

numerierte Überschriften

wer gerne mit numerierten Überschriften arbeitet, für den könnte diese beiden Code Blöcke hilfreich sein:

mit Prüfung auf einen Tag

Prüfung ob Tag auf einem Buch gesetzt ist

Mit diesem Javascript wird geprüft, ob ein bestimmtes Tag-Paar auf dem Buch gesetzt ist. Das Tag-Paar lautet **numbered-active**:

```
<script type="module">

async function isNumberTagPresent(documentToCheck) {
  const bookSlugRegex = /\books\/(.*)\/(page|draft)\//;
  const slugMatchResults = bookSlugRegex.exec(window.location.href);
  if (slugMatchResults === null) {
    return false;
  }

  const bookSlug = slugMatchResults[1];
  const bookResp = await window.$http.get(`/books/${bookSlug}`);
  const parser = new DOMParser();
  const pageDom = parser.parseFromString(bookResp.data, 'text/html');

  // Check for the tag in the provided document
  return pageDom.querySelector('body.tag-pair-numbered-active') !== null;
}

async function toggleCssClass() {
  const isRequired = await isNumberTagPresent(document);
  if (isRequired) {
    document.body.classList.add('numbered-active');
  } else {
    document.body.classList.remove('numbered-active');
  }
}
```

```

    }

    // Check if there are iframes in the document
    const iframes = document.querySelectorAll('iframe');
    for (const iframe of iframes) {
      if (iframe.contentDocument) {
        const isIframeRequired = await isNumberTagPresent(iframe.contentDocument);
        if (isIframeRequired) {
          iframe.contentDocument.body.classList.add('numbered-active');
        } else {
          iframe.contentDocument.body.classList.remove('numbered-active');
        }
      }
    }
  }
}

document.addEventListener("DOMContentLoaded", function() {
  toggleCssClass();
});
</script>

```

Ist der Tag gesetzt wird in die HTML "body"-Elemente eine Klasse geschrieben, die dann im folgenden CSS abgefragt wird.

CSS nur setzen wenn CSS-Klasse vorhanden

Hier wird nun die numerierte Liste erstellt, aber nur wenn im HTML "body"-Element die entsprechende Klasse vorher mittels Javascript reingeschrieben wurde.

```

<style>
  body.numbered-active, body.mce-content-body {
    counter-reset: h1counter;
  }
  body.numbered-active .page-content, body.numbered-active.mce-content-body {
    counter-reset: h2counter;
    counter-increment: h1counter;
  }

```

```
body.numbered-active .page-content h1:not(.break-text):before, body.numbered-active.mce-content-body
h1:not(.break-text):before {
    content: counter(h1counter) ".\0000a0\0000a0";
    counter-increment: h1counter;
}
body.numbered-active .page-content h1, body.numbered-active.mce-content-body h1 {
    counter-reset: h2counter;
}

body.numbered-active .page-content h2:before, body.numbered-active.mce-content-body h2:before {
    content: counter(h2counter) ".\0000a0\0000a0";
    counter-increment: h2counter;
}
body.numbered-active .page-content h2, body.numbered-active.mce-content-body h2 {
    counter-reset: h3counter;
}

body.numbered-active .page-content h3:before, body.numbered-active.mce-content-body h3:before {
    content: counter(h2counter) "." counter(h3counter) ".\0000a0\0000a0";
    counter-increment: h3counter;
}
body.numbered-active .page-content h3, body.numbered-active.mce-content-body h3 {
    counter-reset: h4counter;
}

body.numbered-active .page-content h4:before, body.numbered-active.mce-content-body h4:before {
    content: counter(h2counter) "." counter(h3counter) "." counter(h4counter) ".\0000a0\0000a0";
    counter-increment: h4counter;
}
body.numbered-active .page-content h4, body.numbered-active.mce-content-body h4 {
    counter-reset: h5counter;
}

body.numbered-active .page-content h5:before, body.numbered-active.mce-content-body h5:before {
    content: counter(h2counter) "." counter(h3counter) "." counter(h4counter) "." counter(h5counter)
".\0000a0\0000a0";
    counter-increment: h5counter;
}
</style>
```

Ohne Prüfung auf einen Tag (für alle Inhalte)

```
<style>
  body {
    counter-reset: h1counter;
  }
  body.page-content.mce-content-body {
    counter-reset: h2counter;
    counter-increment: h1counter;
  }
  .page-content h1:not(.break-text):before {
    content: counter(h1counter) ".\0000a0\0000a0";
    counter-increment: h1counter;
  }
  .page-content h1 {
    counter-reset: h2counter;
  }

  .page-content h2:before {
    content: counter(h2counter) ".\0000a0\0000a0";
    counter-increment: h2counter;
  }
  .page-content h2 {
    counter-reset: h3counter;
  }

  .page-content h3:before {
    content: counter(h2counter) "." counter(h3counter) ".\0000a0\0000a0";
    counter-increment: h3counter;
  }
  .page-content h3{
    counter-reset: h4counter;
  }

  .page-content h4:before {
    content: counter(h2counter) "." counter(h3counter) "." counter(h4counter) ".\0000a0\0000a0";
```

```
        counter-increment: h4counter;
    }
    .page-content h4 {
        counter-reset: h5counter;
    }

    .page-content h5:before {
        content: counter(h2counter) "." counter(h3counter) "." counter(h4counter) "." counter(h5counter)
        ".\0000a0\0000a0";
        counter-increment: h5counter;
    }
</style>
```

Revision #4

Created 23 June 2024 11:54:56 by Sascha Jelinek

Updated 21 August 2024 07:53:11 by Sascha Jelinek