Markdown 엣지케이스 가이드 [중첩리스트]

보고된 증상

ordered list 항목 아래에 unordered list를 중첩할 때, 탭 키 2번이상 또는 특정 스페이스바 간격 이상 들여쓰기하면 하위 리스트가 렌더링되지 않는다.

렌더링 실패 예시:

```
1. 은하철도
        * 999
3. 메텔
```

위와 같이 `* 999` 앞에 일정 스페이스이상 (또는 탭 1개 이상)이 들어가면, `* 999`가 리스트 항목이 아닌 일반 텍스트로 렌더링된다.

원인

해당 현상은 **CommonMark 스펙에 명시된 동작**으로 확인된다.

BookStack은 프론트엔드(에디터 프리뷰)에서 `markdown-it`, 서버사이드(저장/열람/PDF)에서 `league/commonmark`을 사용하며, 두 파서 모두 CommonMark 스펙을 따른다.

CommonMark 스펙의 List Items 규칙 (Section 5.2)

> **스펙 원문 (CommonMark Spec 0.31.2, Section 5.2 — List items):**
>
> "Basic case. If a sequence of lines Ls constitute a sequence of blocks Bs starting with a character other than a space or tab, and M is a list marker of width W followed by 1 ≤ N ≤ 4 spaces of indentation, then the result of prepending M and the following spaces to the first line of Ls, and indenting subsequent lines of Ls by **W + N** spaces, is a list item with Bs as its contents."

핵심은 **W + N** 공식이다.

- **W** = 리스트 마커의 폭 (예: `1.` → 2, `10.` → 3, `*` → 1, `-` → 1)
- **N** = 마커 뒤의 공백 수 (1~4칸)

중첩 리스트가 부모의 하위 항목으로 인식되려면, **정확히 W + N칸**만큼 들여쓰기해야 한다. 이보다 과도하게 들여쓰면 indented code block이나 paragraph continuation text로 해석된다.

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

실제 계산 예시

```
1. ㅅㅅㅅㅅ
^^^
¦¦¦
W=2 (리스트 마커 "1."의 폭)
 N=1 (마커 뒤 공백 1칸)
 → W + N = 3칸 들여쓰기 필요
```

따라서 중첩 리스트는 **3칸 스페이스**로 들여쓰기해야 한다:

```
1. ㅅㅅㅅㅅ
   * 999
3. ㅁㅁㅁ
```

`* 999`의 `*`이 `ㅅ`과 같은 열(4번째 열)에서 시작하는 것이 올바른 위치이다.

리스트 마커별 필요 들여쓰기

부모 리스트 마커	W (마커 폭)	N (뒤 공백)	필요 들여쓰기 (W+N)
`*`, `-`, `+`	1	1	**2칸**
`1.` ~ `9.`	2	1	**3칸**
`10.` ~ `99.`	3	1	**4칸**
`100.` ~ `999.`	4	1	**5칸**

과도한 들여쓰기 시 발생하는 문제

CommonMark 스펙에서 리스트 항목 내부의 indented code block은 **W + N + 4칸** 이상의 들여쓰기로 시작된다. 즉 `1.` 리스트 항목(W+N=3) 아래에서 7칸 이상 들여쓰기하면 코드 블록으로 해석될 수 있다.

또한 중첩 리스트 마커(`*`, `-`)가 부모 항목의 텍스트 시작 위치보다 훨씬 뒤에 있으면, 파서는 이를 리스트 마커로 인식하지 않고 일반 텍스트의 일부로 처리한다.

다른 플랫폼에서의 동작

이 동작은 BookStack만의 문제가 아니라 **CommonMark 스펙을 따르는 모든 마크다운 파서에서 동일하게 발생**한다.

플랫폼 / 파서	CommonMark 기반	동일 증상 발생
GitHub (GFM)	O	O
GitLab	O	O
markdown-it (BookStack 프론트엔드)	O	O

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

플랫폼 / 파서	CommonMark 기반	동일 증상 발생
league/commonmark (BookStack 서버사이드)	O	O
Hugo (Goldmark)	O	O
marked.js	O	O
Bitbucket	자체 구현	O (4칸 고정 필요)
Python-Markdown	자체 규칙	O (4칸 고정, 더 엄격)

CommonMark 스펙 Issue #399에서도 이 들여쓰기 규칙에 대한 사용자 혼란이 논의되었으나, 스펙 유지보수자들은 현재 규칙이 의도된 설계라는 입장이다.

Notion에서는 들여쓰기 동작이 코드블럭이 아닌경우 두번이상 원천적으로 불가능함.

올바른 작성법 가이드

기본 원칙

중첩 리스트의 마커(`*`, `-`, `1.` 등)는 부모 항목의 텍스트 시작 위치에 맞춰 들여쓰기한다.

예시 1: ordered list 안에 unordered list

```
1. 첫 번째 항목
   * 하위 항목 A
   * 하위 항목 B
2. 두 번째 항목
   - 하위 항목 C
```

`1.` 뒤 텍스트가 4번째 열에서 시작하므로, 하위 리스트도 4번째 열에서 시작한다 (3칸 들여쓰기).

예시 2: unordered list 안에 ordered list

```
- 상위 항목
  1. 하위 번호 항목
  2. 하위 번호 항목
- 다른 항목
```

`-` 뒤 텍스트가 3번째 열에서 시작하므로, 하위 리스트도 3번째 열에서 시작한다 (2칸 들여쓰기).

예시 3: 다단계 중첩

```
1. 1단계
   * 2단계
```

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

```
    - 3단계
      1. 4단계
```

각 단계마다 부모의 텍스트 시작 위치에 맞춘다.

피해야 할 패턴

```
❌ 탭 키 사용 (에디터마다 탭 폭이 다름)
1. 항목
	* 하위 항목

❌ 과도한 스페이스
1. 항목
        * 하위 항목

❌ 텍스트 위치와 무관한 임의 들여쓰기
1. 항목
    * 하위 항목
```

외부 문서에서 붙여넣기할 때 주의사항

다른 에디터(Word, Notion, 메모장 등)에서 작성된 마크다운을 BookStack에 붙여넣을 때, 들여쓰기가 탭이나 과도한 스페이스로 되어 있을 수 있다. 이 경우 중첩 리스트가 정상 렌더링되지 않으므로, 붙여넣기 후 들여쓰기를 위 규칙에 맞게 조정해야 한다.

간단한 확인법: 프리뷰 패널에서 하위 리스트가 제대로 표시되는지 확인한다. 프리뷰에서 깨지면 저장 후에도 깨진다.

레퍼런스

문서	URL
CommonMark Spec 0.31.2 — Section 5.2: List items	https://spec.commonmark.org/0.31.2/#list-items
CommonMark Spec — Motivation (들여쓰기 규칙 설계 의도)	https://spec.commonmark.org/0.31.2/#motivation
CommonMark Tutorial — Nested Lists	https://commonmark.org/help/tutorial/10-nestedLists.html
CommonMark Spec Issue #399 — 들여쓰기 규칙 논의	https://github.com/commonmark/commonmark-spec/issues/399
GitHub Flavored Markdown Spec (CommonMark 기반)	https://github.github.com/gfm/
markdown-it Issue #215 — 중첩 리스트 들여쓰기	https://github.com/markdown-it/markdown-it/issues/215
Markdown Guide — Basic Syntax (Lists)	https://www.markdownguide.org/basic-syntax/#lists-1

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.