

# InDesign Satzautomation

Übersicht | JavaScript | InDesign Objektmodell | InDesign Programmierung | InDesign XML

---

## Offene Punkte

- **Bezugspunkt** für `geometricBounds`:  
`app.layoutWindows[0].transformReferencePoint = AnchorPoint.TOP_LEFT_ANCHOR`  
oder einfach im Template das für die Automatisierung verwendet wird umstellen.
- **Code Completion** leider nicht kontextsensitiv  
Unter CS4 stehen nur noch die Basisklassen von Javascript zur Verfügung.  
Unter CS3 war es etwas besser, aber letztlich auch unbrauchbar.

# InDesign Satzautomation

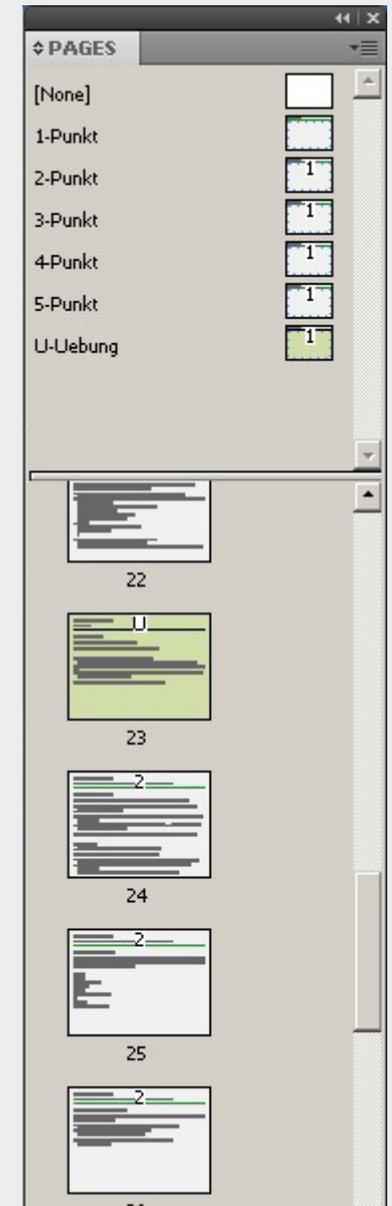
Übersicht | JavaScript | InDesign Objektmodell | InDesign Programmierung | InDesign XML

## Seiten und Musterseiten

Seiten **page** und Musterseiten **masterSpread** sind wichtige InDesign-Objekte!

Nochmal konkret: **pages** vs. **page**

- Mit der Sammlung (so zu behandeln wie ein Array) **pages** kann man:
  - Mit der Eigenschaft **length** die Länge der Sammlung ermitteln  
z. B. für eine for Schleife
  - Mit der Funktion **add()** Seiten hinzufügen
  - Mit **itemByName()** Seiten anhand ihrer Seitenzahl finden  
(Zählung kann auch bei > 1 beginnen)
  - Mit **itemByRange(FROM, TO)** einen Array der Seiten anfordern.
- Dem Seitenobjekt **page** kann man:
  - Mit der Eigenschaft **pageItems** alle Objekte auf der Seite abrufen
  - Mit **textFrames** die Sammlung der Textrahmen abrufen
  - Mit **allGraphics** alle Grafiken der Seite abrufen
  - Mit **appliedMaster** die angewendete Musterseite setzen/abrufen
  - Mit **side** herausfinden ob es sich um  
**PageSideOptions.RIGHT\_HAND**, **.LEFT\_HAND** oder **.SINGLE\_SIDED**
  - Mit der Funktion **place()** einen Text in den Musterseitenrahmen platzieren
  - Mit **move()** die Seite verschieben
  - Mit **remove()** die Seite entfernen



# InDesign Satzautomation

Übersicht | JavaScript | InDesign Objektmodell | InDesign Programmierung | InDesign XML

---

Das Konzept der **Musterseiten** bietet sich für automatisierten Satz an! In einem Template können beliebige Musterseiten angelegt werden, die dann für den Satz verwendet werden.

- Die Sammlung **masterSpreads** hat die üblichen Sammlungsobjekte.  
Mit der Funktion `itemByName()` kann man einfach auf die einzelnen Musterseiten zugreifen.  
`var _mspread = app.activeDocument.masterSpreads.itemByName("A-Master");`
- Musterseiten können mit der Eigenschaft `page.appliedMaster` einer Seite zugewiesen werden
- Alle Musterseitenobjekte sind in der Sammlung `masterPageItems`
- Musterseitenobjekte müssen auf der Seite **gelöst** werden:  
`masterPageItem.override(Seitenreferenz);`
- Nach dem Lösen von Objekten den **Textfluss aktualisieren**:  
`dokument.recompose();`
- **Smart Text Reflow** ab CS4  
Einstellung unter Eigenschaften -> Type



### Mit Seiten spielen

- ! Erstellen Sie ein neues Skript!
- ! Erstellen Sie **5 neue Seiten**.  
Im Objektmodell ist beschrieben wie mit dem Parametern `LocationOptions` und `reference` an verschiedenen Positionen im Dokument Seiten erstellt werden können. Machen Sie sich damit vertraut.
- ! Verwenden Sie die Funktion `itemByRange()` um nur die Seiten 3-5 mit Seitenzahlen per Skript zu nummerieren. Verwenden Sie die Eigenschaft `side` um die Seitenzahlen jeweils am äußeren Rand zu platzieren. Um den Array zu erhalten muss die Funktion wie folgt aufgerufen werden `itemByRange().getElements()`
- ! Verschieben und entfernen Sie Seiten im/aus dem Dokument!
- ! Legen Sie mehrere **Musterseiten** an. Weisen Sie verschiedenen Seiten unterschiedliche Musterseiten zu.
- ! Lösen Sie einen Musterseitentextrahmen, platzieren das Word-Dokument aus dem Ordner `03_material` und testen Sie die Funktion `Smart Text Reflow`.
- ! Auch eine Seite hat die Funktion `place()`. Was macht diese? Machen Sie sich mit den Parameter `[autoflowing: bool=false]` vertraut.

## Textrahmen und Stories – eine schwierige Verwandtschaft

textFrame- und Text-Objekte können zwei Eltern haben:

- **Text-Objekte** (paragraphs, words, characters, insertionPoints)  
Direktes Eltern-Objekt mit parent ist `story`  
`parentTextFrames[0]` führt zum Textrahmen (`textFrame`)
- **Textrahmen**  
Direktes Eltern-Objekt mit parent ist `page` (Die Seite, auf dem der Textrahmen platziert ist)  
`parentStory` führt zur `story` und damit zu den Textobjekten

## Textverkettungen

- Zwei Textrahmen können über die Eigenschaft `nextTextFrame` (bzw. auch `previousTextFrame`) miteinander **verknüpft** werden.  
`_tf1.nextTextFrame = _tf2;`

# InDesign Satzautomation

Übersicht | JavaScript | InDesign Objektmodell | InDesign Programmierung | InDesign XML

## Absatz, Zeichen und Einfügemarke

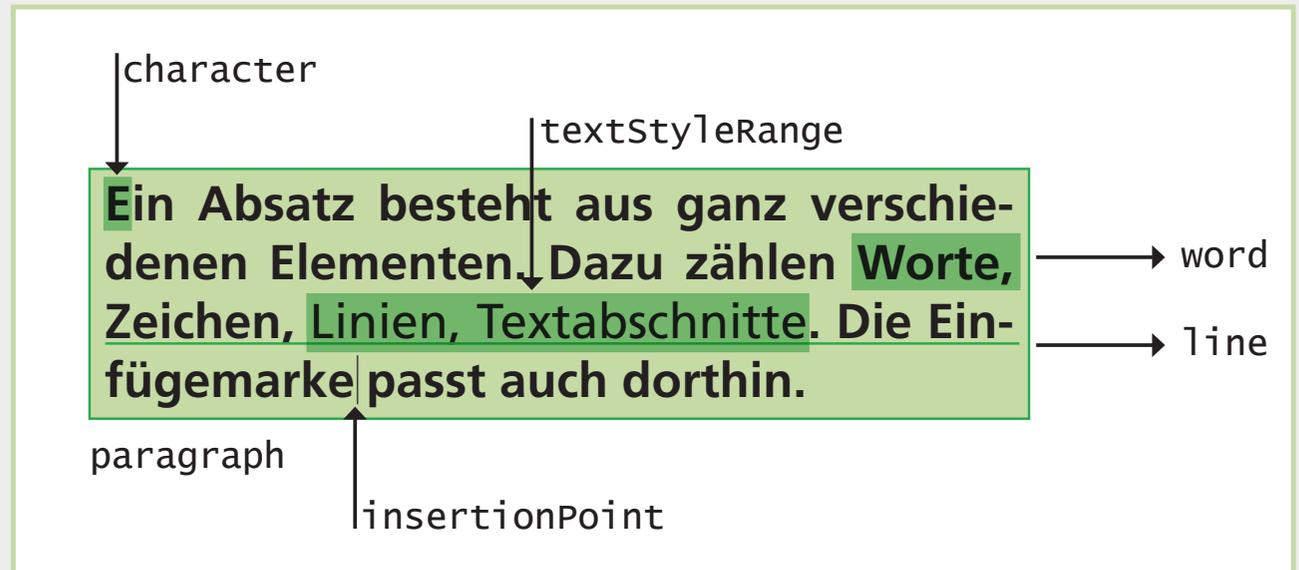
Textobjekte haben vielfältige Verwandtschaftsbeziehungen und Eigenschaften.

- Alle **Formatierungsoptionen** haben eine Entsprechung. z. B.: `appliedFont`, `appliedParagraphStyle`, `fontStyle`, `justification` (Ausrichtung), ...

- Textobjekte können mehr: **baseline vertikale Position**

`horizontalOffset` horizontale Position und `endHorizontalOffset`

- Mit der Eigenschaft `contents` kann der enthaltene Text abgefragt und neu gesetzt werden. `contents` liefert einen String des gesamten Inhalts aber nicht die Formatierung!
- Bereiche mit gleicher Formatierung `textStyleRanges`
- Die Sammlung `characters` bietet die Möglichkeit **Textbereiche** anzusteuern  
`characters.itemByRange(0,9).contents` // liefert die ersten 10 Zeichen



### Sätze formatieren

- Die Dateien befinden sich im Ordner `03_jsx_termin_02` und `03_indd`
- Öffnen Sie die Datei `texte.indd` in InDesign, öffnen Sie das Skript `textFormat.jsx`
- ! Der Inhalt des Beispieldokuments ist nicht formatiert. Im ersten Absatz stehen die Autoren des Texts, im zweiten die Überschrift. Legen Sie in InDesign zwei verschiedene passende **Absatzformate** an und weisen Sie diese den beiden Absätzen zu.
- ! Legen Sie für den Grundtext ein Absatzformat mit einem Einzug in der ersten Zeile an und weisen Sie dieses den restlichen Absätzen zu.
- ! Eine weitere automatische Formatierung von Überschriften ist schwer/nicht möglich – warum?
- ! **Zusatz:** Mit der Eigenschaft `appliedParagraphStyle` werden lokale Formatierungen eines Absatzes überschrieben. Finden Sie heraus, wie Sie lokale Formatierungen mit der Methode `applyStyle()` beibehalten können.
- ! Mit der Eigenschaft `baseline` kann die vertikale (y) Position der ersten Zeile des Absatzes ermittelt werden. Erstellen Sie neben dem zweiten Absatz ein quadratisches schwarzes Rechteck in Schriftgröße (`pointSize`) mit einem Abstand von 3 mm zur linken Satzspiegelkante.