⑦一人称代名詞エッジ抽出プロンプト（幕/場スコープ対応・感情なし・Self-loop・エラー行出力あり）

あなたは、指定スコープ（幕/場）の脚本テキストから、

「一人称代名詞（ich/mich/mir、wir/uns）に基づく Self-loop（自分→自分）の有向辺」を抽出し、CSVのみを出力します。

※ 名詞・他人称の辺は別プロンプトで処理。Aliasリストは参照しません（名詞照合は不要）。

# 出力ヘッダ（④/⑤/⑥と完全一致・順序固定・厳密一致）

Edge\_Id,Work\_Name,Speech\_Id,Source\_Node\_Id,Target\_Node\_Id,Alias\_Id,Alias\_Label,Span\_Start,Span\_End,Matched\_By,Excluded\_Flag,Speech\_Quote,Remarks

# 目的

- 発話者（Source）が一人称代名詞を用いて\*\*自己を指す\*\*箇所を検出し、\*\*Self-loop（Source=Target）\*\*の辺を生成する。

- 位置復元のため \*\*Speech\_Id + Span\_Start/End\*\* を必ず出力。

- 感情スコアは本プロンプトでは出力しない（第2段階で付与）。

# 入力

<WORK\_NAME>

（作品タイトル）

</WORK\_NAME>

<SCOPE>

Unit: act | scene

Value: 例) I または I-3

</SCOPE>

<NODE\_LIST\_CSV>

Node\_Id,Name,Work Name,First Line,Grammatical Number,Existed on initial list?,Remarks

…（このスコープを含むノードリストCSV）…

</NODE\_LIST\_CSV>

<SCOPE\_SCRIPT>

（指定スコープ＝該当の幕/場のみの原文。発話ラベルと本文を含む）

</SCOPE\_SCRIPT>

# 制約

- \*\*CSVのみ\*\*出力（解説・空行・コードフェンス禁止／RFC 4180でエスケープ）。

- 対象語形は \*\*人称代名詞\*\* のみ：`ich`(nom), `mich`(acc), `mir`(dat), `wir`(nom), `uns`(acc|dat)。

- \*\*所有代名詞\*\*（mein/meine/unser/unsere…）は対象外。

- Wikisource等の外部ブロック番号は使用しない。位置は \*\*Speech\_Id + Span\*\* で特定。

- \*\*1出現=1行\*\*（同一発話に複数あれば複数行）。

# 手順（要点）

1) 発話抽出と Speech\_Id 付与

- <SCOPE> に従い当該スコープのみ処理。

- <SCOPE\_SCRIPT> から「話者ラベル＋本文」を列挙し、Scene単位で `Speech\_Id` を採番

（形式：`I-3-U0007`。Sceneなしは `I-U0007`）。

2) 話者（Source\_Node\_Id）の同定

- 話者ラベルを <NODE\_LIST\_CSV> に突合して `Source\_Node\_Id` を確定。

- \*\*グループ話者\*\*は個体に一意分解できる場合のみ各個体に複製（分解不能なら通常出力は行わず、下記「6) エラー行」を生成）。

3) 代名詞トークンの検出

- 対象：`ich/mich/mir/wir/uns`（文頭大文字化等は許容）。\*\*語境界必須\*\*（Whole Word）。

- 各出現ごとに `Span\_Start / Span\_End` を取得し、`Alias\_Label` に実文字列を入れる。

- `Matched\_By=pronoun-coref-1p`。

4) Self-loop の生成（通常出力）

- \*\*Target\_Node\_Id = Source\_Node\_Id\*\*（同一ノード）。

- `Alias\_Id`：`PRON1P-S{Unit}{Value}-N{Source\_Node\_Id}-{Speech\_Id}-{k}`（同発話内連番 `k=1,2,…`）。

- `Excluded\_Flag=n`。

- `Speech\_Quote`：該当箇所を含む最大10語の抜粋（CSV安全）。

- \*\*単数話者が `wir/uns` を用いる\*\*場合も Self-loop を生成（過解釈で第三者を補わない）。`Remarks="単数話者の複数形使用"` 等を任意で付記可。

- \*\*グループ話者の個体分解が可能\*\*な場合は、各個体に同一発話を複製し、それぞれ Self-loop を生成（`Remarks="self-group-split"` 等を付記可）。

5) Edge\_Id（主キー）の生成（決定的）

- 通常出力：

`{WorkSlug}|{Speech\_Id}|S{Source\_Node\_Id}|T{Target\_Node\_Id}|{Alias\_Id}|X{Span\_Start}-{Span\_End}`

- `WorkSlug`：Work\_Name を小文字ASCIIに正規化（ä→ae, ö→oe, ü→ue, ß→ss、記号/空白除去、先頭12文字程度）。

6) ⚠ エラー行の生成（Excluded\_Flag=y）

下記に該当する場合は\*\*行を生成\*\*し、`Excluded\_Flag=y`、`Remarks` に理由を明記する。

- \*\*話者ノードが特定できない\*\*：`Source\_Node\_Id` を空欄、`Remarks="話者特定不能"`。

- \*\*話者が集合ノードで個体に分解できない\*\*：`Source\_Node\_Id` に集合ID、`Remarks="集合話者・個体分解不能"`。

- \*\*詩的表現等で通常の自己言及と見なせない\*\*（歌詞・合唱の反復など）：`Remarks="詩的表現（指示不成立）"`。

付与値の規約：

- `Target\_Node\_Id` は空欄（未確定）。

- `Alias\_Label` は検出した代名詞そのもの。

- `Matched\_By=pronoun-coref-1p`。

- `Alias\_Id` はエラー用：`PRON1P-ERR-S{Unit}{Value}-{Speech\_Id}-{k}`。

- \*\*Edge\_Id\*\*（プレースホルダ規約）：

`{WorkSlug}|{Speech\_Id}|S{SourceOrSNA}|TNA|{Alias\_Id}|X{Span\_Start}-{Span\_End}`

- `SourceOrSNA`：`Source\_Node\_Id` が無ければ `SNA`。

- `TNA`：ターゲット未確定トークン。

# 出力

- 次のヘッダで \*\*CSVのみ\*\*出力（順序・綴り厳守）：

Edge\_Id,Work\_Name,Speech\_Id,Source\_Node\_Id,Target\_Node\_Id,Alias\_Id,Alias\_Label,Span\_Start,Span\_End,Matched\_By,Excluded\_Flag,Speech\_Quote,Remarks