株式会社インフォメーション・ディベロプメント様向け

ハンズオンガイド

Day1 ヒアリングメモから要件定義書へ

改修依頼書と3本のヒアリングメモから、根拠の行番号が付いた要件一覧を作り、Wordの要件定義書にします。どの演習も、最初の数分はAIを開かずに自分で書きます。

2026年10月8日（木） Zoom 個人演習 Claudeデスクトップアプリ

目次

本日のゴール
3回の全体像
当日の流れ
Zoomでの進め方
題材の歩き方
サンプル情報サービス株式会社（架空）
改修依頼の4項目
登場人物
現行画面
配布フォルダの構成
資料の置き場と使いどころ
メモ帳での.mdの表示
手書きメモの置き場所
架空データの扱い
Codeタブの使い方
研修中のファイルの流れ
フォルダの開き方
@ によるファイル指定
文面の送り方
モードの切り替え
差分の見方
コマンド実行の許可
D1-1 文脈の有無による出力の違い
ねらい
できあがりの目安
使うファイルと作るファイル
STEP 1 渡すものと禁じることのメモ
STEP 2 3通りの依頼
STEP 3 メモにない要件と根拠の確かめ
STEP 4 違いの3点と指示の型
うまくいかないとき
早く終わった方へ
D1-2 ヒアリングメモからの要件抽出
ねらい
できあがりの目安
使うファイルと作るファイル
STEP 1 要件候補の手書き
STEP 2 Claudeによる抽出
STEP 3 自分の候補との突き合わせ
STEP 4 要件一覧の確定
うまくいかないとき
早く終わった方へ
D1-3 要件定義書のドラフト作成
ねらい
できあがりの目安
使うファイルと作るファイル
STEP 1 書きにくい章の見当
STEP 2 Markdownでの下書き
STEP 3 Wordへの変換と確認
STEP 4 要確認の数え直し
うまくいかないとき
早く終わった方へ
D1-4 抜け漏れ探し
ねらい
できあがりの目安
使うファイルと作るファイル
STEP 1 勤怠CSVからの抜け探し
STEP 2 Claudeによる抜け漏れ探し
STEP 3 自分の発見との比較
STEP 4 v0.2と確認事項の作成
うまくいかないとき
早く終わった方へ
よくある質問

01

本日のゴール

今日は、AIに要件を書かせるときに何を渡し、何をAIに決めさせないかを、3本のヒアリングメモを使って確かめます。作るファイルは、その学びが身に付いたかを確かめる材料です。

本日の学び

- 文脈の渡し方で、AIの書く要件がどれだけ変わるかを自分の目で確かめる
- 要件を、メモに書いてあることと、これから確かめることに分ける。食い違いやあいまいさはAIに決めさせず、確認する相手を決めて持ち帰る

学びを確かめる成果物

- ☐ `成果物/day1/要件一覧.md`。要件が25件以上あり、全件にファイル名と行番号の根拠が付いている
- ☐ `成果物/day1/要件定義書_v0.2.md`と、それを変換した`要件定義書_v0.2.docx`。書けない箇所は空けずに、要確認と確認相手を書いてある。docxは、時間が足りなければDay2の振り返りで作ります
- ☐ `成果物/day1/確認事項.md`。誰に、何を、なぜ聞くかが5件以上
- ☐ `成果物/day1/D1-1_比較.md`の末尾に、自分の言葉で書いた指示の型が1行

Day2 はこの要件定義書を入力にして画面を設計します。Day2の最初に、講師がDay1の完了状態を全員へ配ります。自分の成果物が途中でも、そこまでのファイルは消さずに残してください。

3回の全体像

各回の演習は、下の力のどれかを練習します。

AIに渡す材料を選ぶ力

AIの出力は、渡したメモと禁止事項でほぼ決まります。何を渡し、何を書かせないかを決めるのは人の仕事です。

AIの出力を確かめる力

もっともらしい要件や指摘ほど、根拠の行に戻って確かめます。採る、直す、捨てるは自分で決めます。

上流の成果物をつなげて見る力

要件から設計、設計から画面と数字までを1本の線で追います。1つの文書だけを見ていては気づかない抜けが見つかります。

回	身に付くこと	作るもの
Day1 要件定義（本日）	文脈の渡し方でAIの書く要件がどう変わるかを確かめる。食い違いやあいまいさはAIに決めさせず、確認する相手を決めて持ち帰る	根拠つきの要件一覧、要件定義書 v0.2、確認事項
Day2 画面設計	作る人が迷わない設計書に何が要るかを、自分で書いてつかむ。設計書を動くモックにして、書き漏れを先に見つける	画面一覧と遷移、KIN030の画面設計書、モック、請求回収の業務フロー
Day3 レビューと実装	AIには観点を分けてレビューさせ、指摘の正しさは根拠で判定する。要件から画面、画面から数字まで追いかけて穴を確かめる。3回のやり方をチームのルールにして持ち帰る	レビュー記録、対応表、収支ダッシュボード、レビュー会資料、利用ルールのたたき台

作るファイルは、下の図のように回をまたいでつながります。



図1：3回の成果物のつながり

3回とも、同じ架空の会社の同じシステムを扱います。前の回で作ったものが次の回の入力になるので、ファイル名は手順のとおりに付けてください。

02

当日の流れ

全体で [180min]です。講義のあとには、必ず演習か理解度チェックが入ります。

枠	中身	形式	時間
導入	3回の見取り図、題材のKINTAIと改修依頼、現行画面を触る、Codeタブで配布フォルダを開く、架空データの注意	説明	[15min]
D1-1	文脈の有無による出力の違い	個人演習	[25min]
D1-2	ヒアリングメモからの要件抽出	個人演習	[40min]
休憩			[10min]
D1-3	要件定義書のドラフト作成	個人演習	[35min]
D1-4	抜け漏れ探し	個人演習	[25min]
まとめ	指示の型3つ、要件定義書のどこを疑うか、理解度チェック	まとめ	[15min]
質疑・アンケート		質疑	[15min]

Zoomでの進め方

- 講師がZoomで画面を共有し、同じ操作を見せながら進めます。手元で使うのは、Claudeデスクトップアプリと、エクスプローラー、メモ帳、LibreOfficeです。画面が1枚の方は、Zoomの画面を小さくして横に並べてください。
- 演習はすべて個人で進めます。発表や、ほかの受講者との話し合いはありません。
- 質問や、詰まったときは、Zoomのチャットに「D1-3 STEP 3変換で文字化け」のように、演習の番号、STEP、起きていることを書いてください。講師が順に答えます。画面を見る必要があるときは、講師から画面共有をお願いすることがあります。
- 各STEPの[Nmin]は目安です。必須のSTEPは、演習の枠に収まる分数で組んであります。時間になったら講師が次のSTEPへ進めます。結果が途中でも、次のSTEPへ移って構いません。終わらなかった分は、研修のあとに同じ手順で続けられます。
- 必須のSTEPは、文面を送る操作を飛ばさないでください。次のSTEPの文面は、前のSTEPの結果を読むので、飛ばすと「読み取れない」と返ってきます。飛ばしてしまったときは、Zoomのチャットで知らせてください。講師が、そのSTEPの結果の例を送ります。
- 見出しに 発展 のバッジが付いたSTEPは、枠の時間に入っていません。必須のSTEPが終わったら、発展を待たずに次の演習へ進みます。
- Claudeの応答を待つ間は、次のSTEPの本文を読んでください。待ち時間を含めて、必須のSTEPが枠に収まるように組んであります。

03

題材の歩き方

3回とも、同じ架空の会社の、同じシステムの改修を扱います。導入の時間に、ここを一緒に見ます。

サンプル情報サービス株式会社（架空）

独立系のSIerで、社員は約300名です。第1開発部が金融、第2開発部が公共と運輸の案件を持ち、ほかに管理部があります。

この会社で全社員が使う唯一のWebシステムが、勤怠・工数管理システムKINTAIです。2014年に稼働し、JSP、Spring MVC、MyBatisで作られています。TERASOLUNA 5.xがベースです。研修でJavaのコードを読む場面はありません。TERASOLUNAはDay2で、入力チェックの書き分け方の決まりとしてだけ使います。

改修依頼の4項目

情報システム課が起票した`資料/day1/改修依頼書_KINTAI-2026-03.md`に、次の4項目が書いてあります。

1. 工数入力画面KIN030で、予定工数を入力できるようにする
2. 1日の作業時間の合計が24時間を超えたら入力エラーにする
3. プロジェクトの収支を見る画面PJ010収支ダッシュボードを新設する
4. 希望時期は2027年4月。詳細は各部のヒアリングで確認すること

4つ目にあるとおり、依頼書だけでは要件になりません。誰が、いつ、何に困っていて、どこまで直せばよいのかは、ヒアリングメモの中にあります。しかも、メモ同士で言っていることが合わない箇所があります。

登場人物

人物	所属	出てくる資料	受け持ち
田中課長	第1開発部 課長	`ヒアリングメモ_01_第1開発部_田中課長.md`	金融の2案件、PJ-2026-011とPJ-2026-014のPM
鈴木主任	第2開発部 主任	`ヒアリングメモ_02_第2開発部_鈴木主任.md`	電力の2案件、PJ-2026-017とPJ-2026-071のPM
佐藤さん	管理部 担当	`ヒアリングメモ_03_管理部_佐藤さん.md`	管理部で勤怠と原価のデータを扱う
山本さん	管理部経理 担当	`ヒアリングメモ_04_管理部経理_山本さん_請求回収.md`	請求と入金を扱う。Day2のD2-4で登場

人物は名字と役職だけで表します。取引先は「A社（架空・信託銀行）」の形で、実在の会社ではありません。

現行画面

配布フォルダの `現行画面/index.html` をダブルクリックすると、ブラウザでKINTAIの現行画面のモックが開きます。サーバは要りません。導入の時間に、次の5画面を一通り触ってください。

画面ID	画面名	使う人	中身
KIN010	メニュー	全員	各画面へのリンク。ログイン中の氏名と締め状態
KIN020	日次勤怠入力	全員	勤務区分、出勤、退勤、休憩（分）、実働（自動計算）
KIN030	工数入力	全員	日付ごとに、プロジェクト（選択）、作業区分、作業時間（0.25時間単位）、備考を行で入れる。合計は表示するが、チェックはしない
KIN040	交通費申請	全員	日付、区間、金額、目的
KIN050	月次締め・CSV	PM・管理部	締め状態、CSV出力、パートナー実績のCSV取込

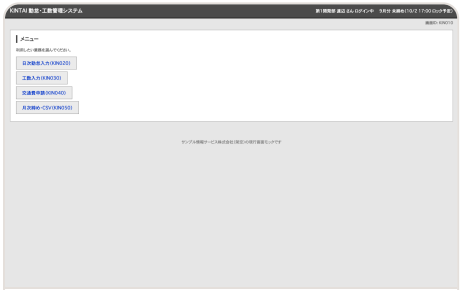


図2：KIN010メニュー



図3：KIN020日次勤怠入力



図4：KIN030工数入力



図5：KIN040交通費申請



図6：KIN050月次締め・CSV

触るときに見ておくこと

- ☐ KIN030で `行を追加` を押し、プロジェクトと作業区分を選んで作業時間に25を入れ、`保存` を押してみる。どのエラーが、画面のどこに出るか
- ☐ KIN020の実働が8時間の日に、KIN030で合計10時間の工数を入れてみる。エラーになるか
- ☐ KIN050で、締め状態とロック予定の日時がどう書いてあるか

ここで見たことは、D1-2以降で要件を読むときの手がかりになります。

配布フォルダの構成

事前準備で、配布ZIPをデスクトップに展開してあります。Claudeに渡すファイルも、作ったファイルの置き場所も、すべてこのフォルダの中です。

```
idup-handson/
├── はじめにお読みください.md
├── CLAUDE.md          Claude が最初に読む。AGENTS.md を読み込む
├── AGENTS.md          この研修での約束ごと
├── .claude/settings.json  資料/ に Claude が書き込めないようにする設定
├── 資料/              読むだけ。書き換えない
│   ├── day1/          ヒアリングメモ3本、改修依頼書
│   ├── day2/          経理のヒアリングメモ
│   ├── day3/          レビュー対象の画面設計書
│   ├── データ/        9月の勤怠工数CSV、マスタ、予定工数表、問い合わせ履歴、売上実績表
│   │   ├── 過去分/    7月と8月の勤怠工数CSV
│   │   └── 請求回収/  6月からの請求一覧と入金明細
│   ├── 参考/          現行のテーブル定義書、CSV出力仕様書、月次締め運用手順書、用語集など
│   ├── 記入例/        別の題材（会議室予約システム）で書いた記入例4本
│   └── 発展/          早く終わった方への発展課題で使う資料
├── 現行画面/          KINTAI の現行画面。ブラウザで開く
├── テンプレート/      要件定義書、画面設計書などの雛形6本
├── tools/             Markdown を Word・Excel・PowerPoint にする変換ツール
└── 成果物/
    ├── day1/  day2/  day3/   作ったものはここに置く
    └── 合流用/        合流版を展開する場所
```

場所	扱い
`資料/`	読むだけです。`.claude/settings.json`の設定で、Claudeのファイル編集とコマンドからの書き込みを止めてあります。pythonのスクリプトは止められないので、コマンドを読んで、`資料/`に書き込む内容のときは拒否します
`.claude/`	Claudeの設定のフォルダです。開いたり消したりしません
`テンプレート/`	要件定義書、画面設計書、レビュー記録、対応表、レビュー会資料、利用ルールの雛形です。自分でコピーはしません。Claudeに渡して、形をそろえさせます
`tools/`	ClaudeがMarkdownをWord、Excel、PowerPointの形式に変換するときに使います。自分で実行する必要はありません
`成果物/day1/`など	自分で書いたメモと、Claudeに作らせたファイルを置きます。研修が終わってもそのまま持ち帰れます

Claudeはフォルダを開くと`CLAUDE.md`を読み、そこから`AGENTS.md`の約束ごとを読み込みます。演習に関わる約束ごとを、下に抜き出します。

- 回答は日本語の です・ます調で書く
- ヒアリングメモにないことを、要件として足さない
- 推測で埋めた箇所には、文末に （要確認） を付ける
- 要件や指摘には、`ヒアリングメモ_01:L23`の形でファイル名と行番号の根拠を付ける

送る文面で毎回くり返さなくても、Claudeはこれを守ろうとします。ただし、約束ごとはClaudeへの頼みごとなので、守られないこともあると考えてください。`資料/`は`.claude/settings.json`の設定で、Claudeのファイル編集とコマンドからの書き込みを止めてあります。pythonのスクリプトは止められないので、コマンドを読んで、`資料/`に書き込む内容のときは拒否します。

資料の置き場と使いどころ

ヒアリングメモのほかに、現行システムの文書、過去の月のデータ、別の題材の記入例、発展課題の資料が入っています。演習のSTEPで使うものと、早く終わった方が使うものがあります。どれも`資料/`の中なので、読むだけで書き換えません。

ファイル	中身	使いどころ
`資料/参考/現行_テーブル定義書_KINTAI.md` `資料/参考/現行_テーブル定義書_KINTAI.xlsx`	KINTAIの今のテーブルと列。mdとxlsxは同じ中身です	D2-2 で、改修で足すテーブルと列を考える。D3-2の発展課題で、変更の影響を調べる
`資料/参考/現行_CSV出力仕様書_KIN050.md`	KIN050が出す勤怠工数CSVの列、文字コード、出力のタイミング	D2-1 でKIN050への影響を考える。D3-3でCSVの行の作り方を確かめる
`資料/参考/月次締め運用手順書.md`	管理部が毎月、手で行っている締めの手順	D1-4 で抜け漏れの根拠にする。D2-4とD3-3で、手作業がどこで減るかを見る
`資料/記入例/`の4本	会議室予約システムという別の題材で書いた、要件一覧、要件定義書、画面設計書、レビュー記録の抜粋	書き方に迷ったときに開く。D1-2、D1-3、D2-2、D3-1から案内します。中身はKINTAIの答えではありません
`資料/データ/過去分/勤怠工数CSV_202607.csv` `資料/データ/過去分/勤怠工数CSV_202608.csv`	7月と8月の勤怠と工数。9月と同じ列です	D1-4 とD3-3の発展課題で、9月と見比べる
`資料/データ/予定工数表_2026下期.xlsx`	田中課長が手で作っている、10月からの予定工数のExcel	D3-3 で、ダッシュボードの予定工数の元にする
`資料/データ/問い合わせ履歴_KINTAI_202510-202609.csv`	ヘルプデスクに届いたKINTAIの問い合わせの記録	D1-4 で抜け漏れの根拠にする。D1-2の発展課題で、困りごとの件数を数える
`資料/データ/請求回収/`	6〜8月の請求一覧と、7〜9月の入金明細。6本のCSVです	D2-4 の発展課題で、消し込みの決まりを3か月分に当てはめる
`資料/発展/`	12の演習の発展課題で使う資料。ファイル名は演習番号で始まります	各演習の「早く終わった方へ」から案内します

> **Tips** xlsxは、Claudeに渡すときは`tools/xlsx_to_md.py`でMarkdownにしてから読ませます。`AGENTS.md`にそう書いてあるので、「このxlsxを読んで」と頼めばClaudeが変換します。自分で見るときはLibreOffice Calcで開き、保存せずに閉じてください。

メモ帳での.mdの表示

この研修では、.mdのファイルはメモ帳で読み書きしてください。Windows 11のメモ帳は、.mdを`書式付き`の表示で開くことがあります。書式付きのままだと空行が行数に入らないので、下の帯の行番号がファイルの行番号とずれます。`#`や`|`も画面に出ません。

> **Tips** メモ帳で.mdを開いたら、下の帯の右側を見てください。`書式付き`と出ていたら押して、`マークダウン構文`に切り替えます。続けて`表示`で`右端で折り返す`の印を外してください。外すと、行番号がファイルの行番号と1対1で合います。Ctrl+Gを押して行番号を入れると、その行へ飛べます。このガイドで.mdをメモ帳で開く手順は、どれもこの表示が前提です。

下の帯が出ていないときは、`表示` → `ステータス バー`に印を付けてください。.mdをダブルクリックして別のアプリが開くときは、右クリック → `プログラムから開く` → `メモ帳`を選びます。開くときに表示のしかたを選ぶ画面が出たら、`Markdown構文ビューで開く`を選びます。`マークダウン構文`と同じ表示です。

手書きメモの置き場所

どの演習も、STEP 1はAIを開かずに自分で書きます。書く道具はWindowsのメモ帳です。書いたメモは、STEP 2以降でClaudeに送る文面の`<ここに…>`のところへ貼ります。

1. スタートメニューの検索に メモ帳 と入れて開きます。
2. メモを書いたら、`ファイル` → `名前を付けて保存`を選びます。
3. 保存先に`idup-handson`の中の`成果物` → `day1`を選びます。
4. ファイル名は、メモ_のあとに演習番号を付けて`メモ_D1-1.md`のようにします。`ファイルの種類`は`すべてのファイル(*.*)`、`エンコード`は`UTF-8`を選んで保存します。
5. 保存したあと、下の帯の右側が`書式付き`に変わっていたら、押して`マークダウン構文`に戻します。書き足すのは、この表示にしてからです。
6. 書き足したら、Ctrl+Sで上書き保存します。

> **Tips** ファイル名は`メモ_D1-3.md`のように演習番号で付けます。D1-2のSTEP 1だけは`自分の要件候補.md`です。

架空データの扱い

題材の会社、取引先、人物、金額、勤怠と工数のデータは、すべて架空のものです。研修中は、自分の案件の資料や実在のお客様の情報をClaudeに渡さないでください。自分の案件で試す手順は、Day3のあとに配る持ち帰り演習集にあります。

04

Codeタブの使い方

研修で使うAIは、Claudeデスクトップアプリの Code タブだけです。ターミナル、VS Code、WSLは使いません。このガイドの画面写真はMacで撮りました。ウィンドウの枠、キーの表示、LibreOfficeの表示の言語がWindowsとは違うので、ボタンの名前を手がかりに進めてください。講師が同じ操作を画面共有で見せます。

研修中のファイルの流れ

どの演習も、下の図の流れでファイルを作ります。Claudeに書かせるのはMarkdownで、Word、Excel、PowerPointへは `tools/` の変換ツールで変えます。直すときはMarkdownを直して、変換し直してください。

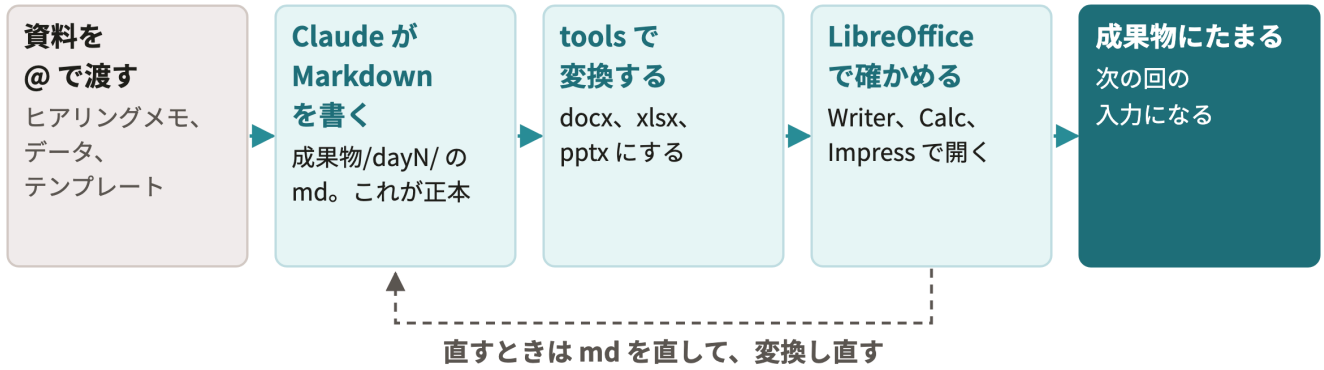


図7：研修中のファイルの流れ

フォルダの開き方

1. Claudeデスクトップアプリを開き、上の `Code` タブを選びます。
2. 環境が `ローカル` になっていることを確かめます。
3. `フォルダを選択` を押し、デスクトップの `idup-handson` を選びます。
4. 画面の上部に `idup-handson` と出れば開けています。事前準備のStep 3と同じ操作です。

@ によるファイル指定

入力欄で `@` を打つと、フォルダの中のファイルを指定できます。 `@` に続けて 改修 のようにファイル名の一部を入れ、出てきた候補から選んでください。

このガイドの送る文面には、先頭に `@資料/day1/改修依頼書_KINTAI-2026-03.md` のような行があります。コピーして貼ったままで、Claudeはそのファイルを読みます。自分でファイル名を打つときだけ、 `@` のあとの候補から選ぶと打ち間違いが起きません。



図8：入力欄で @ を打ち、ファイルの候補が出たところ

文面の送り方

このガイドの黒い枠の文面は、右上の `コピー` で写せます。`<ここに…>` のある文面は、そのまま貼らずに、メモ帳で埋めてから送ります。

1. `コピー` を押し、メモ帳に貼ります。
2. `<ここに…>` を、自分のメモや決めたことに書き換えます。`<` と `>` も消します。
3. メモ帳でCtrl+Aを押して全部を選び、Ctrl+Cでコピーします。
4. Claudeの入力欄にCtrl+Vで貼り、送信します。

入力欄では、Enterを押すと送信されます。入力欄の中で改行したいときはShift+Enterを押します。

モードの切り替え

モード名が出ているのは、入力欄の左下の、＋ とマイクのすぐ右です。押すと切り替えられます。選んだモードは、フォルダごとに次の回まで残ります。

モード	Claudeのふるまい	研修での使い方
手動	ファイルを書き換える前と、コマンドを実行する前に、そのつど確かめを求めます	Day1 のD1-1とD1-2はこれで進めます。Claudeが何をしようとしているかを1回ずつ見るためです
編集を受け入れる	ファイルの書き換えと、フォルダの中でのファイルの作成・移動・コピー・削除は、確かめなしで進みます。pythonで動かす変換ツールなどのコマンドだけが、確かめの対象です	D1-3 の最初に切り替えます。確かめの回数が減るぶん、作業のあとに差分の表示で中身を見ます
プラン	ファイルを書き換えず、進め方の案だけを返します	作らせる前に手順を見たいときに使います
自動	確かめの多くを省いて進みます	研修では使いません



図9：入力欄の左下のモード名を押して出るメニュー

`編集を受け入れる`のときに確かめなしで書き換わったり消えたりするのは、`成果物/`の中のファイルです。`資料/`は`.claude/settings.json`の設定で、Claudeのファイル編集とコマンドからの書き込みを止めてあります。pythonのスクリプトは止められないので、コマンドを読んで、`資料/`に書き込む内容のときは拒否します。

差分の見方

Claudeがファイルを作ったり書き換えたりすると、応答の中に`+12 −1`のような表示が出ます。+は足した行、−は消した行の数です。押すと、変えた中身を確かめられます。

Claudeの作業が終わるたびに、差分の表示で、書き換えたのが頼んだファイルかを見てください。頼んでいない章や行まで消していないかも見ます。

ファイルの削除は差分の表示に出ないことがあります。頼んでいないファイルが消えていないかは、エクスプローラーで`成果物/day1/`を開いて確かめてください。

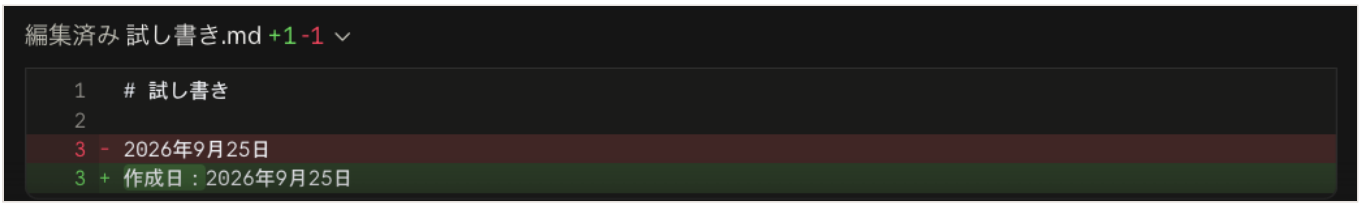


図10：ファイルを変えたあとに出る差分の表示

コマンド実行の許可

変換ツールや環境チェックのように、Claudeがコマンドを実行するときは、実行してよいかを聞かれます。ボタンは`拒否` `常に許可` `一度だけ許可`の3つです。表示されたコマンドを読んでから、下の表のとおりに押します。

コマンドの形	押すボタン
`python tools/md_to_docx.py 成果物/day1/… 成果物/day1/…`のように、`tools/`の変換ツールで`成果物/`の中に書き出すもの。`python`の代わりに`py`で始まることもあります	`一度だけ許可`

コマンドの形	押すボタン
`python tools/xlsx_to_md.py 資料/…` のように、xlsxをMarkdownにして読むだけのもの	**一度だけ許可**
`python tools/check_env.py`	**一度だけ許可**
`pip install` などのソフトの追加、`成果物/` の外への書き出し、何をするのか読み取れないもの	**拒否**。そのあと入力欄で「〜はしないでください」と伝え直します

常に許可 は押しません。押すと、同じ形のコマンドが次から確かめなしで動き、中身を読む機会がなくなります。

05

D1-1 文脈の有無による出力の違い

[25min] 個人演習 座学 [3min]

この演習の [25min]は、座学 [3min]と、STEP 1〜4の [22min]です。座学は演習の最初に入ります。

ねらい

同じ依頼でも、何を渡し、何を禁じるかで、出力がどれだけ変わるかを自分の目で見ます。ここで決める指示の型を、このあとの全演習で使います。

できあがりの目安

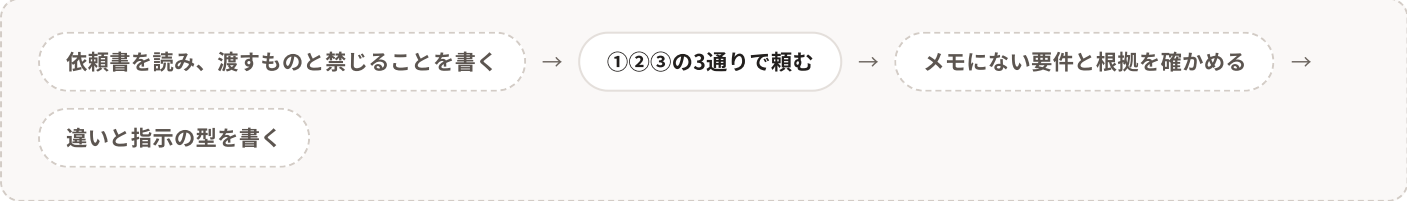
- ☐ `成果物/day1/D1-1_比較.md` に、①②③の3回分の出力が見出しを分けて並んでいる
- ☐ ①の見出しのすぐ下に、上から10件のうちメモにない要件の数が書いてある
- ☐ ③の根拠のうち2つを、メモ帳で開いて確かめた結果が書いてある
- ☐ 違いが自分の言葉で3点、指示の型が1行で書いてある

使うファイルと作るファイル

区分	ファイル	使い方
使う	`資料/day1/改修依頼書_KINTAI-2026-03.md`	STEP 1 で読みます。②③でClaudeに渡します

使う	`資料/day1/ヒアリングメモ_01_第1開発部_田中課長.md`
	`資料/day1/ヒアリングメモ_02_第2開発部_鈴木主任.md`
	`資料/day1/ヒアリングメモ_03_管理部_佐藤さん.md`
		②③でClaudeに渡します

作る	`成果物/day1/メモ_D1-1.md`	STEP 1 の手書きメモ
作る	`成果物/day1/D1-1_比較.md`	3回分の出力と、自分で書く違いと指示の型



この演習は、1つの会話で最後まで進めます。STEPの文面は前のSTEPの続きとして送るので、途中で新しい会話にしないでください。演習が変わったら、新しい会話を始めて構いません。

STEP 1 渡すものと禁じることのメモ

AIを開かない　[5min]

Claudeはまだ使いません。`資料/day1/改修依頼書_KINTAI-2026-03.md`をメモ帳で開いて読みます。自分がこの依頼を部下に渡して要件を書かせるなら、何を一緒に渡し、何をするなと言うかを考えます。

メモ帳で新しいファイルを作り、下の形を貼ってください。渡すものを2行、禁じることを2行書き、`成果物/day1/メモ_D1-1.md`の名前で保存します。保存のしかたは、題材の歩き方の「手書きメモの置き場所」にあります。

メモの形

```
渡すもの
-
-
禁じること
-
-
```

手が止まったら開く −

渡すものの行は、資料の名前に加えて、その資料から何を読み取ってほしいかまで書くと、STEP 2の③で使えます。禁じることの行は、部下が書いてきた要件で困った経験を思い出すと出てきます。たとえば、打ち合わせで誰も言っていない機能が入っていた、数字が推測で書かれていた、などです。

STEP 2 3通りの依頼

[7min]

モードが`手動`になっていることを確かめてから、①②③の順に同じ会話で送ります。①のあとに②、②のあとに③と、足すものを1つずつ増やします。前の応答が終わってから、次の文面を送ってください。

手動モードでは、送るたびに、ファイルを書き換えてよいかの確かめが出ます。書き換える先が`成果物/day1/D1-1_比較.md`になっていることを見てから、許可のボタンを押します。

①何も渡さない

Claudeに送る文面

```
ファイルは読まずに、あなたの知識だけで答えてください。
勤怠システムの工数入力に予定工数を足す要件を書いて。
書いた内容は 成果物/day1/D1-1_比較.md を新しく作り、「## ① 何も渡さない」の見出しの下に保存してください。
```

②資料を渡す

Claudeに送る文面

@資料/day1/改修依頼書_KINTAI-2026-03.md
@資料/day1/ヒアリングメモ_01_第1開発部_田中課長.md
@資料/day1/ヒアリングメモ_02_第2開発部_鈴木主任.md
@資料/day1/ヒアリングメモ_03_管理部_佐藤さん.md

この4つを読んで、勤怠システムの工数入力に予定工数を足す要件を書いて。
書いた内容は 成果物/day1/D1-1_比較.md の末尾に「## ② 資料を渡す」の見出しを足して、その下に保存してください。

③禁止と根拠を足す

Claudeに送る文面 メモ帳で<ここに…>を埋めてから貼る

@資料/day1/改修依頼書_KINTAI-2026-03.md
@資料/day1/ヒアリングメモ_01_第1開発部_田中課長.md
@資料/day1/ヒアリングメモ_02_第2開発部_鈴木主任.md
@資料/day1/ヒアリングメモ_03_管理部_佐藤さん.md

この4つを読んで、勤怠システムの工数入力に予定工数を足す要件を書いて。
- メモにないことは書かない
- 各要件に、根拠のファイル名と行番号を付ける（例 ヒアリングメモ_01:L23）
- 次の2つも守る
<ここに STEP 1 で書いた「禁じること」2行を貼る>
書いた内容は 成果物/day1/D1-1_比較.md の末尾に「## ③ 禁止と根拠を足す」の見出しを足して、その下に保存してください。

できていれば次へ

- ☐ `D1-1_比較.md` に① ② ③の3つの見出しがあり、それぞれの下に要件が並んでいる

STEP 3 メモにない要件と根拠の確かめ

AIを開かない [5min]

`成果物/day1/D1-1_比較.md` をメモ帳で開き、下の帯の右側が `マークダウン構文` になっていることを確かめます。

1. 「## ①何も渡さない」の見出しの下にある要件を、上から10件まで読みます。10件より少ないときは、全件を読みます。

2. その要件のうち、改修依頼書にも、「## ③禁止と根拠を足す」の見出しの下にも出てこないものを探し、行末に【メモにない】と書き足します。③はメモの行を根拠にした要件なので、③に無い要件は、メモに書いていない見込みが高いものです。承認の流れ、通知メール、グラフ表示のように、もっともらしいけれど誰も頼んでいないものを探します。

3. 【メモにない】の数を数え、「## ①何も渡さない」の見出しのすぐ下に、「上から10件のうち メモにない要件○件」と書きます。

4 ③の要件から根拠を2つ選んでください。その根拠のメモをメモ帳で開き、下の帯の右側が **マークダウン構文** になっていることを確かめます。Ctrl+Gで根拠の行番号へ飛び、要件と同じことが書いてあるかを見ます。

5 `D1-1_比較.md` の末尾に「##確かめた結果」の見出しを足します。その下に、選んだ2つの根拠ごとに、合っていたかずれていたかを1行ずつ書きます。ずれていたら、何行ずれていたかも書いてください。

> **メモ帳で書き足したら、保存して閉じてから次へ進みます。** 開いたままClaudeにこのファイルを書き換えさせると、あとでメモ帳の保存をしたときにClaudeの書き換えが消えます。

STEP 4 違いの3点と指示の型

AIを開かない [5min]

`D1-1_比較.md` の末尾に「##違い」の見出しを足し、①②③の違いを1点1行で3点書きます。書くのは件数の差ではなく、中身の差です。どの要件が消えたか、どの要件が増えたか、どの書き方が変わったかを見ます。

続けて「##指示の型」の見出しを足し、「渡す・禁じる・根拠を求める」を自分の言葉で1行にしてください。これから先の演習で送る文面は、どれもこの型でできています。書き終えたら保存して閉じます。

<details>
<summary>違いの書き方の例を見る</summary>

「①は予定工数の承認フローや月次レポートまで書いてきたが、どのメモにも出てこない。③では消えた」

「②は田中課長の言い方と鈴木主任の言い方を混ぜて1つの要件にしていた。③は根拠が2つ付いたので、2人の言うことが違うのが見えた」

「③の根拠を開いたら、1件は行番号が2行ずれていた。中身は合っていた」
</details>

うまくいかないとき

<details>
<summary>①でClaudeがフォルダのファイルを読んで答えた</summary>

①の出力に依頼書やメモの言葉が出ていたら、「ファイルは読まずに」が効いていません。①の見出しの下を消してもらい、「ファイルは一切読まずに、一般的な知識だけで」と言い換えて送り直してください。この配布フォルダではClaudeが最初に `CLAUDE.md` と `AGENTS.md` を読むので、何も言わないと資料を探しに行くことがあります。
</details>

<details>
<summary>③でも、メモにない要件が混ざる</summary>

「③の要件のうち、根拠の行にその内容が書いていないものを挙げてください」と送ってください。挙がったものは、消すか、文末に「(要確認)」を付けさせます。AIはもっともらしい要件ほど、根拠がなくても書きます。消すかどうかを決めるのは自分です。
</details>

根拠の行番号が、メモ帳の行番号とずれる −

まず、メモ帳の下の帯の右側を見てください。**書式付き**と出ていたら押して、**マークダウン構文**に切り替えます。書式付きのままだと空行が行数に入らず、空行の数だけ行番号がずれます。

次に、**表示**で**右端で折り返す**の印を外してください。この2つを直してもずれるときは、Claudeが数え違えています。「根拠の行番号を、ファイルを読み直して数え直してください」と送ってください。どれだけずれたかは、追加と考察で使います。

ファイルが作られない、別の場所に保存された −

差分の表示で、どのファイルが変わったかを見ます。保存先が違ったら「**成果物/day1/D1-1_比較.md**に保存し直してください」と送ります。手動モードで確かめを求められたのに押し忘れていると、ファイルは作られません。

早く終わった方へ

追加と考察

③の根拠を全部開き、行番号がずれている要件の数を数えます。数える前に、メモ帳の表示が**マークダウン構文**になっていることを確かめてください。書式付きの表示のままだと、Claudeが正しくてもずれて見えます。

ずれていたら、なぜずれたのかを考えます。Claudeが見出しや空行を数えに入れたのか。2行にまたがる発言の1行目と2行目のどちらを指したのか。考えたことは`D1-1_比較.md`の末尾に書きます。

発展課題

③の文面の最後に、次の1行を足して送り直します。

Claudeに送る文面

分からないところは、要件を書く前に、私への質問として返してください。

返ってきた質問を読み、ヒアリングで本当に聞くべきものに印を付けます。聞くべき質問が多いほど、メモだけでは要件が決まらないということです。D1-4の確認事項リストの下書きになります。

続けて、同じ1行を付けたまま、第2開発部の高橋主任から届いた依頼メールで要件を書かせてください。メールの頼み方はメモよりずっと粗いので、質問がどれだけ返ってくるかを見ます。

Claudeに送る文面

@資料/発展/D1-1_高橋主任からの依頼メール.md

> このメールから、KINTAI の改修の要件を書いてください。
> 分からないところは、要件を書く前に、私への質問として返してください。

返ってきた質問を、`メモ_D1-1.md`で2つに分けます。3本のヒアリングメモの本文を読めば答えられるものと、高橋主任本人に聞くしかないものです。後者が、次のヒアリングで持っていく質問になります。

06

D1-2 ヒアリングメモからの要件抽出

[40min] 個人演習 座学 [4min] 理解度チェック [3min]

この演習の [40min]は、座学 [4min]、理解度チェック [3min]と、STEP 1〜4の [33min]です。座学は演習の最初とSTEP 3の頭、理解度チェックはSTEP 2のあとに入ります。

ねらい

3本のメモから、要件を根拠付きで抜き出します。メモ同士で言っていることが食い違う箇所を、要件一覧の中に見える形で残します。

できあがりの目安

- ☐ `成果物/day1/要件一覧.md` に要件が25件以上ある
- ☐ 全件に、ファイル名と行番号の根拠がある
- ☐ 「要確認」が3件以上あり、それぞれに誰に何を聞くかが書いてある
- ☐ メモ同士の食い違いが3つ以上、要確認として載っている。食い違いは3本のメモと依頼書の中に5つあります

使うファイルと作るファイル

区分	ファイル	使い方

使う	`資料/day1/ヒアリングメモ_01_第1開発部_田中課長.md`
	`資料/day1/ヒアリングメモ_02_第2開発部_鈴木主任.md`
	`資料/day1/ヒアリングメモ_03_管理部_佐藤さん.md`
		STEP 1 で自分で読み、STEP 2でClaudeに渡します

使う	`資料/day1/改修依頼書_KINTAI-2026-03.md`	STEP 2 で渡します
作る	`成果物/day1/自分の要件候補.md`	STEP 1 で自分で書く要件候補
作る	`成果物/day1/要件一覧.md`	STEP 2 でClaudeに作らせ、STEP 3とSTEP 4で直して確定します

メモを読み、要件候補を10件以上書く → Claudeに抽出させる → 自分の候補と突き合わせる → 抜き取りで根拠を確かめて確定する

この演習は、1つの会話で最後まで進めます。STEPの文面は前のSTEPの続きとして送るので、途中で新しい会話にしないでください。演習が変わったら、新しい会話を始めて構いません。

STEP 1 要件候補の手書き

AIを開かない [10min]

3本のメモをメモ帳で読み、要件の候補を10件以上、`成果物/day1/自分の要件候補.md` に書きます。1本に3分ずつ使い、最後の1分で見直します。

1行に1件です。行末に、どのメモの何行目かを書きます。行番号は、下の帯の右側を `マークダウン構文` にした表示で読みます。2人の言うことが合わないと思った箇所には、行頭に「？」を付けます。

書き方の形。例は図書館の貸出システム

- 貸出の延長を画面から申し込める（01:L行番号）
- ？ 延長を申し込める期限（01:L行番号 と 03:L行番号 で言い方が違う）
- 返却が遅れた人にメールで知らせる（02:L行番号）

ヒント 食い違いを探す場所 −

同じ話題が2本以上のメモに出てくる箇所を探し、言っていることが同じかを見比べてください。数字、期限、誰がやるかは、話す人によって言い方がずれやすいところです。

> **Tips** 書き方に迷ったら、`資料/記入例/記入例_要件一覧_会議室予約.md` を開きます。会議室予約システムという別の題材で、1件の粒度と根拠の書き方が見られます。

> **Tips** 要件は「〜したい」のままにせず、「〜できる」「〜を表示する」のように、できあがった状態で書くと、あとでClaudeの出力と比べやすくなります。

STEP 2 Claudeによる抽出

[7min]

自分の候補はまだ見せません。先に見せると、Claudeがそれに引っぱられて、比べる意味がなくなります。

Claudeに送る文面

```
@資料/day1/ヒアリングメモ_01_第1開発部_田中課長.md
@資料/day1/ヒアリングメモ_02_第2開発部_鈴木主任.md
@資料/day1/ヒアリングメモ_03_管理部_佐藤さん.md
@資料/day1/改修依頼書_KINTAI-2026-03.md

この4つから要件を抜き出して、表にしてください。
列は 要件ID / 要件 / 分類 / 根拠 / 発言者 / 要確認 の6つです。
- 要件IDは REQ-001 から順に振る
- 分類は 機能・非機能・制約・課題 のどれか
- 根拠は「ヒアリングメモ_01:L23」の形で、1件ごとに必ず付ける
- メモ同士で言っていることが食い違う箇所は、要確認の列に「食い違い」と書き、相手のメモの行も根拠に並べる
- 1件に1つのことだけを書く
```

- メモにないことは書かない
表は 成果物/day1/要件一覧.md に保存してください。

手動モードなので、ファイルを書き換えてよいかの確かめが出ます。書き換える先が`成果物/day1/要件一覧.md`になっていることを見てから、許可のボタンを押します。

届いた表の件数を見てください。25件に届かないときは、「1件に2つ以上のことが入っている要件を分けてください」と送ります。

STEP 3 自分の候補との突き合わせ [7min]

ここで初めて、自分の候補をClaudeに見せます。Claudeには仕分けだけをさせ、どう直すかは自分で決めます。

Claudeに送る文面

@成果物/day1/要件一覧.md
@成果物/day1/自分の要件候補.md

自分の要件候補と要件一覧を突き合わせて、次の3つに分けた表を返してください。要件一覧はまだ書き換えないでください。
1. 自分の候補にだけあるもの
2. 要件一覧にだけあるもの
3. 同じことを指しているが、中身が食い違うもの
それぞれ、根拠のメモの行も並べてください。

返ってきた表を見ながら、3つの区分ごとに扱いを決めます。

区分	扱い
1.自分の候補にだけあるもの	根拠のメモの行を開き、書いてあれば要件一覧に足します
2.要件一覧にだけあるもの	自分の候補に無かった理由が気になるものを3件まで選び、根拠の行を開いて確かめてください。根拠の行に書いていなければ、消すか要確認にします。選ばなかったものはそのまま残し、STEP 4の抜き取りで確かめます
3.中身が食い違うもの	要確認にし、誰に何を聞けば決まるかを書きます

決めたことを、次の文面の`<ここに…>`に自分で書いて送ります。

Claudeに送る文面 メモ帳で<ここに…>を埋めてから貼る

成果物/day1/要件一覧.md を次のとおり直してください。
<ここに、足す要件・消す要件・要確認にする要件と、それぞれ誰に何を聞くかを自分で書いて貼る>
要確認の列は「要確認: 営業部の課長と経理の担当に、請求書の発行日をどちらに合わせるかを確かめる」のように、相手と聞くことを書いてください。

STEP 4 要件一覧の確定

最初の [5min]はAIを開かない　[9min]

根拠の確かめは自分でやります。Claudeに「根拠は正しいか」と聞いても、同じ読み違いをしたまま「正しい」と返ることがあるからです。手順1と2は、Claudeを使わずに進めます。

1. 要件一覧から、根拠のメモが01、02、03に1件ずつになるように3件を選びます。STEP 3で開いたものは選びません。
2. 根拠のメモをメモ帳で開き、下の帯の右側が`マークダウン構文`になっていることを確かめます。Ctrl+Gで根拠の行へ飛び、要件と同じことが書いてあるか、発言者の列が合っているかを見ます。
3. ずれていたら、要件一覧の該当行を自分で直すか、Claudeに直させます。
4. 件数、根拠のない行、要確認の数を数え、できあがりの目安と見比べます。数えるのはClaudeに頼んでも構いません。

時間が残ったら見直すところ

- ☐ 「〜したい」の願望のまま残っている要件がない
- ☐ 1件に2つ以上のことが入っていない
- ☐ 発言者と根拠のメモが合っている
- ☐ 食い違いの要確認に、聞く相手が2人とも書いてある

うまくいかないとき

AIがメモにない要件を足してくる

「根拠の列にある行を読み直し、その行に書いていない要件を挙げてください」と送ります。挙がったものは消すか、要確認にします。「一般的に必要」「通常は」という言い方で入ってくる要件は、たいていメモにありません。

25件に届かない

「1件に2つ以上のことが入っている要件を分けてください」「非機能と制約の要件を、メモからもう一度探してください」と送ります。それでも届かないときは、メモの中の困りごとの発言を拾えているかを見てください。困りごとは、裏返すと要件になります。

表の列がずれる、崩れる −

メモ帳で`要件一覧.md`を開き、下の帯の右側を`マークダウン構文`に切り替えます。書式付きの表示では`|`が画面に出ません。行の先頭と末尾の`|`の数を見てください。要件の中に`|`が入っていると列がずれます。

「表の列が6つそろっているかを全行確かめて、ずれた行を直してください」と送ると直ります。

根拠が「ヒアリングメモ_01」だけで、行番号がない −

「根拠に行番号のない要件を挙げ、行番号を付けてください」と送ります。1つの要件が複数の行にまたがるときは「L23-L25」の形で書かせます。

早く終わった方へ

追加と考察

AIが拾わなかった要件は、メモの中でどんな書き方をされていたかを見ます。雑談のような前置き、「ちなみに」で始まる一言、「困ってはいないけど」という否定の形、別の話題の途中に出てくる数字。自分の現場のヒアリングメモで、同じ書き方が出てくる場所を思い浮かべてください。

発展課題

要件に優先度を付けさせます。優先度は、情報システム課が決めた判断基準に当てはめて付けます。

Claudeに送る文面 メモ帳で<ここに…>を埋めてから貼る

```
@成果物/day1/要件一覧.md
@資料/発展/D1-2_優先度の判断基準_情報システム課.md
@資料/データ/問い合わせ履歴_KINTAI_202510-202609.csv

各要件に優先度の列を足してください。値は「初期リリース必須」か「次版」のどちらかです。
判断基準のファイルの順番で判定し、どの基準に当たったかを列に書いてください。
問い合わせ履歴の件数で決める基準に当たりそうな要件は、同じ困りごとが月に何件あったかを数えて、その数も書いてください。
判断の根拠になったメモの発言を、ファイル名と行番号で示してください。メモに判断の材料がない要件は「要確認」にしてください。
私の考えでは、初期リリースに必ず入れたいのは次の要件です。
<ここに、自分が初期リリース必須だと思う要件IDを3つ書く>
```

自分の3つとClaudeの判断が食い違ったら、どちらの根拠が強いかを比べます。問い合わせの件数を数えた要件は、数え方が合っているかをCalcで問い合わせ履歴を開いて1件確かめてください。

07

D1-3 要件定義書のドラフト作成

[35min] 個人演習 座学 [4min]

この演習の [35min]は、座学 [4min]と、STEP 1〜4の [31min]です。座学は演習の最初とSTEP 3の頭に入ります。

ねらい

要件一覧をテンプレートに流し込み、Wordの要件定義書にします。分からない箇所は推測で埋めず、「要確認」と確認の相手を書いて残します。

できあがりの目安

- ☐ `成果物/day1/要件定義書_v0.1.docx` がLibreOffice Writerで開ける
- ☐ テンプレートの「書き方」の章を除いて、見出しが全部ある
- ☐ 空欄がない。書けない箇所には「要確認」と、確認する相手が書いてある

使うファイルと作るファイル

区分	ファイル	使い方
使う	`テンプレート/要件定義書テンプレート.md`	見出しと順番をこのとおりにします
使う	`成果物/day1/要件一覧.md`	D1-2 で確定したもの
使う	`tools/md_to_docx.py`	ClaudeがWordへの変換に使います
作る	`成果物/day1/メモ_D1-3.md`	STEP 1 の手書きメモ
作る	`成果物/day1/要件定義書_v0.1.md`	Markdownの下書き。直すときはこのmdを直す
作る	`成果物/day1/要件定義書_v0.1.docx`	変換したWord

> **Tips** この演習の最初に、モードを `編集を受け入れる` に切り替えてください。入力欄の左下のモード名を押して選びます。`編集を受け入れる` では、ファイルの書き換えと、フォルダの中でのファイルの作成・移動・コピー・削除は、確かめなしで進みます。pythonで動かす変換ツールなどのコマンドだけが、確かめの対象です。Claudeの作業が終わるたびに、差分の表示で変わったファイルと中身を見ます。

書きにくい章と確認相手を書く → Markdownで書かせる → Wordにして開く → 要確認を章ごとに数え、1章を書き直す

この演習は、1つの会話で最後まで進めます。STEPの文面は前のSTEPの続きとして送るので、途中で新しい会話にしないでください。演習が変わったら、新しい会話を始めて構いません。

STEP 1 書きにくい章の見当

AIを開かない [5min]

テンプレート/要件定義書テンプレート.md をメモ帳で開き、下の帯の右側を **マークダウン構文** にします。書式付きの表示では `##` が画面に出ません。

`##` で始まる見出しを上から見ます。最初の「書き方」の章は、書くときの決まりです。残りの章を、要件一覧の中身で埋まりそうな章と、埋まりそうにない章に分けます。分けた結果は書かなくて構いません。

成果物/day1/メモ_D1-3.md に、一番書きにくい章と、その章を誰に確認するかを2行で書きます。

<details>
<summary>書き方の例を見る</summary>

「非機能要件の性能。ストレスなく使えること、としか言われていない。今の画面が何秒で開くかを情報システム課に聞く」

</details>

STEP 2 Markdownでの下書き

[12min]

Claudeに送る文面 メモ帳で<ここに…>を埋めてから貼る

```
@テンプレート/要件定義書テンプレート.md
@成果物/day1/要件一覧.md

テンプレートの見出しと順番をそのまま使い、要件一覧の中身を流し込んで、要件定義書を Markdown で書いてください。保存先は成果物/day1/要件定義書_v0.1.md です。
- 「書き方」の章は守る決まりとして読み、書き終えたら消す
- ほかの見出しは消さない。書くことがない章も見出しを残し、「要確認」と書く
- 要件一覧にないことを推測で書いた文には、文末に「(要確認: 確認する相手)」を付ける
- 各要件には、要件一覧の要件IDを残す
- 要件一覧で「食い違い」になっている要件は、どちらかに決めずに両方の言い分を並べる

私が一番書きにくいと思った章と、確認する相手は次のとおりです。この章は特に、推測で埋めずに要確認で残してください。
<ここに STEP 1 のメモ2行を貼る>
```

できあがったら、差分の表示を開いて章の並びを見ます。テンプレートの見出しが途中で抜けていないか、見出しの名前が勝手に変わっていないかを確かめます。

> **Tips** 書き方に迷ったら、`資料/記入例/記入例_要件定義書_抜粋_会議室予約.md` を開きます。別の題材で、要確認の付け方と確認相手の書き方が見られます。

> **Tips** 要件定義書を直すときは、Markdownの方を直して変換し直してください。Wordで直すとMarkdownと食い違い、Day2でClaudeに渡すときに古い中身を読ませることになります。

STEP 3 Wordへの変換と確認 [9min]

変換の前に、同じ名前のファイルをWriter・Calc・Impressで開いていたら閉じます。開いたままだと書き込めず、変換が止まります。

Claudeに送る文面

成果物/day1/要件定義書_v0.1.md を tools/md_to_docx.py で Word にして、成果物/day1/要件定義書_v0.1.docx に保存してください。

1. コマンドの確かめが出たら、`python tools/md_to_docx.py 成果物/day1/要件定義書_v0.1.md 成果物/day1/要件定義書_v0.1.docx` の形になっているかを見て `一度だけ許可` を押します。
2. エクスプローラーで `成果物` → `day1` を開き、`要件定義書_v0.1.docx` をダブルクリックします。LibreOffice Writer以外が開くときは、右クリック → `プログラムから開く` → `LibreOffice Writer` を選びます。
3. F5 を押すとナビゲーターが開き、見出しの一覧が出ます。1.文書情報 から12.確認事項 までの12章が、抜けずに並んでいるかを見ます。

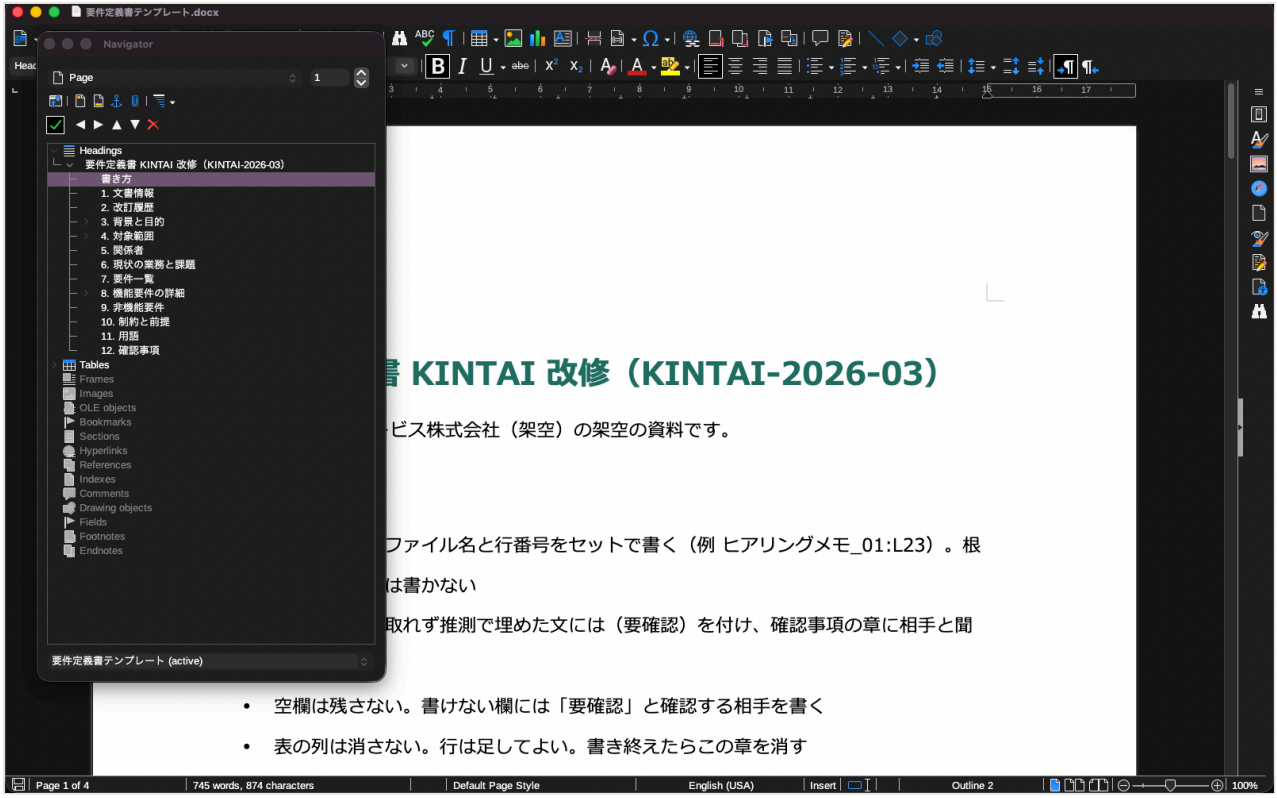


図11：要件定義書をLibreOffice Writerで開き、F5でナビゲーターを出したところ。英語表示のMac版で撮っています。ナビゲーターは別ウィンドウで出ることがあります

見るところ

- ☐ 見出しが見出しの書式になっていて、ナビゲーターに並ぶ
- ☐ 表に罫線があり、見出しの行に色が付いている
- ☐ `|` や `**` のような記号が、そのまま文字で残っていない
- ☐ 日本語が化けていない

STEP 4 要確認の数え直し

[5min]

要確認が多すぎる章を1つ見つけて、その章だけを書き直します。章ごとの数はClaudeに数えさせ、割合が一番高い章だけを自分で数えて確かめます。

Claudeに送る文面

成果物/day1/要件定義書_v0.1.md の「要確認」を、## の章ごとに数えて表で返してください。ファイルは書き換えないでください。
列は 章 / 文の数 / 要確認を含む文の数 / 割合 の4つです。
- 表の外は、句点までを1文と数える
- 表は、見出しの行と区切りの行を除き、1行を1文と数える
- 7章 要件一覧 の優先度の列と、12章 確認事項 の期限の列に書いた要確認は数えない

- 1つの文に要確認が2つあっても、1文と数える
- 最後に、割合が一番高い章を1つ挙げる

1. `成果物/day1/要件定義書_v0.1.md` をメモ帳で開き、下の帯の右側を `マークダウン構文` にします。
2. Claudeが挙げた章の中で、要確認を含む文と表の行を数えます。数え方は、送った文面の箇条と同じです。
3. `メモ_D1-3.md` に、章の名前、Claudeの数、自分で数えた数を1行で書きます。数は、どちらも要確認を含む文の数です。
4. 自分で数えた数が、Claudeの表の文の数の半分を超えていたら、その章だけを書き直します。書き直すのは、要件一覧を見れば自分で書ける要確認2つまでです。直すのはMarkdownです。半分を超えていなければ、書き直しはしません。
5. Markdownを保存して閉じます。

Wordへの変換し直しは、時間が残ったときだけ、Writerで開いているdocxを閉じてからSTEP 3の文面で行います。D1-4ではMarkdownのv0.1からv0.2を作るので、Markdownを直してあればv0.2のWordに入ります。

さらに時間が残ったら、その章の残りの要確認に、確認の相手が具体的に書いてあるかを見てください。要確認が0件の章も、メモにない数字や言い切りで埋まっていないかを確かめます。

うまくいかないとき

pythonが見つからない

Claudeが「pythonが見つかりません」と返したら、「pyで実行してください」と送ります。`AGENTS.md` にも同じことが書いてあるので、多くはClaudeが自分で `py` に切り替えます。どちらも動かないときは、Windowsの `設定` → `アプリ` → `アプリの詳細設定` → `アプリ実行エイリアス` で、`python.exe` と `python3.exe` をオフにしてからClaudeデスクトップアプリを開き直してください。それでも動かなければ、Pythonの導入が済んでいないので講師に知らせてください。

変換で文字化けする

Writerで化けているときは、まず元のMarkdownをメモ帳で開いて化けていないかを見ます。Markdownが化けていなければ、Writerの書体がMeiryo以外になっていないかを見ます。Markdownをメモ帳で書き直した場合は、保存するときの `エンコード` が `UTF-8` になっていたかを確かめてください。

LibreOfficeで開けない

エクスプローラーでdocxの大きさが0 KBなら、変換が失敗しています。Claudeの応答にエラーの文が出ていないかを見て、その文を貼って「変換のエラーを直してください」と送ってください。ダブルクリックで別のアプリが開くときは、右

クリック → `プログラムから開く` でLibreOffice Writerを選びます。

テンプレートの見出しが減った、名前が変わった −

「テンプレートの見出しとv0.1の見出しを並べ、足りないものと名前が違うものを直してください」と送ります。

変換で「LibreOfficeで開いているファイルを閉じてから」と出た −

変換先のファイルをWriter・Calc・Impressで開いたままにしていると、書き込めずに止まります。開いているウィンドウを閉じて、ファイル名は変えずに「もう一度同じ変換をしてください」と送ってください。

前の演習が終わらず、使うファイルがない −

講師が配るDay1当日の合流版で追いつけます。`要件一覧.md` と `要件定義書_v0.1.md` が入っています。

1. 講師がZoomのチャットに合流版のURLを貼ります。ブラウザで開くと `合流_Day1当日.zip` がダウンロードされます。
2. エクスプローラーで `ダウンロード` を開き、ZIPを右クリック → `すべて展開` を選びます。`参照` で `idup-handson` → `成果物` → `合流用` を選び、`展開` を押します。
3. `成果物/day1/` に自分の途中のファイルがあれば、コピーの前に名前の末尾へ `_自分` を付けます。`要件一覧.md` なら `要件一覧_自分.md` です。上書きで消さないためです。
4. `成果物` → `合流用` → `day1_当日` → `成果物` → `day1` を開き、`要件一覧.md`、`要件定義書_v0.1.md` を、配布フォルダの `成果物/day1/` にコピーします。

送る文面の @ で指すファイルは、どれも `成果物/day1/` のような自分の配布フォルダの中のファイルです。`成果物/合流用/` の中のファイルは、コピーするまでは文面で指しません。コピーした先のファイルを指してください。

フォルダごと重ねずに、ファイルを選んでコピーしてください。URLが開けないときは、講師がZoomのチャットでファイルを直接送ります。受け取ったファイルは、同じ場所にコピーしてください。

早く終わった方へ

追加と考察

非機能要件の章に、メモにない数字が入っていないかを見ます。応答3秒、同時接続100人、稼働率99.9%のような数字は、要件定義書らしく見えるのでAIが入れがちです。メモにある速さの言い方を、どの数字にして誰に確かめるかを書き出してください。

発展課題

用語集の章を足し、ヒアリングメモで言い方が揺れている語をそろえさせます。社内の用語集に加えて、2019年のKINTAI改修で書かれた過去の要件定義書とも突き合わせます。

Claudeに送る文面

@成果物/day1/要件定義書_v0.1.md
@資料/参考/用語集.md
@資料/発展/D1-3_過去の要件定義書_抜粋_KINTAI2019.md

要件定義書に「用語集」の章を足してください。ヒアリングメモで言い方が揺れている「工数」「作業時間」「稼働」を中心に、同じものを指す言葉を1つにそろえ、本文もそろえた言葉に直してください。
資料/参考/用語集.md や過去の要件定義書と定義が違う語は、どちらを採るかを書かずに要確認にしてください。

要確認になった語は、誰が決めることなのかを考えます。過去の文書を書いた部署か、今回の依頼を出した情報システム課か、使っている現場か。

08

D1-4 抜け漏れ探し

[25min] 個人演習 座学 [3min]

この演習の [25min]は、座学 [3min]と、STEP 1〜4の [22min]です。座学は演習の最初に入ります。

ねらい

AIに頼む前に、自分でデータを開いて要件の抜けを探します。AIの抜け漏れ探しと並べて、どちらが何を拾うかを知ります。

できあがりの目安

- ☐ 自分で見つけた抜けが1件以上、要件定義書v0.2と要件一覧に入っている
- ☐ `成果物/day1/確認事項.md` に、相手、聞くこと、なぜ聞くか の3列で5件以上ある
- ☐ `成果物/day1/要件定義書_v0.2.md` がある。`要件定義書_v0.2.docx` は、時間が足りなければDay2の振り返りで作ってよい

使うファイルと作るファイル

区分	ファイル	使い方
使う	`資料/データ/勤怠工数CSV_202609.csv`	2026年9月の全員の勤怠と工数。LibreOffice Calcで開きます
使う	`資料/データ/問い合わせ履歴_KINTAI_202510-202609.csv` `資料/参考/月次締め運用手順書.md`	STEP 2 でClaudeに渡し、抜けの根拠にします
使う	`成果物/day1/要件定義書_v0.1.md`	D1-3 で作ったもの
作る	`成果物/day1/メモ_D1-4.md`	自分で見つけた抜けと、比べた結果
作る	`成果物/day1/要件定義書_v0.2.md` `成果物/day1/要件定義書_v0.2.docx`	抜けを反映したもの。Day2の入力になります
作る	`成果物/day1/確認事項.md`	ヒアリングで確かめることの一覧
直す	`成果物/day1/要件一覧.md`	見つけた抜けを要件として足します。Day3の対応表に入ります

CSVを開いて抜けを1つ探す → Claudeに4観点で探させる → 自分とAIの発見を並べる → v0.2と確認事項を作る

この演習は、1つの会話で最後まで進めます。STEPの文面は前のSTEPの続きとして送るので、途中で新しい会話にしないでください。演習が変わったら、新しい会話を始めて構いません。

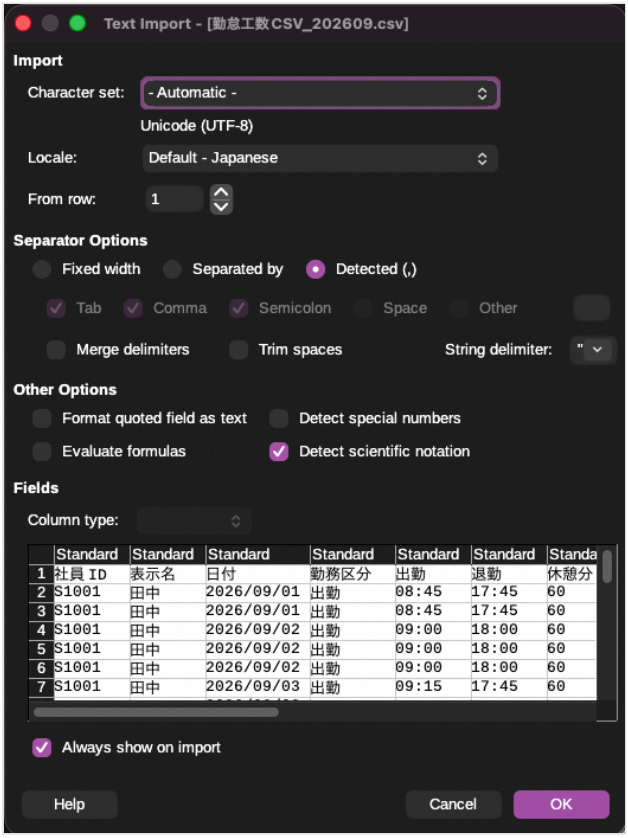
STEP 1 勤怠CSVからの抜け探し

AIを開かない [8min]

1. エクスプローラーで **資料** → **データ** を開き、**勤怠工数CSV_202609.csv** を右クリック → **プログラムから開く** → **LibreOffice Calc** を選びます。

2. **テキストのインポート** の画面が出たら、**文字エンコーディング** を **Unicode (UTF-8)**、区切りのオプションを **コンマ** にして **OK** を押します。

3. 左端の行番号の1を押して1行目を選び、**データ** → **オートフィルター** を押します。列ごとに絞り込めるようになります。



	Standard	Standard	Standard	Standard	Standard	Standard	Standa
1	社員 ID	表示名	日付	勤務区分	出勤	退勤	休憩分
2	S1001	田中	2026/09/01	出勤	08:45	17:45	60
3	S1001	田中	2026/09/01	出勤	08:45	17:45	60
4	S1001	田中	2026/09/02	出勤	09:00	18:00	60
5	S1001	田中	2026/09/02	出勤	09:00	18:00	60
6	S1001	田中	2026/09/02	出勤	09:00	18:00	60
7	S1001	田中	2026/09/03	出勤	09:15	17:45	60

図12：CSVをCalcで開くと、**テキストのインポート** の画面が表示されます。英語表示のMac版ではText Importと出ます

列	中身
社員ID・表示名	社員IDはSで始まります
日付・勤務区分	1日に複数のプロジェクトを付けた日は、同じ日付の行が複数あります

列	中身
出勤・退勤・休憩分・実働分	KIN020で入れた勤怠。実働は分で入っています
PJコード・作業区分・作業分	KIN030で入れた工数。作業分も分です
備考	入力した人が自由に書いた一言

列の中身を見ながら、要件定義書v0.1に書いていないことが起きている行を1つ探します。

見つけたら、メモ帳で`成果物/day1/メモ_D1-4.md`を作って3行で書きます。1行目は社員IDと日付、2行目は何が起きているか、3行目は要件定義書のどの章に書くべきことかです。

ヒント −

9月の営業日は19日で、9/21・22・23は休日です。日付で9/22を絞り込んでみてください。もう1つは備考の列です。「修正」という言葉が入っている行を探してください。

CSVは保存せずに閉じます。閉じるときに保存を聞かれたら`保存しない`を選びます。`資料/`の中は書き換えない約束です。

STEP 2 Claudeによる抜け漏れ探し [3min]

自分の発見はまだ見せません。Claudeが何を拾うかを、自分の発見と比べるためです。

勤怠CSVに加えて、ヘルプデスクの問い合わせ履歴と、管理部の月次締め運用手順書も渡します。困りごとが実際に起きていたかと、手作業でしのいでいるところが、抜けの根拠になります。

Claudeに送る文面

```
@成果物/day1/要件定義書_v0.1.md
@資料/データ/勤怠工数CSV_202609.csv
@資料/データ/問い合わせ履歴_KINTAI_202510-202609.csv
@資料/参考/月次締め運用手順書.md

要件定義書の抜け漏れを、例外・運用・権限・データ移行 の4つの観点で探してください。
- 観点ごとに表にし、列は 観点 / 抜けていること / 根拠 / 誰に確認するか
- 根拠は、CSV なら社員ID と日付、問い合わせ履歴なら受付番号、手順書なら手順の番号、ヒアリングメモならファイル名と行番号
- 要件定義書はまだ書き換えないでください
```

STEP 3 自分の発見との比較

AIを開かない [6min]

`メモ_D1-4.md` に、自分の発見とClaudeの指摘を並べる表を書きます。表は5行までです。このSTEPではClaudeに新しく頼みません。STEP 2の応答の表は、読んで写すだけにします。

1. 下の表の形をメモ帳に貼ります。
2. 自分で見つけた抜けを、1件1行で先に書き、自分の列に○を付けます。Claudeの指摘にも同じものがあれば、Claudeの列にも○を付けます。
3. 残りの行に、Claudeの指摘から要件定義書に入れたいものを選んで書き、Claudeの列に○を付けます。抜けていることの列は、Claudeの応答の表からコピーして貼って構いません。5行を超える分は書きません。
4. 行ごとに、要件にするか、運用で済ませるかを書きます。
5. 表の下に、自分とClaudeがそれぞれどんな抜けを拾ったかを1行で書きます。

メモに貼る表の形

```
| 抜けていること | 自分 | Claude | 要件にするか、運用で済ませるか |
|---|---|---|---|
| | ○ | | |
```

> **Tips** 表の下の1行は、どこを見て見つけた抜けかで考えると書けます。文書を読んで見つかる抜けと、データを開かないと見つからない抜けがあります。

STEP 4 v0.2と確認事項の作成

[5min]

STEP 3 の表で「要件にする」とした抜けを、文面の `<ここに…>` に自分の言葉で書いてから送ります。反映する抜けは、自分で見つけた1件だけでも構いません。Claudeの指摘から採るものも、そのまま貼らずに書き直してください。自分の言葉で書き直せない指摘は、採る理由がまだ自分の中にないので、今回は採りません。

Claudeに送る文面 メモ帳で<ここに…>を埋めてから貼る

```
@成果物/day1/要件定義書_v0.1.md
@成果物/day1/要件一覧.md

次の抜けを要件定義書に反映して、成果物/day1/要件定義書_v0.2.md として保存してください。v0.1 は書き換えずに残してください。
文書情報の版を 0.2 にし、改訂履歴に 0.2 の行を足してください。
<ここに、反映すると決めた抜けを自分の言葉で書いて貼る>
```

同じ抜けを 成果物/day1/要件一覧.md にも、続きの要件IDで足してください。根拠は、CSV なら社員ID と日付、メモならファイル名と行番号にしてください。

最後に、v0.2 の（要確認）を全部拾い、成果物/day1/確認事項.md に表でまとめてください。列は 相手 / 聞くこと / なぜ聞くか の3つです。

続けてWordにします。時間が足りなければ、この変換はDay2の振り返りの時間に回して構いません。変換の前に、同じ名前のファイルをWriter・Calc・Impressで開いていたら閉じます。開いたままだと書き込めず、変換が止まります。

Claudeに送る文面

成果物/day1/要件定義書_v0.2.md を tools/md_to_docx.py で Word にして、成果物/day1/要件定義書_v0.2.docx に保存してください。

1. コマンドの確かめが出たら、`成果物/day1/要件定義書_v0.2.md` を `成果物/day1/要件定義書_v0.2.docx` にする形になっているかを見て `一度だけ許可` を押します。

2. エクスプローラーで `成果物` → `day1` を開き、`要件定義書_v0.2.docx` を右クリック → `プログラムから開く` → `LibreOffice Writer` を選びます。

できていれば完了

- [ ] 自分で見つけた抜けがv0.2に入っていて、文書情報の版が0.2になっている
- [ ] 同じ抜けが `要件一覧.md` にも、要件IDと根拠つきで入っている
- [ ] `確認事項.md` が5件以上あり、「なぜ聞くか」が空いていない
- [ ] 変換した方は、`要件定義書_v0.2.docx` がWriterで開ける

うまくいかないとき

CSVを開いたら文字化けした

`テキストのインポート` の画面で `文字エンコーディング` が `Unicode (UTF-8)` になっていなかったはずです。いったん閉じて、保存せずに開き直します。ダブルクリックでメモ帳が開いた場合も、右クリック → `プログラムから開く` → `LibreOffice Calc` でやり直してください。

LibreOfficeで開けない

Calcが起動しないときは、事前準備のStep 5で `tools/sample/確認用.pptx` が開けたかを思い出してください。開けていなければLibreOfficeの導入が済んでいません。その間は、CSVをメモ帳で開いて9/22の行をCtrl+Fで探しても進められます。

AIがメモにない要件を足してくる

この演習は、メモにない抜けを探すのが目的なので、メモにないこと自体は問題ではありません。見るのは根拠です。CSVの行も、メモの行も示せない指摘は、「どのデータからそう言えますか」と聞き返し、答えられなければ採りません。

v0.1が書き換えられた

差分の表示で、変わったファイルが`要件定義書_v0.1.md`になっていたら、「v0.1は元に戻し、変更はv0.2に書いてください」と送ります。

変換で「LibreOfficeで開いているファイルを閉じてから」と出た

変換先のファイルをWriter・Calc・Impressで開いたままにしていると、書き込めずに止まります。開いているウィンドウを閉じて、ファイル名は変えずに「もう一度同じ変換をしてください」と送ってください。

前の演習が終わらず、使うファイルがない

講師が配るDay1当日の合流版で追いつけます。`要件一覧.md`と`要件定義書_v0.1.md`が入っています。

1. 講師がZoomのチャットに合流版のURLを貼ります。ブラウザで開くと`合流_Day1当日.zip`がダウンロードされます。
2. エクスプローラーで`ダウンロード`を開き、ZIPを右クリック → `すべて展開`を選びます。`参照`で`idup-handson` → `成果物` → `合流用`を選び、`展開`を押します。
3. `成果物/day1/`に自分の途中のファイルがあれば、コピーの前に名前の末尾へ`_自分`を付けます。`要件一覧.md`なら`要件一覧_自分.md`です。上書きで消さないためです。
4. `成果物` → `合流用` → `day1_当日` → `成果物` → `day1`を開き、`要件一覧.md`、`要件定義書_v0.1.md`を、配布フォルダの`成果物/day1/`にコピーします。

送る文面の @ で指すファイルは、どれも`成果物/day1/`のような自分の配布フォルダの中のファイルです。`成果物/合流用/`の中のファイルは、コピーするまでは文面で指しません。コピーした先のファイルを指してください。

フォルダごと重ねずに、ファイルを選んでコピーしてください。URLが開けないときは、講師がZoomのチャットでファイルを直接送ります。受け取ったファイルは、同じ場所にコピーしてください。

早く終わった方へ

追加と考察

CSVで見つかる汚れを、要件として扱うべきものと、運用で済むものに分けます。休日の出勤は、画面の入力チェックで止めるのか、管理部が月末に確かめるのか。締めのあとの修正は、画面で直せるようにするのか、管理部に依頼する運用にするのか。分けた理由を`メモ_D1-4.md`に書きます。

発展課題

締めたあとの修正が、毎月どのくらい、どんな形で起きているかを見ます。`資料/発展/D1-4_8月分の締め後修正依頼メール.md`と`資料/データ/過去分/勤怠工数CSV_202608.csv`を並べ、メールで届いた依頼のうち、CSVの備考に修正の跡が残っているものを数えてください。CSVはCalcで開き、備考の列をオートフィルターで絞ると探せます。

数えた結果から、`確認事項.md`に何を足すかを、相手と聞くことまで自分で書いてください。

そのあと、確認事項を、ヒアリングの相手ごとの質問票にさせます。

Claudeに送る文面

@成果物/day1/確認事項.md

確認事項を、相手ごとの質問票にしてください。田中課長、鈴木主任、佐藤さん、情報システム課のように相手ごとに見出しを分け、1問ずつ「はい・いいえ」か数字で答えられる聞き方にしてください。
保存先は 成果物/day1/質問票.md です。

「はい・いいえ」で答えられない質問が残ったら、それは会って聞くべき質問です。

09

よくある質問

演習が時間内に終わりませんでした

途中までででも、ファイルは消さずに残してください。D1-3とD1-4に追いつくための当日の合流版と、Day2の最初に全員へ配る完了状態があります。どちらも、講師がZoomのチャットにURLを貼り、ダウンロードして`すべて展開`し、中のファイルを`成果物/day1/`にコピーしてください。自分の途中のファイルは、末尾に`_自分`を付けて残してください。手順はD1-3とD1-4の「うまくいかないとき」にあります。自分の続きをやりたい方は、研修のあとに同じ手順で進められます。

「許可」を何度も求められて進みません

D1-3 からは、モードを`編集を受け入れる`に切り替えてください。`編集を受け入れる`では、ファイルの書き換えと、フォルダの中でのファイルの作成・移動・コピー・削除は、確かめなしで進みます。pythonで動かす変換ツールなどのコマンドだけが、確かめの対象です。確かめが減るぶん、作業のあとに差分の表示で中身を見てください。

Claudeの返事が途中で止まりました

少し待っても動かないときは、「続けてください」と送ります。ファイルが途中まで書かれていたら、「`成果物/day1/要件一覧.md`を読み直して、途中から書き足してください」のように、そのファイルの名前を入れて送ってください。

資料フォルダのファイルを書き換えてしまいました

`資料/`は`.claude/settings.json`の設定で、Claudeのファイル編集とコマンドからの書き込みを止めてあります。pythonのスクリプトは止められないので、コマンドを読んで、`資料/`に書き込む内容のときは拒否します。ただし、自分でメモ帳から上書き保存したときの書き換えは止められません。配布ZIPを別の場所に展開し直し、書き換えたファイルだけを`資料/`に上書きで戻してください。`成果物/`は触らなくて構いません。ZIPは教材サイトの配布物から、いつでもダウンロードできます。

根拠の行番号がメモ帳の行番号と合いません

まず、メモ帳の下の帯の右側を見てください。`書式付き`と出ていたら押して、`マークダウン構文`に切り替えます。書式付きのままだと空行が行数に入らず、行番号がずれます。

次に、`表示`で`右端で折り返す`の印を外してください。この2つを直しても合わなければ、Claudeが数え違えています。「根拠の行番号を、ファイルを読み直して数え直してください」と送ります。

自分の案件のメモで試してもよいですか −

研修中は配布の架空データだけを使ってください。自分の案件で回す手順は、Day3のあとに配る持ち帰り演習集にあります。社内のルールでAIに渡してよい情報を確かめてから試してください。

Javaが分かりませんが、ついていけますか −

研修でコードを読む場面はありません。KINTAIがJavaで作られているのは題材の背景としてだけで、演習で扱うのはヒアリングメモ、要件定義書、画面設計書、画面のモックです。

© Givery, Inc.

※題材のサンプル情報サービス株式会社、取引先、人物、金額、勤怠と工数のデータは、研修のために作った架空のものです。