausrichtung†: <\*An>, wobei n den Typ der Ausrichtung angibt; <AT> = Geviertbox oben/rechts, <AO> = ICF oben/rechts, <AC> = Mitte, <AM> = ICF unten/links, <AB> = Geviertbox unten/links, <AL> = Grundlinie†
- Zeichen halber Breite aufrecht halten† (ein nur in vertikalem Text verwendetes Zeichenattribut): <Ln>, wobei <L0> = seitwärts, <L1> = aufrecht und <L\$> = zurück zur Stilvorlage
- Betonungszeichen†: <Mn>, wobei n das Betonungszeichen ist
- Senden anwenden an Nicht-CJK-Zeichen†: <Y1> zum Einschalten, <Y0> zum Abschalten und <Y\$>, wenn die aktuelle Stilvorlageneinstellung verwendet werden soll

\*Wenn auf einen dieser Befehle ein \$ folgt (wie z. B. <f\$>), werden für die Attribute die Werte verwendet, die in der aktuellen Stilvorlage gesetzt sind. Ist aktuell keine Zeichenstilvorlage aktiv, wird die Stilvorlage **Normal** verwendet. (Das in der Liste verwendete Zeichen # kennzeichnet einen numerischen Wert.)

†Diese Funktion steht nur in bestimmten Spracheneditionen von QuarkXPress zur Verfügung. Sie können jedoch mit jeder Sprachenedition von QuarkXPress Text importieren und exportieren, der diese Funktionen im XPress Tags Format enthält.

## Sprachen

Folgende Sprachen lassen mithilfe des Tags <n##> anwenden.

- Keine: <n254>
- Bulgarisch: <n72>
- Katalanisch: <n73>
- Chinesisch (Vereinfacht): <n52>
- Chinesisch (Traditionell): <n53>
- Kroatisch: <n68>
- Tschechisch: <n56>
- Dänisch: <n9>

## ZEICHEN- UND ABSATZFORMATIERUNG

- Holländisch: <n5>
- Englisch (International): <n2>
- Englisch (US): <n0>
- Estnisch: <n44>
- Finnisch: <n17>
- Französisch: <n1>
- Deutsch: <n3>
- Deutsch (neue Rechtschreibung): <n70>
- Deutsch (Schweiz, reformiert): <n69>
- Deutsch (Schweiz): <n19>
- Griechisch: <n20>
- Ungarisch: <n43>
- Isländisch: <n21>
- Italienisch: <n4>
- Japanisch: <n14>
- Koreanisch: <n51>
- Lettisch: <n45>
- Litauisch: <n41>
- Norwegisch (Bokmal): <n12>
- Norwegisch (Nynorsk): <n80>
- Polnisch: <n42>
- Portugiesisch (brasilianisch): <n71>
- Portugiesisch (europäisch): <n10>
- Rumänisch: <n39>
- Russisch: <n49>
- Slowakisch: <n57>
- Slovenisch: <n66>
- Spanisch: <n8>
- Schwedisch: <n7>
- Türkisch: <n24>
- Ukrainisch: <n62>

### Rubi Text

Rubi Text ermöglicht das Kommentieren von Basiszeichen mit kleineren Rubi-Zeichen. Diese Funktion steht nur in bestimmten Spracheneditionen von QuarkXPress zur Verfügung. Sie können jedoch mit jeder Sprachenedition von QuarkXPress Text importieren und exportieren, der diese Funktionen im XPress Tags Format enthält.



Tags für Rubi Text besitzen folgende Form:

```
<A(\#68Basistext\#36<@><Zeichenattribute>\#9Rubitext\#132,50,C,A,0,2,B,T)>
```

Der Inhalt dieser Tags ist folgender:

- `<A()>` umschließt das Tag.
- `/#68` und `/#36` markieren den Anfang und das Ende des Basistextes.
- `<@><Zeichenattribute>` erlaubt die Stilgebung für den Rubi Text (siehe "[Zeichenattribute](#)").
- `\#n` kennzeichnet den Anfang des Rubi Textes, dabei ist `n` = die Anzahl der Rubi-Zeichen (im obigen Beispiel 9).
- `\#132` kennzeichnet das Ende des Rubi Textes und den Anfang der Codes für die Rubi-Positionierung.
- `50` ist ein Prozentsatz der Basis-Schriftgröße.
- `C` ist die Ausrichtung, dabei ist `L` = links, `C` = Mitte, `R` = rechts, `J` = Blocksatz, `F` = erzwungener Blocksatz, `O` = 1–2–1 (JIS) Regel und `E` = gleichmäßiger Abstand.
- `A` ist die Grundausrichtung, dabei ist `A` = keine, `L` = links, `C` = Mitte, `R` = rechts, `J` = Blocksatz, `F` = erzwungener Blocksatz, `O` = 1–2–1 (JIS) Regel und `E` = gleichmäßiger Abstand.
- `0` ist der horizontale Versatz des Rubi in Punkt.
- `2` ist der Überhang, dabei ist `0` = keiner, `1` = bis zu einem Rubi-Zeichen, `2` = bis zu einem halben Rubi-Zeichen, `3` = bis zu einem Basis-Zeichen, `4` = bis zu einem halben Basis-Zeichen und `5` = unbeschränkt.
- `B` ist die Platzierung relativ zum Basistext, dabei ist `A` = oberhalb und `B` = unterhalb.
- `T` ist die automatische Ausrichtung, dabei ist `T` = wahr und `F` = falsch.

### Zeichen gruppieren

Die Gruppierung von Zeichen ermöglicht das Gruppieren horizontaler Zeichen in einem vertikalen Textverlauf. Diese Funktion steht nur in bestimmten Spracheneditionen von QuarkXPress zur Verfügung. Sie können jedoch mit jeder Sprachenedition von QuarkXPress Text importieren und exportieren, der diese Funktionen im XPress Tags Format enthält.

Tags für gruppierte Zeichen besitzen folgende Form:

```
<A(\#72ABC\#40<t-10h80>)>
```

Der Inhalt dieser Tags ist folgender:

- `<A()>` umschließt das Tag.
- `/#72` und `/#40` markieren den Anfang und das Ende der gruppierten Zeichen.
- `<t-10h80>` sind Anwendungen von Spationierung und horizontaler Skalierung (siehe "[Zeichenattribute](#)").

### Schriftengruppen

Mithilfe von Schriftengruppen definieren Sie eine Zusammenstellung von Schriften, die auf unterschiedliche Arten von Text angewendet werden.

Genau wie Stilvorlagendefinitionen werden auch die Definitionen von Schriftengruppen in XPress Tags Dateien gespeichert. Wenn Sie beispielsweise Text exportieren, der eine Schriftengruppe mit dem Namen "Schriftengruppe 1" verwendet, wird die Definition von Schriftengruppe 1 in Form eines Tags wie des folgenden am Anfang der XPress Tag Datei exportiert:

```
@Schriftengruppe 1=[F]<"HiraMinPro-W3",100,0,h100,"GrecoStd-DB",100,0,h100,"Corbel",100,0,h100,"HiraMinPro-W3",89.999,-5,h105,"Times-Roman",100,0,h100>
```

Im Text sehen Schriftengruppen-Tags folgendermaßen aus:

```
<h105z10.89b-0.6f"HiraMinPro-W3",("Fontset 1",12)>
```

Der Inhalt dieses Tags ist folgender:

- `h105` = horizontale oder vertikale Skalierung (siehe "[Zeichenattribute](#)").
- `z10.8` = Prozentsatz der aktuellen Schriftgröße
- `b-0.6` = Grundlinienversatz in Punkt
- `f"HiraMinPro-W3"` = Schrift
- `("Schriftengruppe 1", 12)` = Name der anzuwendenden Schriftengruppe und aktuelle Punktgröße der Schrift

➔ Diese Funktion steht nur in bestimmten Spracheneditionen von QuarkXPress zur Verfügung. Sie können jedoch mit jeder Sprachenedition von QuarkXPress Text importieren und exportieren, der diese Funktionen im XPress Tags Format enthält.

### OpenType Attribute

Das Ein-/Ausschalten der Stilgebung ähnelt der aktuell verwendeten Methode für Schriftstile. Eine Wiederholung eines Tags schaltet den aktuellen Status des Attributs um.

Ein Stern (\*) zeigt ein Glyphen-Attribut an. Das Format für die Anwendung von Glyphen-Attributen lautet `<DoO,F"zero",I0,f"ACaslonPro-Bold">0<oC>`, wobei `DoO` = öffnendes Tag, `F"zero"` = Funktion, `I0` = Glyphen-Index (`I0` = die erste Glyphe), `f"AcaslonPro-Bold"` = Schrift, `0` = Basiszeichen und `oC` = schließendes Tag.

### Grundausrüstung

- Nur Kapitalchen: `a2sc`
- Alternative Annotations-Formen\*: `nalt`
- Alternative Brüche\*: `afrc`
- Kontextbedingte Varianten: `calt`
- Bedingte Ligaturen: `dlig`
- Nenner: `dnom`
- Brüche: `frac`

- Hangul\*: `hngl`
- Historische Zeichenvarianten\*: `hist`
- Horizontale Kana-Varianten†: `hkna`
- Kursivzeichen†: `ital`
- Standardligaturen: `liga`
- Dividend: `numr`
- Ordinalzeichen: `ordn`
- Schmuckzeichen\*: `ornm`
- Proportionale Versalziffern: `plin`
- Proportionale Mediävalziffern: `pold`
- Ruby Notationsformen†: `ruby`
- Wissenschaftliche Tiefstellung: `sinf`
- Stilistische Zeichenvarianten\*: `salt`
- Kapitälchen: `smcp`
- Tiefgestellt: `subs`
- Hochgestellt: `supr`
- Zierschrift: `swsh`
- Titelschriftvarianten: `titl`
- Versalziffern für Tabellen: `tlin`
- Mediävalziffern für Tabellen: `told`
- Durchgestrichene Null\*: `zero`
- Vertikale Kana-Varianten†: `vkna`

\*Dies ist ein Glyphen-Attribut; siehe oben.

†Diese Funktion lässt sich nur in einer Sprachenedition von QuarkXPress anwenden, die ostasiatische Features unterstützt.

#### **Alternative Formen:**

- Formen für Experten\*: `expt`
- Hojo Kanji Formen\*: `hojo`
- JIS2004 Formen†: `jp04`
- JIS78 Formen†: `jp78`
- JIS83 Formen†: `jp83`
- JIS90 Formen†: `jp90`
- NLC Kanji Formen\*: `nick`
- Vereinfachte Formen†: `smp1`
- Traditionelle Formen†: `trad`

## ZEICHEN- UND ABSATZFORMATIERUNG

- Traditionelle Namens-Formen\*: `tnam`

\*Dies ist ein Glyphen-Attribut; siehe oben.

†Diese Funktion lässt sich nur in einer Sprachenedition von QuarkXPress anwenden, die ostasiatische Features unterstützt.

### Alternative Metrik:

- Volle Breite†: `fwid`
- Halbe Breite†: `hwid`
- Drittelbreite†: `twid`
- Viertelbreite†: `qwid`
- Proportionale Breite†: `pwid`
- Alternative vertikale Metrik†: `valt`
- Proportionale alternative Breite†: `palt`
- Proportionale alternative vertikale Metrik†: `vpal`
- Alternative halbe Breite†: `halt`
- Alternative vertikale halbe Metrik†: `vhal`

\*Dies ist ein Glyphen-Attribut; siehe oben.

†Diese Funktion lässt sich nur in einer Sprachenedition von QuarkXPress anwenden, die ostasiatische Features unterstützt.

## Absatzattribute

XPress Tags für Absatzattribute müssen am Absatzanfang stehen. Die mit XPress Tags festgelegten Formate bleiben solange gültig, bis am Anfang des nächsten Absatzes neue Werte definiert oder XPress Tags zum Ändern der Stilvorlage eingegeben werden.

- Links ausgerichteter Absatz: `<*L>`
- Zentrierter Absatz: `<*C>`
- Rechts ausgerichteter Absatz: `<*R>`
- Blocksatz: `<*J>`
- Erzwungener Blocksatz: `<*F>`
- Tabulatoren setzen\*: `<*t(##.#,#,"1 oder 2 Zeichen")>`. Die Werte in Klammern sind **Position** (in Punkt), **Ausrichtung** und **Füllzeichen**. Geben Sie für keine Füllzeichen den Wert 1 gefolgt von zwei Leerzeichen ein. Soll ein Füllzeichen verwendet werden, eine 1 gefolgt vom Füllzeichen eingeben (wobei das Füllzeichen zweimal eingeben werden muss). Sollen zwei Füllzeichen festgelegt werden, geben Sie eine 2 und die beiden unterschiedlichen Füllzeichen ein. Die Optionen für **Ausrichtung** lauten 0 für **Links**, 1 für **Zentriert**, 2 für **Rechts**, 4 für **Punkt**, 5 für **Komma** und ein beliebiges druckbares Zeichen (in Anführungszeichen) für **Ausrichten an**. **Hinweis:** Wenn der Text zur Stilvorlage **Normal** zurückkehrt, wird die Ausrichtung nicht auf die in der

Stilvorlage **Normal** festgelegte Ausrichtung zurückgesetzt. Vielmehr muss die neue Ausrichtung zusätzlich zur erneuten Zuweisung der Stilvorlage **Normal** spezifiziert werden. Alle Tab-Stopps für einen Absatz tauchen innerhalb der Klammern durch Komma getrennt auf.

- Festlegen der Absatzattribute\*: `<*p(##.##.##.##.##.##.##.##,G oder g(P, S))>`. Die Werte in den Klammern stehen für **Linker Einzug**, **Erste Zeile**, **Rechter Einzug**, **Zeilenabstand**, **Abstand vor**, **Abstand nach** und **Am Raster ausrichten**. Zum Beispiel würde `<*p(16,36,16,14,9,18,g)>` einen Text folgendermaßen formatieren: 16 pt **Linker Einzug**, 36 pt **Erste Zeile**, 16 pt **Rechter Einzug**, 14 pt **Zeilenabstand**, 9 pt **Abstand vor**, 18 pt **Abstand nach** und er würde nicht am Grundlinienraster ausgerichtet. Sie können die erste Zeile eines Absatzes als Prozentsatz eines Geviertleerzeichens definieren, indem Sie einen dezimalen Prozentwert an der Position **Erste Zeile** verwenden (z. B. `0.6` für 60%). An der Position **Am Raster ausrichten** ist `G` = am Grundlinienraster ausrichten, `g` = nicht am Grundlinienraster ausrichten und die beiden Werte in Klammern geben an, ob am Seitenraster (`P`) oder am Rahmenraster (`B`) ausgerichtet werden soll und wie die Zeichen auszurichten sind (`T` — oben/rechts, `C` = Mitte, `S` = Grundlinie, `B` = unten/links).
- Definition für Silbentrennung und Blocksatz `<*h"Spezifikation Name">`. Silbentrenn- und Blocksatz-Definitionen müssen bereits in dem QuarkXPress Projekt festgelegt sein, bevor Sie Text mit Tags importieren, die sich auf diese beziehen. Wenn Sie mit XPress Tags eine S&B-Definition festlegen und diese Gruppe nicht in der S&B-Liste des Projekts aufzufinden ist, verwendet QuarkXPress statt dessen die S&B-Gruppe Standard. Die maximale Länge für Stilvorlagennamen und Silbentrenn- und Blocksatzdefinitionen beträgt 63 Zeichen.
- Absatz mit Linie oben\*\*: `<*ra(##,##,"Farbname",##,##,##,## oder ##%)>`. Die Werte in Klammern sind **Stärke**, **Stil**, **"Farbname"**, **Tonwert**, **Linker Einzug**, **Rechter Einzug** und **Versatz**. Linienbreite und Einzüge werden in Punkt eingegeben. Wird der Buchstabe „T“ vor den Wert für den linken Einzug gesetzt, richtet sich die Einzugslänge nach der ersten Textzeile. Die Optionen für **Linienstil** lauten `0` für **Normal**, `1` für **Gepunktet**, `2` für **Gepunktet 2**, `3` für **Strich-Punkt**, `4` für **Alle Punkte**, `5` für **Doppel**, `6` für **Dünn-Dick**, `7` für **Dick-Dünn**, `8` für **Dünn-Dick-Dünn**, `9` für **Dick-Dünn-Dick** und `10` für **Dreifach**. Der **Tonwert** wird in Prozent angegeben Sie können den Versatzwert einer Linie in Punkt oder als Prozentsatz angeben (`##%`). Beispiel: `<*ra(4,5,"Blau",100,T12,12,50%)>` ergibt eine **Linie oben** 4 pt breit im Stil **Doppel**, 100 % **Blau**, 12 pt **Linker Einzug** und **Rechter Einzug** (ausgerichtet nach der ersten Textzeile) und einen **Versatz** von 50 %.
- Absatz mit Linie unten\*\*: `<*rb(##,##,"Farbname",##,##,##,## oder ##%)>`. Die Werte in Klammern sind **Stärke**, **Stil**, **"Farbname"**, **Tonwert**, **Linker Einzug**, **Rechter Einzug** und **Versatz**. Linienbreite und Einzüge werden in Punkt eingegeben. Wird der Buchstabe „T“ vor den Wert für den linken Einzug gesetzt, richtet sich die Einzugslänge nach der ersten Textzeile. Die Optionen für **Linienstil** lauten `0` für **Normal**, `1` für **Gepunktet**, `2` für **Gepunktet 2**, `3` für **Strich-Punkt**, `4` für **Alle Punkte**, `5` für **Doppel**, `6` für **Dünn-Dick**, `7` für **Dick-Dünn**, `8` für **Dünn-Dick-Dünn**, `9` für **Dick-Dünn-Dick** und `10` für **Dreifach**. Der **Tonwert** wird in Prozent angegeben Sie können den Versatzwert einer Linie in Punkt oder als Prozentsatz angeben (`##%`). Beispiel: `<*rb(2,0,C,50,6,18,30%)>` ergibt eine **Linie unten** mit einer Breite von 2 pt im Stil **Normal**, 50 % **Cyan**, 6 pt **Linker Einzug** und 18 pt **Rechter Einzug** und einen **Versatz** von 30 %.

- Hängende Initiale<sup>\*\*</sup>: `<*d(Zeichenzahl,Zeilenzahl)>`
- Mit nächstem Absatz zusammenhalten<sup>\*\*</sup>: `<*kn1>` oder `<*kn0>1` = mit nächstem ¶ zusammenhalten; 0 = nicht zusammenhalten
- Zusammenhalten<sup>\*\*</sup>: `<*ktA>` oder `<*kt(#,#)>` A = Alle; #, # = Start Zeilenzahl, Ende Zeilenzahl. Um zu den Einstellungen in der Stilvorlage **Normal** zurückzukehren, geben Sie `<*kt($)>` ein.
- Gruppen hängender Zeichen: `<*s"Name der Gruppe hängender Zeichen">`. Falls das Layout keine Gruppe hängender Zeichen mit diesem Namen besitzt, wird keine solche Gruppe angewendet. Wenn Sie jedoch später eine Gruppe hängender Zeichen mit diesem Namen hinzufügen, wird sie auf den Text angewendet.
- Aufzählungszeichenstil: `<*bb"Aufzählungszeichenstil-Name"(7.2,0,0)>`. Falls das Layout keinen Aufzählungszeichenstil dieses Namens enthält, wird kein Aufzählungszeichenstil angewendet. In der eingeklammerten Liste bedeutet die erste Zahl den Mindestabstand zwischen dem Aufzählungszeichen und dem Text. Die zweite Zahl ist der Wert für Beginnen bei und die dritte Zahl das Flag für Neu nummerieren (0 = falsch, 1=wahr). Beachten Sie, dass der letzte Wert für Aufzählungsstile keine Bedeutung besitzt.
- Nummerierungsstil: `<*bb"Nummerierungsstil-Name"(7.2,0,0)>`. Falls das Layout keinen Nummerierungsstil dieses Namens enthält, wird kein Nummerierungsstil angewendet. In der eingeklammerten Liste bedeutet die erste Zahl den Mindestabstand zwischen dem Aufzählungszeichen und dem Text. Die zweite Zahl ist der Wert für Beginnen bei und die dritte Zahl das Flag für Neu nummerieren (0 = falsch, 1=wahr).
- Gliederungsstil: `<*bo"Gliederungsstil-Name"(7.2,0,0)>`. Falls das Layout keinen Gliederungsstil dieses Namens enthält, wird kein Gliederungsstil angewendet. Die eingeklammerte Werteliste besitzt für Gliederungsstile keine Bedeutung. In der eingeklammerten Liste bedeutet die erste Zahl den Mindestabstand zwischen dem Aufzählungszeichen und dem Text. Die zweite Zahl ist der Wert für Beginnen bei und die dritte Zahl das Flag für Neu nummerieren (0 = falsch, 1=wahr). Beachten Sie, dass der letzte Wert für Aufzählungsstile keine Bedeutung besitzt.
- Einzugsstufe: `<*l1>`. Die niedrigste Stufe ist 10 (Stufe 1) und die höchste 18 (Stufe 9).

\*Wenn das Zeichen \$ einige oder alle Formatcodes ersetzt (z. B. `<*t$>`), werden die Werte der aktuellen Stilvorlage verwendet. Ist aktuell keine Stilvorlage aktiv, wird die Stilvorlage **Normal** verwendet. Alle numerischen Werte in diesen beiden Befehlen werden in Punkt ausgedrückt.

\*\*Formatcodes können einzeln oder in ihrer Gesamtheit durch das Zeichen \$ ersetzt werden, um die Definitionen der aktuellen Stilvorlage zu aktivieren, oder sie lassen sich durch das Zeichen 0 ersetzen, um keine Linien festzulegen (z. B. `<*ra$>` und `<*ra0>`).

†Diese Funktion steht nur in bestimmten Spracheneditionen von QuarkXPress zur Verfügung. Sie können jedoch mit jeder Sprachenedition von QuarkXPress Text importieren und exportieren, der diese Funktionen im XPress Tags Format enthält.

## Andere Attribute

In diesem Abschnitt werden Attribute aufgelistet, die nicht in andere Kategorien gehören.

- Glyph: `<DoO, F"zero", I0, f"ACaslonPro-Bold">0<oC>`, wobei `DoO` = öffnendes Tag, `F"zero"` = Funktion, `I0` = Index der alternativen Glyphen, `f"AcaslonPro-Bold"` = Schrift, `0` = Basiszeichen und `oC` = schließendes Tag.
- Hyperlink: `<A(3, "HYPB", \#002\#000\#000\#000)[2]>Verlinkter Text<A(3, "HYP\" \" , \#nnn\#xxx)[n]>`, dabei ist `<A(3, "HYPB", \#002\#000\#000\#000)[2]>` das öffnende Tag (ändert sich nie), `\#nnn` der Typ der Verlinkung (`\#000` = URL, `\#004` = Anker, `\#008` = Seite) und `\#xxx` zeigt die Erstellungsreihenfolge dieses Hyperlinks im Projekt an (`\#001` = als erstes erstellt, `\#002` = als zweites erstellt und so weiter).
- Transcodierungssequenzen: `<EX>Unicode Wert, Sprache, Legacy Codewert <EX>`. Für Sprachencodes, siehe "*Sprachen*".
- Nicht kodierte Glyphen: `<DO, gxxxx, f"Font"P> <DC>`, dabei ist `DO` = öffnendes Tag, `g` = Glyphen, `f` = Schrift, `P` = Stil (`P`, `B`, `I` oder `BI`) und `DC` = schließendes Tag

## Ein-/Ausschalten von Gestaltungsmitteln

- Einschalten der Eigenschaft "a": `<o("aaaa")>`
- Einschalten der Eigenschaften "x", "y" und "z": `<o("xxxx", "yyyy", "zzz")>`
- Einschalten der Eigenschaften "a" und "b", anschließendes Einschalten der Eigenschaft "c" und Ausschalten der Eigenschaft "b": `<o("aaaa", "bbbb")>beliebiger<o("cccc", "bbbb")> Text`
- Einschalten der Eigenschaften "a" und "y": `<o("xxxx", "aaaa", "xxxx", "yyyy")>`

## Zurückkehren zur Gestaltung mit Stilvorlagen

- So kehren Sie von den Eigenschaften "x" und "y" zu den Zeichenattributen in der aktuell zugewiesenen Absatzstilvorlage zurück: `<o($ "xxxx", "yyyy")>`
- So kehren Sie von den Eigenschaften "x" und "a" zu den Zeichenattributen in der aktuell zugewiesenen Zeichenstilvorlage zurück: `<o($$ "xxxx", "aaaa")>`
- So kehren Sie von allen OpenType-Eigenschaften zu den Zeichenattributen in der aktuell zugewiesenen Absatzstilvorlage zurück: `<o($)>`
- So kehren Sie von allen OpenType-Eigenschaften zu den Zeichenattributen in der aktuell zugewiesenen Zeichenstilvorlage zurück: `<o($$)>`

# Stilvorlagen

Es ist möglich, mit Hilfe von XPress Tags Stilvorlagen zuzuweisen und eine Beziehung zwischen Absatz- und Zeichenstilvorlagen herzustellen.

## Definieren von Stilvorlagen

Die Definitionen einer Stilvorlage können ausschließlich aus Absatzattributen, ausschließlich aus Zeichenattributen oder aus einer Kombination beider bestehen.

- Definieren von Absatzstilvorlagen mit Standard-Zeichenattributen:  
`@Stilvorlagename=[Sp" ", " "] <Absatz- und Zeichenattribute> (Harter Zeilenumbruch) Zum Beispiel: @Paragraph1=[Sp" ", " "]<*L*h"Standard" *kn0*kt0*ra0*rb0*d0*p(0,0,0,0,0,0,g(B,S)) PBs100t0h100z14k0b0c"Red"f"Times-Roman">`
- Zeichenstilvorlage definieren: `@Stilvorlagename=<Zeichenattribute>(Harter Zeilenumbruch) Zum Beispiel: @Char1=<Ps100t-3h100z10k0b0cK f"Palatino-Roman">`
- Definieren von Absatzstilvorlagen mit Zeichenstilvorlagen:  
`@Stilvorlagename=[Sp" ", " ", "Name der Zeichenstilvorlage"]<Absatzattribute>(Harter Zeilenumbruch) Zum Beispiel: @Paragraph1=[Sp" ", "Absatz1", "Zeichen1"] <*L*h"Standard" *kn0*kt0*ra0*rb0*d0*p(0,0,0,0,0,0,g(B,S))>`
- Eine Absatzstilvorlage auf eine andere aufbauen und Nächster Stil anwenden:  
`@Stilvorlagename=[Sp"basiert auf Name der Absatzstilvorlage", "Name der nächsten Absatzstilvorlage", "Name der Zeichenstilvorlage"]<Absatzattribute>(Harter Zeilenumbruch) Zum Beispiel: @Absatz2=[Sp"Absatz1", "Absatz3", "Zeichen1"]<*t(121,1, "1."227,1,"1 ")>`
- Eine Zeichenstilvorlage auf eine andere aufbauen:  
`@Stilvorlagename=[St" ", " ", " ", "basiert auf Name der Zeichenstilvorlage"] <Zeichenattribute>(Harter Zeilenumbruch) Zum Beispiel: @Zeichen2=[St" ", " ", " ", "Zeichen1"] <Pbf"ArialMT">`



## Anwenden von Stilvorlagen

Stilvorlagen werden mit dem Zeichen @ angewendet. Beim Anwenden einer Zeichenstilvorlage können alle Zeichenattribute auf die Vorgabenattribute der Zeichenstilvorlage gesetzt werden, indem man dem @ ein „x“ voranstellt. Dadurch werden eventuell vorhandene Aufhebungen von Zeichenattributen gelöscht. So weist z. B. der Code <x@\$> die Attribute der Zeichenstilvorlage **Normal** zu und setzt alle ggf. geänderten Zeichenattribute außer Kraft.

## Anwenden von Absatzstilvorlagen

- Absatzstilvorlage Normal anwenden: @\$:Absatztext
- Absatzstilvorlage Kein Stil anwenden: @:Absatztext
- Anwenden einer definierten Absatzstilvorlage: @Stilvorlagenname:Absatztext

Wenn einem Absatz die Stilvorlage **Normal** zugewiesen werden soll (deren Attribute im QuarkXPress-Projekt definiert sind), muß der Absatz mit dem Code @\$: beginnen verwendet wird.

Wenn Sie einem Absatz eine bestimmte Stilvorlage zuordnen wollen, beginnen Sie ihn mit dem Code @Stilvorlagenname: verwendet wird.

Um einem Absatz **Kein Stil** zuzuweisen, muß am Anfang des ersten Absatzes, der nicht mit Stilvorlagen assoziiert werden soll, der Code @: verwendet wird.

- ➔ Wenn eine Stilvorlage einem Absatz mit Hilfe von XPress Tags zugewiesen wird, bleibt diese Stilvorlage solange auch für die nachfolgenden Absätze gültig, bis eine andere Stilvorlage oder **Kein Stil** mit dem Code @: verwendet wird.

In Absätzen mit assoziierter Stilvorlage können Attribute auch für Zeichen (lokal) angewendet werden. Diese Attribute bleiben gültig, bis man sie widerruft oder eine andere Stilvorlage zugewiesen wird.

Wenn Sie Text importieren, dem im Projekt bereits enthaltene Stilvorlagennamen zugeordnet sind, wendet QuarkXPress automatisch die Zeichen- und Absatzattribute der bereits im Projekt vorhandenen Stilvorlagen an. Beim Importieren von Text mit Stilvorlagennamen in ein Projekt, in dem diese Stilvorlagennamen noch nicht existieren, fügt QuarkXPress die importierten Stilvorlagen automatisch in die Liste der Palette **Stilvorlagen** ein. Wenn die neue Stilvorlage im getaggtten Text nicht definiert ist, legt QuarkXPress in der neuen Stilvorlage die Zeichen- und Absatzattribute wie in der Stilvorlage **Normal** fest und fügt diese Absatzstilvorlage in die Stilvorlagenliste der Palette **Stilvorlagen** ein.

Folgende Zeichen sind für Stilvorlagennamen nicht zulässig: " : = @

## Anwenden einer Zeichenstilvorlage

- Zeichenstilvorlage Normal anwenden: <@\$>
- Anwenden der Zeichenstilvorlage des Absatzes: <@\$p>
- Zeichenstilvorlage Kein Stil anwenden: <@>
- Anwenderdefinierte Zeichenstilvorlage anwenden: <@Stilvorlagenname>

### Weitere XPress Tags für Stilvorlagen

Die mögliche Verbindung zwischen Zeichenstilvorlage und Absatzstilvorlage erfordert zusätzliche XPress Tags.

- Einstellen des Schriftstils nach den Zeichenattributen der zugewiesenen Absatzstilvorlage: `<$>`
- Einstellen des Schriftstils nach den Zeichenattributen der aktuell zugewiesenen Zeichenstilvorlage: `<$ $>`
- Einstellen aller Zeichenattribute nach den Zeichenattributen der zugewiesenen Absatzstilvorlage: `<a.$>`

➡ Dieser Befehl wendet keine Zeichenstilvorlage an.

- Einstellen aller Zeichenattribute nach den Zeichenattributen der zugewiesenen Zeichenstilvorlage: `<a.$ $>`

# Farben

Eine Nicht-Prozessfarbe muss exakt definiert werden, damit sie beim Importieren korrekt interpretiert werden kann. Für diese Farben wird an den Anfang der XPress Tag Datei ähnlich wie für Stilvorlagen eine Farbdefinition platziert.

## Definieren einer individuellen Farbe

```
@Farbenname=[C]<"Farbenklasse",Farbenspezifikation>
```

Wobei folgendes gilt:

**Farbenname** = Bezeichnung der Farbe

**[C]** = bezeichnet eine anwenderdefinierte Farbe

```
colorspec = "Bibliotheksname",S oder P,#,"Farbunterklasse"
#.#,#.#,#.#,#.#,#.#,#.#
```

**S** oder **P** = "S" kennzeichnet eine Volltonfarbe, "P" kennzeichnet eine Prozessfarbe

Falls es sich um eine Volltonfarbe ("S") handelt, folgt eine Ziffer, die den verwendeten Rasterwert kennzeichnet, # = {1=Cyan, 2=Magenta, 3=Gelb, 4=Schwarz}

Zum Beispiel: @CMYK-M50Y100spotY=[C]<"CMYK",S,3,0,0.5,1,0>

**Bibliotheksname** = der Farbname aus dem Farbmusterbuch oder der Bibliothek (Kurzname)

**Farbunterklasse** = ähnlich wie die Farbklasse. Mögliche Werte: "CMYK", "LAB", "Hexachrome" und "RGB". Diese Backup-Definition wird benutzt, wenn eine Bibliothek fehlt.

**#.#,#.#,#.#,#.#,#.#,#.#** = Numerische Farbspezifikationen

## Detaillierte Definition anwenderdefinierter Farben

**Farbenklasse** = {"CMYK", "RGB", "HSB", "LAB", "DIC", "MULTI-INK", "FOCALTONE", "PANTONE@ ..." (es gibt 14 Pantone Optionen, z. B. "PANTONE@ solid coated"), "TOYO", "TRUMATCH", "Web Safe Colors", "Web Named Colors"}

Falls die **Farbenklasse** = "RGB", "HSB", "LAB", "Web Safe Colors" oder "Web Named Colors", dann

**Farbspezifikation** = S oder P, #, #.#,#.#,#.#

Beispiel: @Rot=[C]<"RGB",P,1,0,0>

Falls Farbklassse = "CMYK", dann

Farbspezifikation = S oder P,###,###,###,###

Beispiel: @CMYK-M50Y100spotY=[C]<"CMYK",S,3,0,0.5,1,0>

Falls Farbklassse = "DIC", "FOCALTONE", "PANTONE@...", "TOYO", "TRUMATCH", dann

Farbenspezifikation = "Bibliotheksname",S oder P,,"Farbenunterklasse",###,###,###,###,###,###

Beispiel 1: @DIC 399p spotB=[C]<"DIC", "DIC 399p",S,4,"CMYK",0.55,0.14,0.47,0>

Beispiel 2: @PANTONE 259 HexC=[C]<"PANTONE@ solid in hexachrome@ coated", "PANTONE 259 HC",S,4,"Hexachrome",0.4,1,0,0.25,0,0>

Falls Farbklassse = "MULTI-INK", dann

Farbspezifikation = "Farbname", "Farbklassse",I oder C,###,##

Wobei folgendes gilt:

I oder C = die Farbe ist der Alias-Name einer Druckfarbe oder eine anwenderdefinierte Farbe

I = Alias-Name der Druckfarbe

C = anwenderdefinierte Druckfarbe

### = der Prozentwert für Multi-Ink

# = der Index der Druckfarbe Dieser Wert wird nur für Komponenten von benannten Druckfarben aufgeschrieben.

Beispiel: @Mink-M70Lab20=[C]<"MULTI-INK", "Process Magenta", "CMYK", I,0.7,1"LAB-L50A45B-75spotB", "LAB", C,0.2>

### Anwenden einer individuellen Farbe

Einmal definierte Farben können mit den normalen Farb-Tags zugewiesen werden.

Zum Beispiel: <c"Red">, or <c"Mink-M70Lab20">

# Sonderzeichen

XPress Tags ermöglichen die Definition von Zeichenkodierungen und das Einsetzen und Bearbeiten von Sonderzeichen wie bedingten Trennstrichen und geschützten Leerzeichen.

## Kodierung

Eine der folgenden drei Kennzeichnungen für erweiterten Zeichensatz wird automatisch an den Anfang einer XPress Tags-Datei gestellt, die Sie durch den Befehl **Text sichern** (Menü **Ablage/Datei**) erzeugen.

- Unicode (UTF-16): `<e8>`
- Unicode (UTF-8): `<e9>`
- Mac Roman (x-mac-roman): `<e0>`
- Windows Latin (windows-1252): `<e1>`
- Western (iso-8859-1): `<e2>`
- Japanese Win (windows-932-2000): `<e3>`
- Japanese Mac (x-mac-japanese): `<e21>`
- Korean Windows (MS codePage 949): `<e19>`
- Korean Mac (KSC5601): `<e20>`
- Traditional Chinese (BIG5): `<e6>`
- Simplified Chinese (GB2312): `<e7>`

## Escaped-Zeichen

Wenn Sie bestimmte Zeichen als Text verwenden möchten, die von XPress Tags normalerweise als Teil eines bestimmten Codes gelesen werden, sind die folgenden Sonderzeichen erforderlich:

- @: `<\@>`
- <: `<\<>`
- \: `<\\>`

### Andere Sonderzeichen

Einige Sonderzeichen haben eigene XPress Tags Codes. Verwenden Sie folgende Codes, um diese Zeichen zu definieren.

- Neue Zeile (Weicher Umbruch): `<\n>`
- Bedingter Zeilenumbruch: `<\d>`
- Trennstrich\*: `<\->`
- Einzug hier: `<\i>`
- Rechts eingezogener Tabulator: `<\t>`
- Standard-Leerzeichen\*: `<\s>`
- Halbgeviert-Leerzeichen\*: `<\e>`
- Interpunktionsabstand\*: `<\p>`
- Flexibles Leerzeichen\*: `<\f>`
- Geviertstrich\*: `<\_>`
- Halbgeviertstrich\*: `<\a>`
- Bedingter Trennstrich: `<\h>`
- Seitenzahl des vorherigen Textrahmens: `<\2>`
- Seitenzahl der aktuellen Seite `<\3>`
- Seitenzahl des nächsten Textrahmens: `<\4>`
- Neue Spalte: `<\c>`
- Neuer Rahmen: `<\b>`
- Geviert-Leerzeichen\*: `<\m>`
- 1/3-Geviert-Leerzeichen\*: `<\#>`
- 1/4-Geviert-Leerzeichen\*: `<\$>`
- 1/6-Geviert-Leerzeichen\*: `<\^>`
- Ziffern-Leerzeichen\*: `<\8>`
- 1/24-Geviert-Leerzeichen\*: `<\{>`
- 1/8-Geviert-Leerzeichen\*: `<\[>`
- Word Joiner\*: `<\j>`
- Ideographisches Leerzeichen\*: `<\o>`
- Bedingten Stilmarken: `<\r>`

\*Durch ein **!** vor einem beliebigen Befehl in dieser Gruppe wird das Leerzeichen oder der Trennstrich geschützt (z. B. `<\!m>` - geschütztes Geviert-Leerzeichen).

### ASCII-Codes

Einige Textverarbeitungsprogramme machen es unter Umständen erforderlich, für die Eingabe der XPress Tags für Sonderzeichen den dezimalen ASCII-Code zu verwenden.

Der XPress Tags Code für diese Zeichen lautet `<\#Dezimalwert>`. Das Zeichen # ist Teil des Codes. Der XPress Tags Code muß für ASCII drei Ziffern enthalten. Bei vierstelligen ASCII-Codes sollten Sie darauf achten, die vorangestellte Null nicht mit einzugeben. Hier eine Liste der dezimalen ASCII-Codes für einige häufig vorkommende Zeichen:

- Dezimaler ASCII-Code für ein Zeichen\*: `<\#Dezimalwert>`
- Neuer Absatz (Harter Umbruch): `<\#13>`
- Tabulator: `<\#9>`
- Halbgeviertstrich: `<\#208>`
- Geviertstrich: `<\#209>`
- Doppeltes Anführungszeichen unten: `<\#210>`
- Doppeltes Anführungszeichen oben: `<\#211>`
- Einfaches Anführungszeichen unten: `<\#212>`
- Einfaches Anführungszeichen oben (Apostroph): `<\#213>`

\*Das Voranstellen eines ! vor dem entsprechenden Befehl macht aus dem Zeichen ein geschütztes Zeichen.

# Indexierung

Sie können QuarkXPress Indextags im XPress Tags Format importieren und exportieren. Text kann vom Autor oder dem Ersteller eines Index mithilfe einer Textverarbeitungsanwendung getaggt und dann zu Layoutzwecken in QuarkXPress importiert werden. Alternativ kann Text mit Index-Tags zum Zwecke der Bearbeitung auch aus QuarkXPress exportiert werden.

## Markieren von Text für einen Index

Die XPress Tags zum Indexieren erlauben das Einfügen von Indexmarken am Texteneinfügekpunkt oder die Angabe eines zu indexierenden Textbereichs.

- So fügen Sie am Texteneinfügekpunkt eine Indexmarke ein: `<XC,Tag Info>`
- So geben Sie den Beginn eines indexierten Textbereichs an: `<XO>`
- So geben Sie das Ende eines indexierten Textbereichs an: `<XC,Tag Info>`

## Definieren von Indextag-Daten

Indextags enthalten Informationen zur Ebene, zum Stil und zum Geltungsbereich eines Eintrags.

- Eintrag auf der ersten Ebene: "Eintrag der ersten Ebene", "", Sortieren unter, Stilinformation, Bereich, Zusatzinformation, "Querverweistext" Zum Beispiel: `<XO>20. Jahrhundert<XC, "20. Jahrhundert", "", "Zwanzigstes Jahrhundert", "Indextext 1", 6, 1, "Zeitalter der Moderne">`
- Eintrag auf der zweiten Ebene: "Eintrag der ersten Ebene", "Eintrag der zweiten Ebene", "", Sortieren unter, Stilinformation, Bereich, Zusatzinformation, "Querverweistext" Zum Beispiel: `<XO>Geisteswissenschaften<XC, "20. Jahrhundert", "Geisteswissenschaften", "", "", "Indextext 2", 6, 1, "Künste">`
- Eintrag auf der dritten Ebene: "Eintrag der ersten Ebene", "Eintrag der zweiten Ebene", "Eintrag der dritten Ebene", "", Sortieren unter, Stilinformation, Bereich, Zusatzinformation, "Querverweistext" Zum Beispiel: `<XO>Literatur<XC, "20. Jahrhundert", "Geisteswissenschaften", "Literatur", "", "", "Indextext 3", 6, 1, "Bücher">`



- Eintrag auf der vierten Ebene: "Eintrag der ersten Ebene", "Eintrag der zweiten Ebene", "Eintrag der dritten Ebene", "Eintrag der vierten Ebene", "Sortieren unter", "Stilinformation", "Bereich", "Zusatzinformation", "Querverweistext" Zum Beispiel: <XO>Englisch<XC, "20. Jahrhundert", "Geisteswissenschaften", "Literatur", "Englisch", "", "Indextext 4", 6, 1, "Großbritannien">
- Stildaten: "Stilvorlagename" oder "" für Eintragsstil
- Gültigkeitsbereich: 0, 1, 2, 3, 4, 5 oder 6 (Auswahlbeginn = 0, Auswahltext = 1, Für Stil = 2, angegebene Anzahl Absätze = 3, Bis zum Ende von = 4, Seitenzahl unterdrücken = 5, Querverweis = Code)
- Sortieren als: "Sortieren als Text"
- Auswahlbeginn: 0
- Auswahltext: 1
- Für Stil: *Stilvorlagename*
- Festgelegte Anzahl von Absätzen: *Anzahl der Absätze*
- Seitenzahl unterdrücken: 0
- Bis zum Ende von: 0, 1 (Story = 0, Dokument = 1)
- Seitenzahl unterdrücken: 0
- Querverweis: 0, 1, 2 (Siehe = 0, Siehe auch = 1, Siehe hier = 2)

# Rechtliche Hinweise

© 1986-2022 Quark Software Inc. und seine Lizenzgeber. Alle Rechte vorbehalten.

Geschützt durch folgende Patente der Vereinigten Staaten: 5,541,991; 5,907,704; 6,005,560; 6,052,514; 6,081,262; 6,633,666 B2; 6,947,959 B1; 6,940,518 B2; 7,116,843; weitere Patente sind angemeldet.

Quark, das Quark Logo, Quark Publishing System und QPS sind Marken oder eingetragene Marken von Quark Software Inc. und seiner verbundenen Unternehmen in den Vereinigten Staaten und/oder anderen Ländern. Alle anderen Marken sind Eigentum ihrer jeweiligen Besitzer.