

TRAINING

AI協働開発ハンズオン講座

機電系トラック Day2

バイブコーディング入門・プロンプト・データの下処理



本資料の二次転載禁止について

本資料の二次転載を**禁止**します。

本資料は株式会社ギブリーが作成した著作物です。受講者ご本人の学習および業務での参照に限りご利用いただけます。社外への配布、SNS等への掲載、第三者への共有はご遠慮ください。

研修中の画面撮影・録画も、資料が写り込む範囲ではお控えください。

本日の進み方

今日は手を動かす時間が全体の8割です



01

Day1の復習と環境確認

1週間前の環境をもう一度動かして、Day1の要点を戻します

Day1で扱ったこと

今日はこの4点を前提に手を動かします

生成AIの仕組み

文章の続きを確率で選ぶ道具です。計算結果と事実は、AIの外で別に確かめます

得意と苦手

下書き・整形・言い換えは得意です。社内の最新情報と、正確でなければ困る数値は苦手です

機密の伏せ方

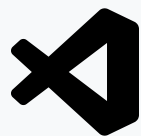
客先名・製品名・プロジェクト名・取引先名は貼らず、業界と工程の粒度に丸めます

Copilotの基本操作

VSCodeの右側のチャットに日本語で頼み、返ってきたコードを貼って動かします

3番目の機密の伏せ方は、午後の業務テーマの言語化でもう一度戻ってきます。

書く場所、頼む相手、動かす場所の3つだけです



VSCode

ファイルを作って書く場所です。今日はここでindex.htmlを作ります



GitHub Copilot

VSCodeの右側で日本語の依頼を受け取ります。返ってくるのはコードです



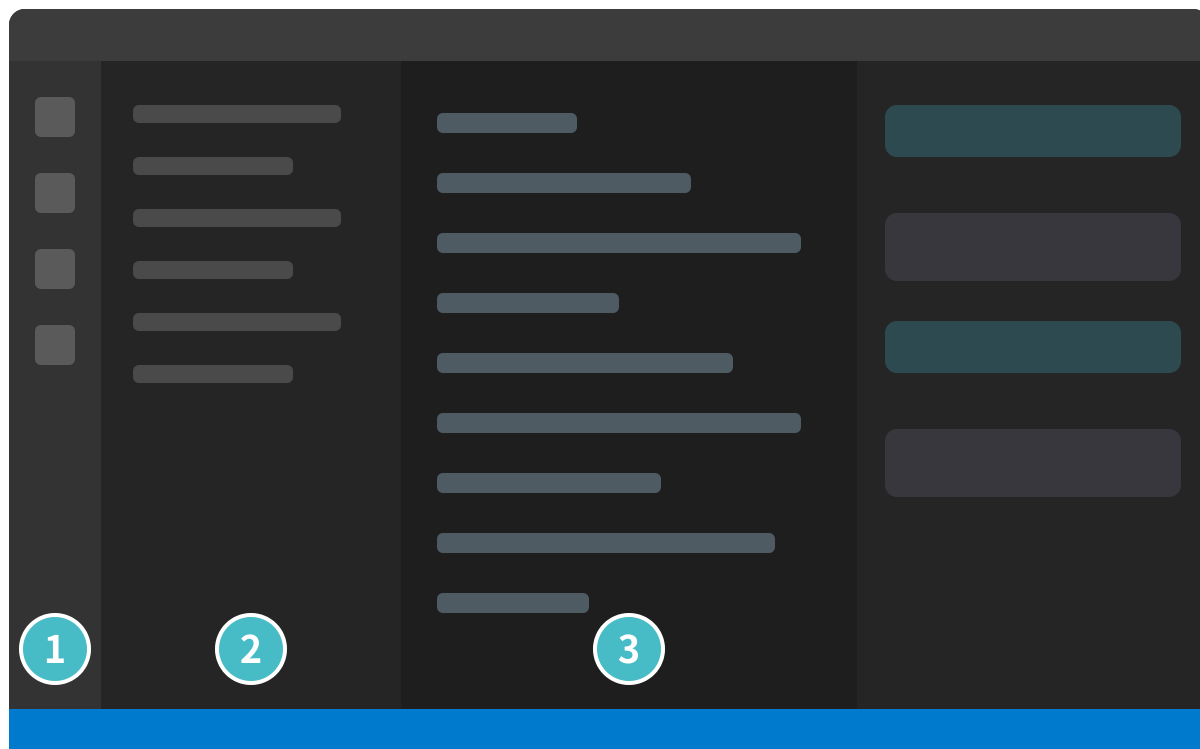
Edge

保存したHTMLを開いて動かします。表示の確認はこちらでします

この3つ以外はインストールしません。追加の環境構築は今日ありません。

今日の入口の画面

この4か所が見えていれば準備は整っています



1 アクティビティバー

左端の縦の並びです。いちばん上の書類が重なったアイコンを押すと、フォルダの中身が出ます

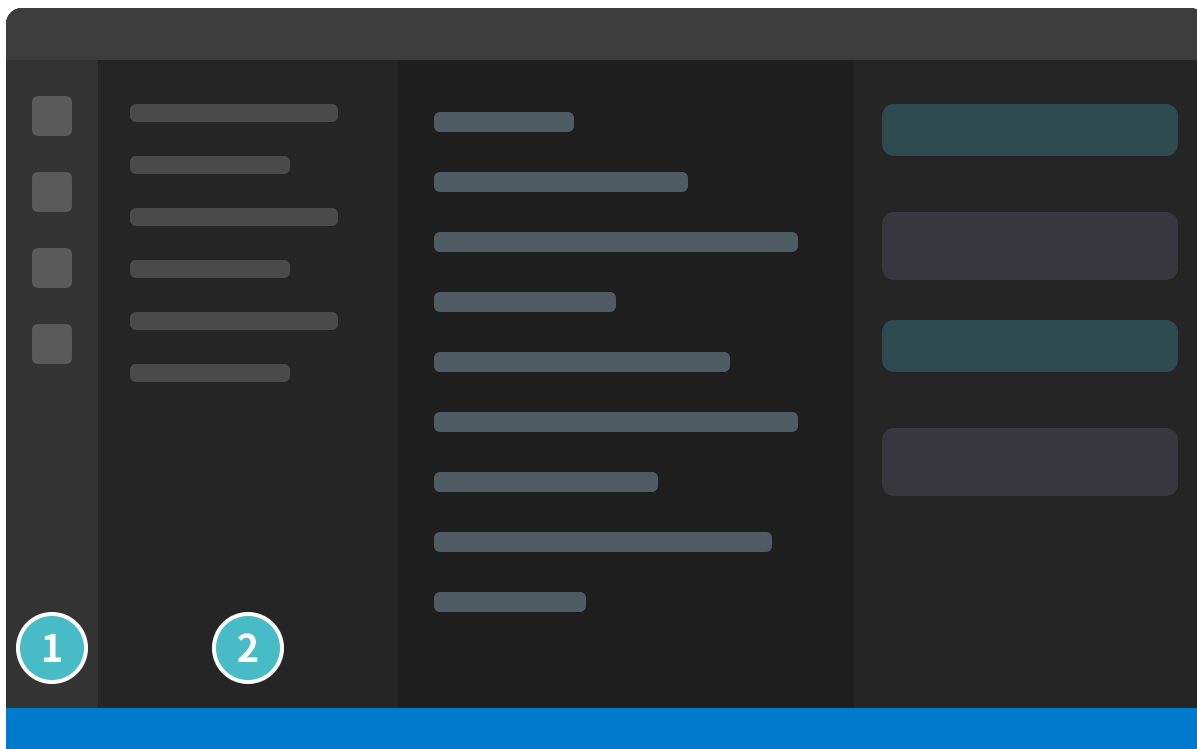
2 エクスプローラー

開いているフォルダの中身です。ここに day2 と出ていれば、今日の作業フォルダを開けています

3 エディタ

ファイルの中身を書く場所です。Copilotから返ってきたコードは、ここに貼り付けます

今日の入口の画面（続き）



1 チャットパネル

右側です。ここに日本語で頼みます。出ていない場合は右上のCopilotアイコンで開きます

2 ステータスバー

最下部です。左下のアカウントアイコンにGitHubのユーザー名が出ていればサインイン済みです

5つ通せば、この後の130分でつまずきません

どんな場面か

前回から1週間空いています。サインインが切れていたり、チャットパネルが開かなくなっていたりする方が毎回出ます。この後の130分は全員が同じ画面で進むので、先にそろえます。ここで直しておけば、演習中に環境の話で止まる時間がなくなります。

やること

- 1 VSCodeを起動します。スタートメニューか、デスクトップのアイコンから開きます
- 2 画面右上のCopilotアイコンをクリックし、右側にチャットパネルが開くことを確認します
- 3 画面左下のアカウントアイコンを見ます。GitHubのユーザー名が出ていなければクリックしてサインインし直します

環境の動作確認（続き）

やること

- 1 チャットに「こんにちは。今日は何ができますか」と入力してEnterを押し、日本語の返事が返ることを確認します
- 2 デスクトップに day2 という名前のフォルダを作り、VSCodeの「ファイル」>「フォルダーを開く」でそのフォルダを開きます

できあがるもの

成果物はありません。デスクトップに day2 フォルダができていて、VSCodeのエクスプローラーに DAY2 と表示されていれば達成です。

環境確認の完了条件

4つ全部そろって完了です

達成の目安

- 右側にチャットパネルが開いている
- 左下のアカウントアイコンにGitHubのユーザー名が出ている
- 日本語で入れた質問に、日本語の返事が返ってきた
- エクスプローラーの見出しに DAY2 と出ている

つまづいたら

- サインインの画面から進まない。ブラウザが開いて承認を待っている状態です。開いたタブで承認します。承認画面が出てこない場合は通信の制限なので、手を挙げてください
- Copilotのアイコンが見当たらない。VSCodeのバージョンが古い可能性があります。1.116以降でチャットが標準搭載です
- フォルダを開かずにファイルだけ開いている。この状態だと後で保存先が分からなくなります。手順5からやり直します
- 返事が返ってこない。まず画面右下の通知を見ます。戻らない場合は隣の画面を見ながら先に進んでかまいません
- day2 フォルダを作った場所が分からなくなった。デスクトップ以外に作ってしまった場合は、作り直したほうが早いです

環境確認の完了条件（続き）

時間が余ったら

- チャットに「HTMLで画面に今日の日付を出すコードを書いて」と入れて、返ってきたコードを眺めておいてください。午前の演習で使う形と同じです
- 返ってきた文章の書き出しが、Day1で試したときと同じ調子かどうかを見比べてください
- エクスプローラーで右クリックして、どんなメニューが出るかを先に見ておいてください。次の章でここから新しいファイルを作ります

使うフォルダは2つ

もらうフォルダと、自分で作るフォルダを分けます

配布フォルダ

デスクトップに展開すると 機電系_Day2 という名前で作れます。01_データ読み込み.html、設備データ_2026Q1.xlsx、README.txt と、xlsx.full.min.js・papaparse.min.js・chart.umd.jsが入った lib フォルダが並びます

day2 フォルダ

自分でデスクトップに作ります。今日作るものは全部ここに入れます。午前に index.html、午後に prompt.txt、clean.csv、chart.html、theme.txt が増えます

2つの行き来

午後のグラフの時間に、配布フォルダの lib フォルダの中にある chart.umd.js を day2 フォルダにコピーします。行き来はこの1回だけです

Day3でもこの day2 フォルダを使います。研修が終わっても消さないください。

動かないときの確認先

症状ごとに見る場所を決めておくと、自力で戻せます

症状	見るところ	対処
Copilotのアイコンが無い	画面右上のアイコンの並び	VSCoDeを開き直します。1.116より古い場合は手を挙げてください
サインインを求められる	画面左下のアカウント	GitHubアカウントで入り直し、開いたブラウザで承認します
返事が返らない	画面右下の通知	通信の制限が原因のことがあります。戻らなければ講師へ
前回のフォルダが無い	ファイル>最近使用した項目	day2 フォルダを作り直して開きます
表示が全部英語	日本語パックの有効化	日本語パックを有効にしてVSCoDeを再起動します

該当する症状が無い方は、この表は見なくて大丈夫です。同じ表が受講者資料にも載っています。 出典:
https://code.visualstudio.com/updates/v1_116

ここまでと、この先

ここまでにしたこと

全員のVSCodeでチャットパネルが開き、日本語の返事が返るところまで戻しました。
デスクトップに day2 フォルダができ、配布フォルダとの役割も分けました。Day1の4点も確認しました。



これからやること

ここからの130分で、日本語で頼んで動くものを1つ作ります。題材は自分で選びます。ゲームでもかまいません。コードは1行も書きません。手を動かす時間が大半で、講師は席を回ります。



そのあとどこで効くか

ここで身につく「伝えて、動かして、ずれを言う」の回し方が、午後のExcel整形とDay3のミニアプリ作成でそのまま使う進め方になります。午前は、その回し方に慣れるための時間です。

02

バイブコーディング入門

日本語で頼んで、ブラウザで動くものを1つ作れるようになります

設計を固める前に動かして、触ってから決めます

これまでの進め方

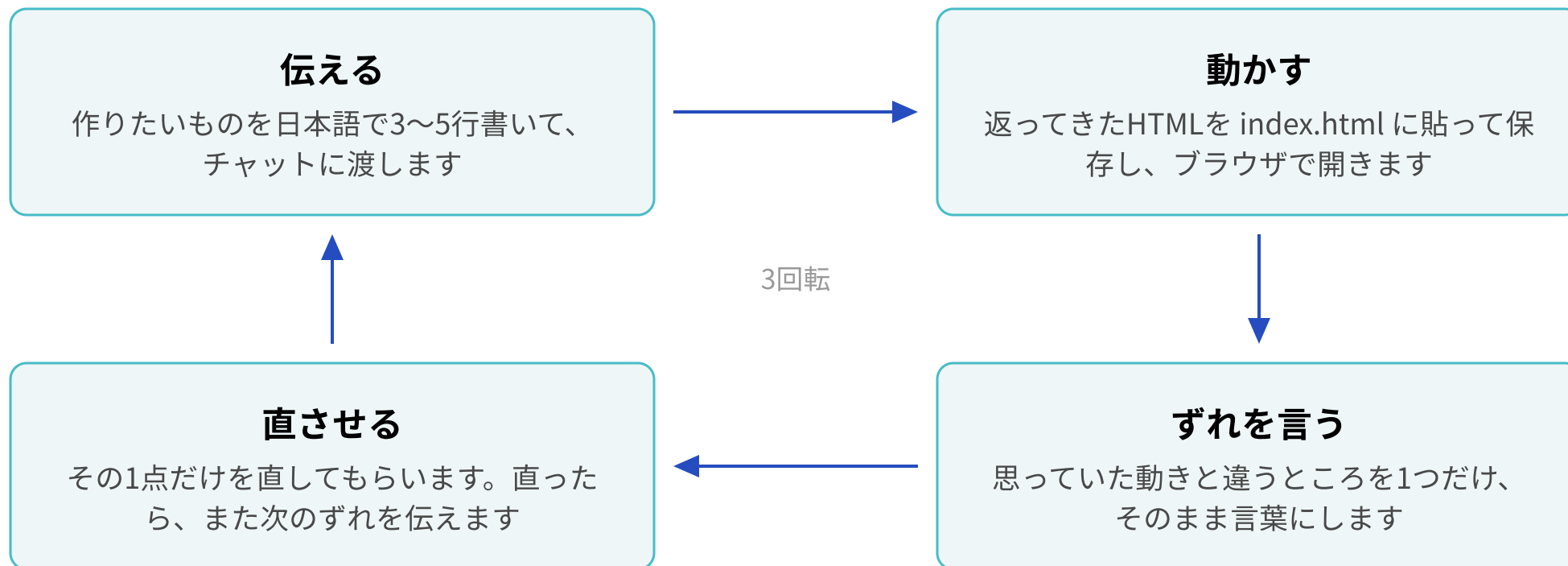
仕様を決める、設計する、作る、動かす、の順に進みます。動くものが目の前に出るのは最後です。途中で思っていたのと違うと気づいても、前に戻りにくくなります

バイブコーディング

日本語で伝える、動かす、触って決める、の3つで回します。最初の1回は5分ほどで画面が出ます。違うと思ったら、その場で言い直します

作るときの手順

1回で完成させません。3回転で形にします



1回転目は動けば合格、2回転目で見た目、3回転目で機能を1つ足す。この配分で進めてください。

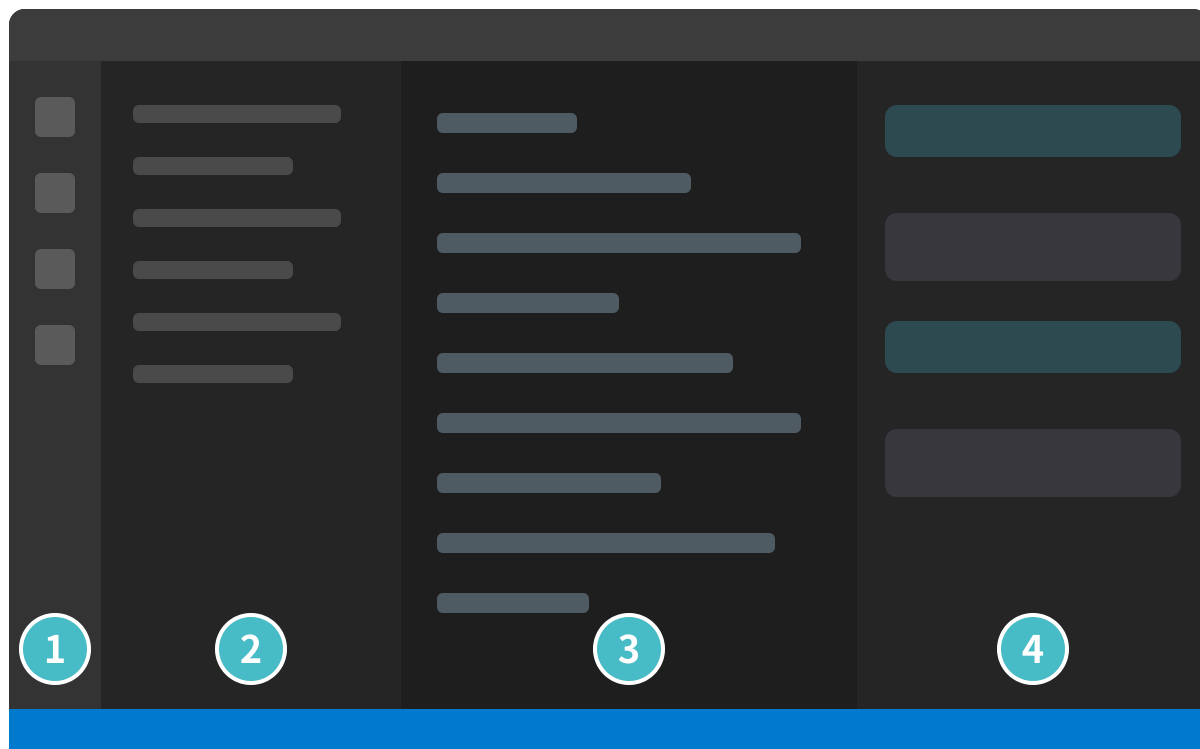
この4つから選んで大丈夫です

方向	作るもの	動いたと分かる合図
数を数える	部品の員数カウンター。ボタンを押すと数が1増え、リセットで0に戻ります	押すたびに画面の数字が変わる
単位を換算する	mmとinch、N・mとkgf・mの換算。左に入れると右に出ます	入力を変えると右の数字が追う
確認漏れを防ぐ	始業前点検のチェックリスト。チェックした数と残りを表示します	チェックすると残数が減る
ゲーム	もぐらたたき、数当て、じゃんけん。勝ち負けがすぐ出ます	得点が増える、勝敗が出る

Day1の最後をお願いした「作ってみたいもの」があればそれを使ってください。実際の型番・公差・客先名は使わず、架空の数字にしてください。

ファイルを作る場所

新しいファイルはエクスプローラーの中で作ります



- 1 アクティビティバー**
左端のいちばん上、書類が重なったアイコンを押すとエクスプローラーが開きます
- 2 エクスプローラー**
DAY2 の見出しの下で右クリックし、「新しいファイル」を選びます。ここが index.html を作る場所です
- 3 エディタ**
作ったファイルがここを開きます。貼り付けてから Ctrl+S で保存します
- 4 チャットパネル**
返答の右上にコピーのボタンがあります。コード枠の中を手でなぞらず、このボタンを使います

ブラウザで動かすまで

保存とダブルクリックの2つで画面が出ます

1

フォルダを開く

VSCodeで「ファイル」>「フォルダーを開く」を選び、デスクトップの day2 フォルダを開きます

2

index.html を作る

エクスプローラーで右クリック>新しいファイル。index.html と入力してEnterを押します

3

コードを貼る

Copilotの返答の右上のコピーを押し、エディタに貼って Ctrl+S で保存します

4

ブラウザで開く

デスクトップの day2 フォルダを開き、index.html をダブルクリックします。VSCodeの中では動きません

5

直したら再読み込み

保存したあと、ブラウザで F5 を押すと直した内容が出ます

Windowsで拡張子が非表示だと index.html.txt になっていることがあります。エクスプローラーの「表示」>「ファイル名拡張子」にチェックを入れて確認してください。

何を作るか、画面に何を出すか、操作したら何が起きるか

チャットにそのまま貼って使える例です。3～5行あれば1回目は足够了。最後の1行は今日ずっと付けてください。

もぐらたたきゲームをHTMLで作ってください。

- ・ 3×3のマスに、30秒間ランダムでもぐらが出ます
- ・ もぐらをクリックすると得点が1増えます
- ・ 残り時間と得点を画面の上に表示します
- ・ 1つのHTMLファイルにまとめ、外部ファイルの読み込みはしないでください

最後の1行を外すと、外部のURLから部品を読み込むコードが返ってくることがあります。社内の通信制限で止まると画面が真っ白になります。

この演習の到達点

見た目は粗くてかまいません。動けば達成です

ブラウザで開いて 操作すると何かが変わる

作るものが違ってても、この地点に着けば同じです。配色や配置の完成度は今日は問いません

最初の1回でだいたい画面は出ます。残りの時間は、出てきたものを自分が見えるものに寄せる時間です。

直らないときの伝え方

症状をそのまま書くと、直す確率が上がります

伝わりにくい

「動きません」「バグってます」だけを送ります。AIはどこを直せばよいか分からないので全体を書き直し、動いていた部分まで壊れます

伝わる

「もぐらをクリックしても得点が0のままです。残り時間は動いています」のように、動いている部分と動かない部分を分けて書きます

直しの頼み方

1回に1つだけ直します

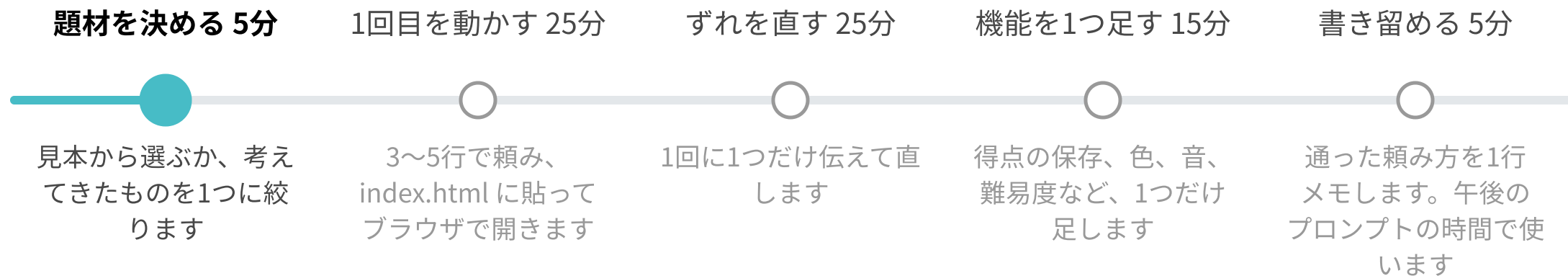
止まったときに、そのまま貼れる形です。今の状態、直したいこと、守ってほしいこと、の順に書きます。

いま作っているもぐらたたきで、もぐらをクリックしても得点が0のままです。
残り時間のカウントダウンは動いています。
得点が増えない原因だけを直してください。ほかの動きは変えないでください。
直したHTML全体をもう一度出してください。

「ほかの動きは変えないでください」を付けないと、動いていた部分まで書き直されることがあります。

演習の時間の使い方

最初の30分で1回動かします



30分たっても画面が出ない方は手を挙げてください。全体を止めて、原因を3分だけ扱います。

設計書を書かずに動くものが出る感覚を、自分の手で1回通します

どんな場面か

コードを書かずに動くものが出てくる感覚は、話を聞いても分かりません。1回自分で通しておく、午後にExcelを前にしたときに「これも頼めば済むのでは」と気づけるようになります。Day3のミニアプリ作成も、この進め方で作ります。

やること

- 1 題材を1つ決めます。見本の4つか、Day1の最後に考えてきたものから選びます。5分で決めて先へ進みます
- 2 3～5行で頼みます。何を作るか、画面に何を出すか、操作したら何が起きるか、1つのHTMLにまとめる、の4つを書きます
- 3 返ってきたコードを day2 フォルダの index.html に貼り、Ctrl+S で保存してダブルクリックで開きます

演習 自分の題材で作る（続き）

やること

- ① 動いたら、その時点のファイルを index_ok.html という名前でもう1つ保存します。作り込んで壊れたときに、ここへ戻れます
- ② 違うところを1回に1つだけ伝えて直します。まとめて頼むと、動いていた部分が壊れます
- ③ 最後に、通った頼み方を prompt.txt に1行だけ書き留めます

できあがるもの

day2 フォルダの index.html です。あわせて index_ok.html と prompt.txt が残ります。提出はありません。

完成の目安と逃げ道

3つそろえば完成です。見た目は問いません

達成の目安

- ブラウザで開いて画面が出る
- クリックなどの操作で、画面の何かが変わる
- 自分が頼んだ直しが1つ以上反映されている

つまづいたら

- 画面が真っ白。まず Ctrl+S を押します。それでも白いなら F12 の Console の赤い文字を貼ります
- ファイルを開くとコードが文字のまま出る。ファイル名が index.html.txt になっています。拡張子の表示をオンにして名前を直します
- 直したのに変わらない。保存していないか、ブラウザで F5 を押していません
- 作り込んだら壊れた。index_ok.html の中身を index.html に貼り直して戻ります
- 題材が大きすぎて終わらない。1画面で終わる大きさに削ります。項目を減らす、機能を1つに絞る、で戻せます
- コードを貼る場所が分からない。エクスプローラーに DAY2 と出ているか確認します。出ていなければフォルダを開き直します

完成の目安と逃げ道（続き）

時間が余ったら

- 得点や入力内容をブラウザに保存して、画面を閉じてでも残るようにする
- スマートフォンの画面幅で崩れないようにする
- 難易度や単位の切り替えを付ける
- 配色を職場の掲示物に合わせて、印刷しても読める色にする

Q. 作ったものをブラウザで開いたら、画面が真っ白でした。最初にやることはどれですか

A Copilotに最初から作り直してくださいと頼む

B F12で出るエラー文をそのままCopilotに貼る

C 別のブラウザで開き直す

D パソコンを再起動する

正解は次のページで確認します。まず自分で考えてみてください。

正解はB。エラー文をそのまま渡すのが最短です

F12で出るエラー文をそのままCopilotに貼る

B

画面が真っ白になるのは、途中で処理が止まっているためです。F12で開く Console に、止まった場所が赤い文字で出ています。その1行をそのまま貼ると、原因の箇所だけを直してもらえます。意味を読み取る必要はありません。

A

最初から作り直してくださいと頼む
動いていた部分まで書き直され、別の不具合が増えます。作り直しは最後の手段です

C

別のブラウザで開き直す
原因がコードにある場合は、どのブラウザで開いても同じ結果になります

D

パソコンを再起動する
保存忘れやコードの誤りは再起動では直りません。時間だけがかかります

赤い文字が出ていないのに白いままなら、保存忘れかファイル名の誤りです。Ctrl+S と拡張子を確認してください。

休憩

10分休憩します

休憩 10min

再開時刻は講師が板書します。席を離れる前に、index.html を Ctrl+S で保存してください

午後の1本目はプロンプトの基礎です。作ったものは閉じずに残しておいてください。

午後への引き継ぎ

午前にうまくいった頼み方を、貼って使える形に残します

ここまでにしたこと

午前は、作りたいものを日本語で書いてCopilotに渡し、返ってきたHTMLをブラウザで動かすところまでを一通り通しました。1回で思いどおりに動いた依頼と、5回書き直した依頼が混ざっているはずです。



これからやること

その差がどこから来たのかを分解します。前半60分で頼み方の型を確認し、後半40分の演習で、自分がよく使う言い回しをprompt.txt という1つのファイルにまとめます。



そのあとどこで効くか

午後のデータ整形は、このprompt.txt を貼るところから始まります。Excelの欠損に勝手な数字を入れさせないための1行も、ここで作ります。Day3のデータ分析とレポート作成でも同じ文を使い回します。

03

プロンプトの基礎

頼み方の4要素を知り、自分が繰り返し使う依頼文を1本持ち帰れる状態になります

書かなかった条件は、AIが自分で決めて先に進みます

書かなかったこと	AIが自分で決めたこと	返ってきたもの
対象の期間	全期間をひとまとめにした	4月から6月の3か月が1行に丸められた
集計の単位	設備ごとではなく全体にした	8台分が平均値1つになった
返してほしい形	文章で説明することにした	表が欲しかったのに説明文が5行
空欄の扱い	埋めてよいと判断した	空欄だった稼働時間に、それらしい数字が入った

図面に寸法の記載がなければ、加工側が妥当な値を判断して作ります。指示が抜けた部分は相手が決める、という点は同じです。

プロンプトの4要素

役割・文脈・制約・出力形式の4つで依頼文を組み立てます

役割

答える立場を決めます。「製造現場の設備保全担当として」と書くだけで、用語の細かさと省略の度合いが揃います

文脈

何のためにやるのか、手元に何があるのかを書きます。「Excelから貼った設備8台分の稼働データがあります。月次の点検報告に使います」

制約

やってほしくないことを書きます。「推測で数字を作らない」「単位付きの文字列を残さない」「外部のファイルは読み込まない」

出力形式

返してほしい形を指定します。「設備ID / 稼働時間 / 不良数 の3列の表で」。形が決まると、前の結果と見比べられます

同じ依頼の書き分け

4要素を足すと、返ってくる形が毎回そろいます

直しにくい書き方

「設備のデータをまとめて」。役割も文脈も制約も出力形式も書かれていません。返ってくるのは、それらしい説明文が5行。数字は当てずっぽうで、どこを直せばよいかも分かりません。2回目を送っても、また違う形で返ってきます。

そのまま使える書き方

4要素をすべて入れると、返る形が固定されます。全文は次のページに載せます。ここで扱う稼働データは午後の第4章で実際に配ります。今は中身ではなく、書き方の形だけ見てください。

良い例の全文

行の左端が、その行の役割です。6行で1つの依頼文になります

前のページの右側の全文です。行頭の角かっこが、その行が4要素のどれにあたるかを示しています。読み上げずに、行と要素の対応だけ目で追ってください。

[役割] 製造現場の設備保全担当として回答してください。

[文脈] Excelから貼り付けた設備8台分の稼働データがあります。月次の点検報告に使います。

[制約] 稼働時間は数値に直してください。単位の h は列名に移してください。

[制約] 空欄は0で埋めず、空欄のまま残して、空欄の件数を最後に書いてください。

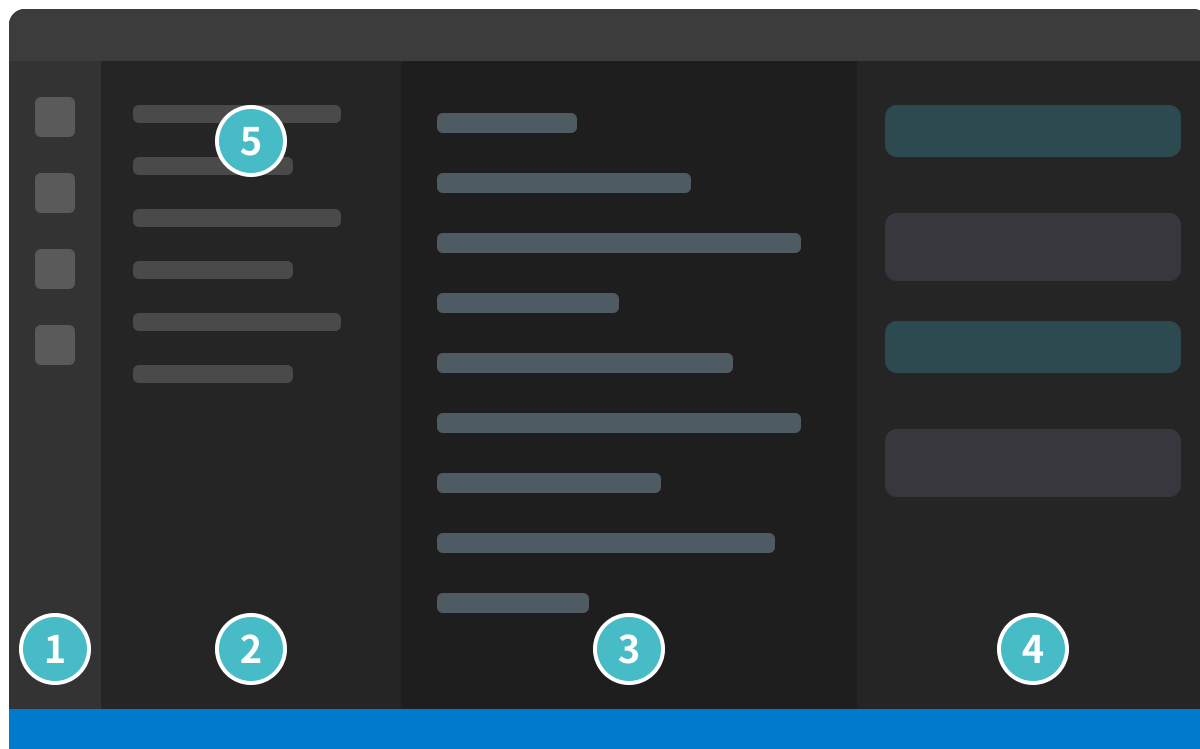
[制約] データに無い値を推測で作らないでください。

[出力形式] 結果は表で返してください。列は 設備ID / 稼働時間 / 不良数 の3列とします。

角かっこは説明のために付けています。実際に送るときは外して構いません。付けたままでも動きます。

依頼文を打つ場所

プロンプトを打つのは、**右側のチャットの下にある入力欄**です



- 1 アクティビティバー**
一番左の縦帯です。上から1番目のファイルのアイコンでエクスプローラーが開きます
- 2 エクスプローラー**
day2 フォルダの中身が並びます。prompt.txt はここで右クリックして新しいファイルから作ります
- 3 エディタ**
中央です。prompt.txt を開いて、依頼文を貼り付けて Ctrl+S で保存します
- 4 チャットパネル**
右側です。Ctrl+Alt+I で開きます。依頼文はこの一番下の入力欄に貼ります。送信は Enter、途中で改行するときは Shift+Enter です
- 5 ステータスバー**
一番下の帯です。Copilot のアイコンに斜線が出ていたらサインインが切れています

出力形式の3行

依頼文の後ろにこの3行を足すだけでも結果が変わります

4要素を全部書くのが面倒なときは、この3行だけ足してください。出力形式と制約の要点をまとめたものです。

- ・ 結果は表で返してください。列は 設備ID / 稼働時間 / 不良数 とします。
- ・ 数値は半角の数字だけにして、単位は列名に書いてください。
- ・ 元のデータに無い値は空欄のままにし、推測で埋めないでください。

3行目を落とすと、空欄にそれらしい数字が入ります。午後に配るExcelには欠損が2か所あります。

現場の言葉のまま貼らず、業界と工程の粒度に丸めてから送ります

客先名と製品名

「A社向けの車載コネクタ」ではなく「自動車部品の量産ライン」と書きます。工程と業界まで丸めても、返ってくる整形の手順は変わりません

型番と実測値

「EQ-1420 の 10.05mm」ではなく「設備A」「上限をわずかに超えた値」と書きます。桁と単位が合っていれば練習としては足ります

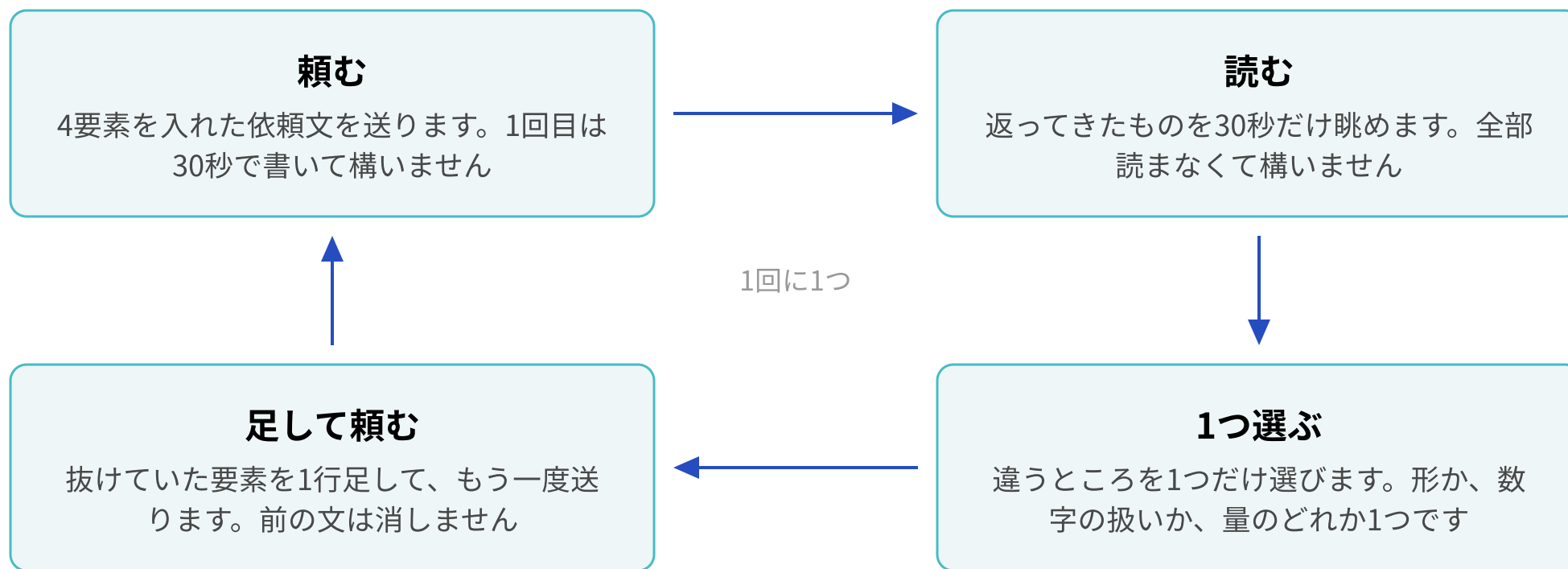
送信ボタンの前

貼ろうとしている文を、社外に出しても差し支えないかという目で1回読み返します。読み返しは30秒で終わります

常駐先の環境で使う場合は、常駐先のルールが優先します。判断に迷うものは貼らずに持ち帰ってください。

直しの1回転

違うところは**1回に1つだけ**伝えます。まとめて頼むと合っていた部分まで変わります



Copilotの提供元がGitHubです。サインイン先も出典も同じ場所です



GitHub

Copilotの提供元です。
サインインで使うアカウント
もこのものです

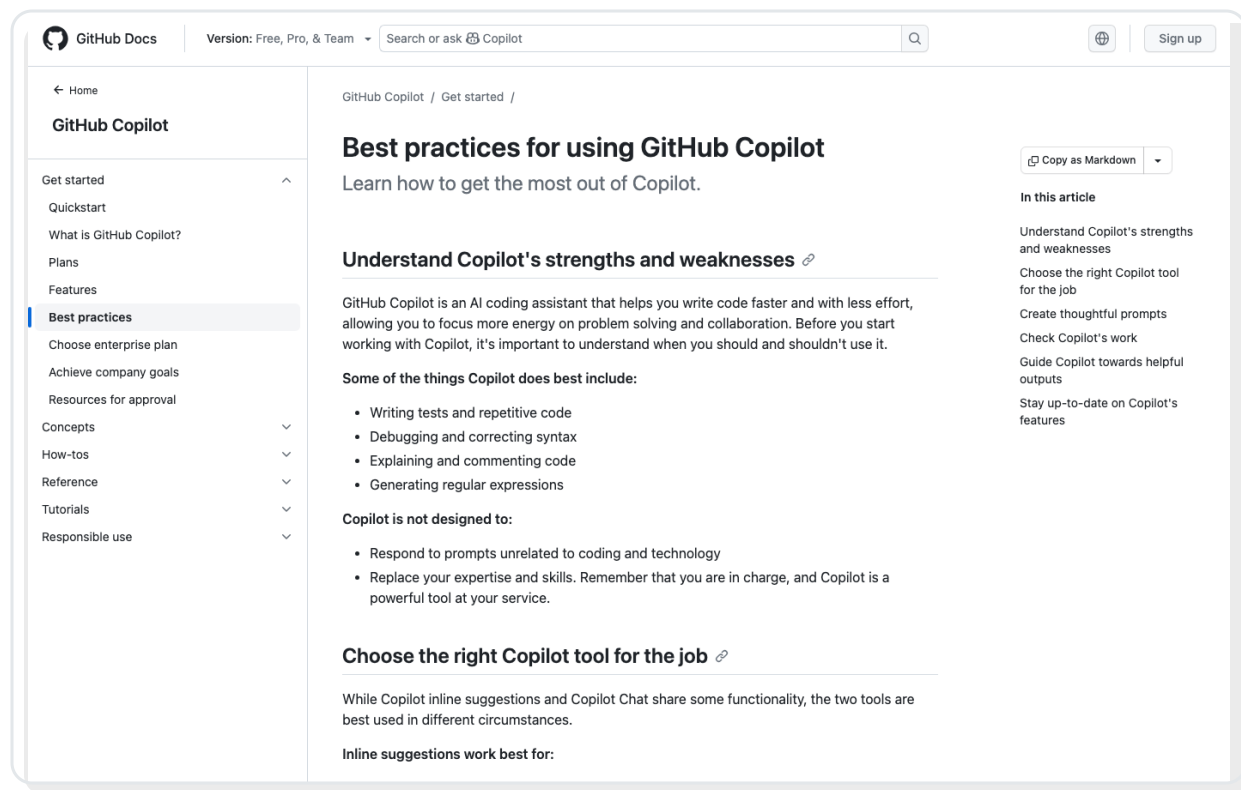


GitHub Copilot

VSCodeの中で動く機能です。
頼み方のコツも提供元が公開
しています

午前の環境確認で画面左下に出ていたユーザー名は、左のGitHubのアカウントです。

具体的に書く、出力を必ず確認する。提供元も同じことを書いています



画面: GitHub 公式ドキュメントの Best practices ページ

練習用の10行

次の演習で貼るのは、**この10行**です。架空の練習用データです

設備	稼働時間	停止回数	不良数
EQ-001 成形機A	192.0	3	12
EQ-002 成形機B	178.5	5	18
EQ-003 切削盤A	205.0	2	6
EQ-004 切削盤B	(空欄)	4	15
EQ-005 溶接ロボットA	210.0	1	4

仕込みは3か所です。EQ-004 の稼働時間が空欄、EQ-006 の稼働時間が単位付きの文字列、EQ-009 の停止回数が空欄。午後に配る 設備データ_2026Q1.xlsx とは別の、練習用の表です。

練習用の10行（続き）

設備	稼働時間	停止回数	不良数
EQ-006 溶接口ボットB	182.5 h	6	21
EQ-007 検査装置A	158.0	8	27
EQ-008 検査装置B	166.0	7	24
EQ-009 研削盤A	188.0	(空欄)	9
EQ-010 塗装ブースA	174.0	5	16

同じ話題を2通りで送り、返ってくるものの差を自分の目で確かめます

どんな場面か

午前は思いついた順に頼んで、返ってきたものを見ながら直しました。午後のデータ整形では同じ書き方だと、欠損に勝手な数字が入り、単位付きの文字列がそのまま残ります。ここで書き直しの型を1本作っておけば、午後はそれを貼るだけで始められます。

やること

- 1 前のページの10行の表をコピーしてチャットに貼り、続けて「稼働データを分析して」とだけ送ります。返ってきたものは30秒だけ眺めます（必須・5分）
- 2 新しいチャットを開いて同じ表をもう一度貼り、役割・文脈・制約・出力形式を1つずつ足して送り直します。制約には「元のデータに無い値は推測で埋めないでください」を必ず入れます（必須・15分）
- 3 1回目と2回目で変わった点を prompt.txt に3行で書き留めます。見るのは、返ってきた形・数字の扱い・説明の量の3点です（必須・10分）

演習 依頼文の書き直し（続き）

やること

- 1 余った方は、「推測で埋めないでください」の行だけを外して、稼働時間が空欄の EQ-004 を含めた表を送ります。空欄に数字が入ることを確かめます
- 2 余った方は、自分がよく使う3行を prompt.txt の先頭にまとめ、次から貼るだけで使える形にします

できあがるもの

day2 フォルダの prompt.txt。中身は、書き直した依頼文の全文と、気づいた差の3行です。Day3のデータ分析でも同じファイルを使います。

4つそろえば完了です。見た目のきれいさは基準に入れません

達成の目安

- 2回目の返答が、設備ID・稼働時間・不良数の列を持つ表の形になっている
- 稼働時間の列に「182.5 h」が残っておらず、182.5 という数値になっている
- EQ-004 の稼働時間が空欄のまま残っていて、勝手な数字が入っていない
- day2 フォルダの prompt.txt に、書き直した依頼文の全文が貼られて保存されている

つまずいたら

- 表ではなく説明文で返る: 出力形式の1行が依頼文の途中に埋もれています。末尾に移してから送り直します
- 2回目も1回目と似た答えが返る: 同じ会話の続きになっています。チャット上部の新しいチャットのアイコンで会話を作り直してから貼ります
- EQ-004 に数字が入ってしまう: 制約の行が他の文とつながっています。行頭に「・」を付けて独立した行にします
- 表を貼ると途中で送信される: 改行が Enter になっています。改行は Shift+Enter です
- prompt.txt が見つからない: day2 フォルダではなく別の場所に作られています。エクスプローラーの day2 の直下に作り直します

演習の達成基準（続き）

時間が余ったら

- 同じ依頼を「表で返してください」「CSV形式で返してください」「箇条書きで返してください」の3通りで送り、Excelに貼りやすいのはどれかを確認めます
- 役割だけを「新人の作業者に説明するつもりで」に差し替えて送り、言葉の細かさがどう変わるかを見ます
- 制約に「稼働時間が長い順に並べてください」を1行足して、並び替えまで頼めるかを試します

1回目と2回目の差

差を作ったのは、制約と出力形式の2つです

1回目に多かった返答

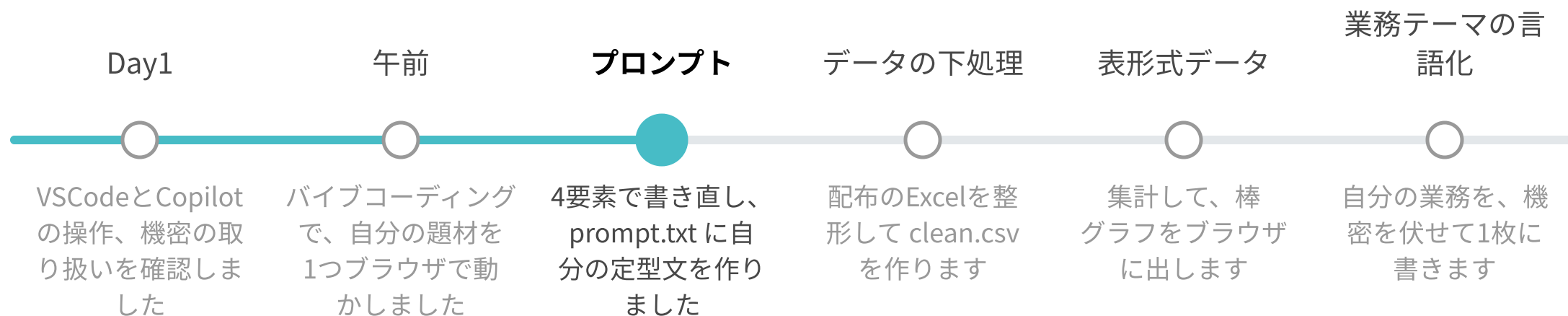
「稼働データを分析して」だけだと、10台をまとめた説明文が5行前後で返ります。設備ごとの数字は出ず、稼働時間が空欄だった EQ-004 には、平均に近い数字が入っていることがあります。停止回数の合計も、空欄を0として数えられています。

2回目に多かった返答

4要素を足すと3列の表で返り、182.5 h は 182.5 になり、EQ-004 の稼働時間は空欄のまま残ります。最後に「空欄1件」と件数が付くこともあります。この形なら、元の表と件数で突き合わせられます。

今日の現在地

prompt.txt が、このあと80分の入口になります



午後の後半へ

ここからは現場のExcelに近い汚れたデータを扱います

ここまでにしたこと

休憩の前に、役割・文脈・制約・出力形式の4つを足した依頼文を prompt.txt に保存しました。午前のバイブコーディングから数えて、ここまでは自分で選んだ題材と練習用の10行だけを扱っています。



これからやること

ここからは、その依頼文を練習用ではないデータに当てます。設備データ_2026Q1.xlsx の3つのシートを整えて、あらかじめ決めてある確認値と一致するところまで持っていきます。



そのあとどこで効くか

整えた表は、この後の章で集計と棒グラフに使います。Day3の分析演習も、同じように整った表から始まります。今日ここで作る clean.csv は来週そのまま開きます。

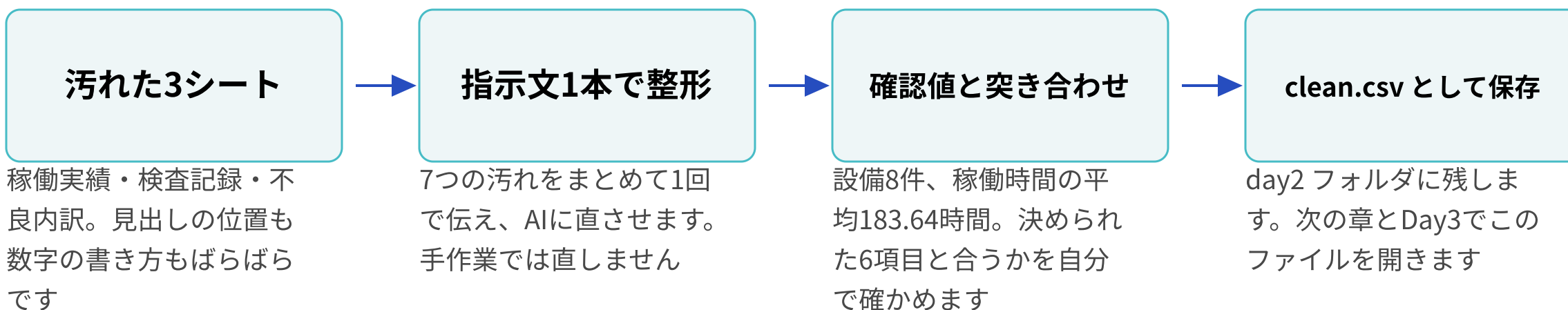
04

データの下処理

汚れたExcelをAIに渡して、集計できる形に整えられるようになります

この章の到達点

この章の終わりに、3シートが1枚の表になります



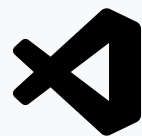
今日扱うファイル

設備データ_2026Q1.xlsx の3シートをまとめて扱います

シート	中身	目立つ汚れ
稼働実績	設備8台の稼働時間・停止回数・不良数	1行目にタイトル、見出しは3行目、途中に空白行。稼働時間の値がすべて単位付きの文字列、欠損が2か所
検査記録	ロット10件の寸法測定値と判定	測定値に全角数字「1 0.0 5」、日付が3通りの書き方、判定列に全角ハイフン
不良内訳	工程ごとの不良項目と件数	工程名が縦方向の結合セルで、2行目以降が空になっています

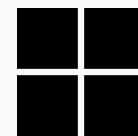
この3シートは1つのファイルに入っています。シートを切り替えるのはブラウザ側ではなく、AIに渡すときに人が選びます。

VSCodeで書いて、ブラウザで動かします。役割は分かれています



VSCode

依頼文を打ち、返ってきた
コードを保存する場所です。
ここでは動きません



Edge

保存したHTMLを開いて動かす
場所です。Excelを渡すのも
こちらです

Excelを読み込む画面はブラウザ側にあります。VSCodeの中でExcelを開こうとしないでください。

ブラウザは自分でファイルを開けません。人がドラッグして渡します

配布フォルダを開く

デスクトップに展開した機電系_Day2 フォルダの中に、01_データ読み込み.html と 設備データ_2026Q1.xlsx、それに lib フォルダが入っています

HTMLをブラウザで開く

01_データ読み込み.html をダブルクリックします。VSCodeではなくブラウザで開きます。ファイルを落とす点線の枠が出ます

Excelをドラッグして落とす

設備データ_2026Q1.xlsx を点線の枠にドラッグします。稼働実績・検査記録・不良内訳の3つのシート名が並びます

中身をコピーして貼る

見たいシート名を押し、「この表をコピー」でタブ区切りのままコピーして、VSCodeのCopilot Chatに貼ります。この時点ではまだ何も保存しません

見出しの位置と空白行

見出しは3行目にあり、途中に空白行が入っています

行	Excelの画面での見え方	AIに渡ったときの姿
1行目	2026年度 設備稼働実績（第1四半期）	文字が1つだけ入った行として、そのまま入ります
2行目	（空）	空の行として入ります
3行目	設備ID／設備名／稼働時間／停止回数／不良数	見出しですが、4行目以降の中身と区別する材料がありません
4行目から	設備ごとのデータが並びます	普通に読めます
途中の1行	（空）	ここで表が終わったと判断されることがあります

縦方向の結合セル

結合セルは、範囲の先頭の1行にしか値が残りません

行	Excelの画面で見える工程名	読み込んだあとの中身	そのまま数えると
4行目	成形（3行分の結合セル）	成形	1件だけ数えられます
5行目	（同じセルの続き）	空欄	工程名なしとして落ちます
6行目	（同じセルの続き）	空欄	工程名なしとして落ちます
7行目	切削（2行分の結合セル）	切削	1件だけ数えられます
8行目	（同じセルの続き）	空欄	工程名なしとして落ちます

空欄を直前の工程名で埋めてから数えると、成形33件・切削21件・検査7件、合計61件になります。埋めないまま数えると、成形は先頭の1行分しか残りません。

数値のふりをした文字列

単位付き・全角数字・日付は、まとめて1回で直させます

稼働時間の「182.5 h」、測定値の「1 0 . 0 5」、3通りに割れている日付。この3つは見た目が数字でも、そのままでは計算も並べ替えもできません。直す依頼はまとめて1回で渡します。

次の3点を直してください。

- 1) 「182.5 h」のように単位が付いた文字列は、単位を外して半角の数値にします
- 2) 「1 0 . 0 5」のような全角の数字は半角に直します
- 3) 日付は 2026-04-01 の形に揃えます。2026/04/01 と 2026/4/2 と 2026-04-03 が混ざっています
直したあと、数値に変換できなかったセルが何件あったかを最後にご記入ください。

最後の1行が事故を防ぎます。書かないと、変換できなかったセルが黙って0に置き換わり、平均が下がります。

欠損の扱い

空欄を0で埋めると、平均が**約23時間**下がります

空欄を0で埋めた場合

稼働時間の欠損1件を0として8台で割ります。平均は約160.7時間になります。この設備は止まっていた、という読み方が生まれてしまいます

空欄を除いた場合

欠損1件を除いて7台で割ります。平均は183.64時間です。除いた件数を必ず併記して、7台の平均であることを明示します

整形の依頼文

この7項目をそのままコピーして、演習で使います

ここまでの5ページで見た汚れを、7つの箇条書きにまとめたものです。自分で書き起こさず、この文をコピーしてから表を貼ってください。

設備保全の担当者として、下に貼る表を整えてください。

- 1) 見出しは3行目です。1行目と2行目、途中の空白行は無視してください
- 2) 工程の列が空の行は、直前の行の工程名で埋めてください
- 3) 「182.5 h」のような単位付きは単位を外して半角の数値にしてください
- 4) 「1 0 . 0 5」のような全角数字は半角に直してください
- 5) 日付は 2026-04-01 の形に揃えてください
- 6) 空欄は0で埋めず、空欄のまま残して件数を数えてください
- 7) 数値に変換できなかったセルの件数を最後に書いてください

整形できたかの確認

この6つの数字と一致すれば、下処理は成功です

確認項目	期待する値
設備の件数	8件
稼働時間の欠損	1件
停止回数の欠損	1件
稼働時間の平均	183.64 時間
稼働時間の中央値	182.5 時間
数値に変換できなかったセル	0件

毎月手で直している作業を、指示文1本に置き換えます

どんな場面か

毎月届く設備のExcelを、集計の前に手で直している方が多いはずです。見出しをずらし、単位を消し、全角を直す。この手直しを、今日の依頼文1本で置き換えられるかどうかを試します。

やること

- 1 配布フォルダの 01_データ読み込み.html をダブルクリックしてブラウザで開き、設備データ_2026Q1.xlsx を点線の枠にドラッグして落とします。稼働実績シートを押し、「この表をコピー」でコピーします [5分]
- 2 Copilot Chatを開き、整形の依頼文のページから7項目をコピーして先に貼ります。そのすぐ下に、コピーした稼働実績シートの中身を貼って送ります [10分]
- 3 返ってきた表を、確認の6項目と1つずつ突き合わせます。設備8件／欠損1件と1件／平均183.64／中央値182.5／変換できなかったセル0件 [8分]

演習 下処理をやりきる（続き）

やること

- ① 合わなかった項目だけを、もう一度チャットで伝えて直します。全部やり直させると別の場所がずれます [4分]
- ② 整形後の表をコピーし、VSCodeの左側で右クリック→新しいファイル→ clean.csv と入力→貼り付け→Ctrl+S で保存します [5分]

できあがるもの

day2 フォルダの clean.csv です。VSCodeで開いたときに、設備8台分の行と見出し1行が入っていれば完成です。

下処理の達成基準

この3つがそろえば、次の章に進んで大丈夫です

達成の目安

- day2 フォルダに clean.csv があり、VSCodeで開くと設備8台分の行が並んでいます
- 確認の6項目のうち5つ以上が一致しています。設備件数8件と平均183.64時間の2つは必ず合わせます
- 稼働時間の列に「h」の文字と全角の数字が1つも残っていません

つまづいたら

- 設備件数が9件や10件になる。タイトル行か空白行を数えています。依頼文の1番だけをもう一度送ります
- 平均が160時間台になる。欠損を0で埋めています。依頼文の6番だけをもう一度送ります
- 変換できなかったセルが0件にならない。単位付きか全角が残っています。「どのセルが残っているか一覧で出してください」と続けて頼みます
- 返答が途中で切れる。3シート分をまとめて貼っています。稼働実績シートだけを貼り直します
- clean.csv が保存できない。ファイル名が clean.csv.txt になっていないか、エクスプローラーの表示→ファイル名拡張子で確認します

下処理の達成基準（続き）

時間が余ったら

- 稼働時間と停止回数の関係を1行で書いてみてください。次の章の統計の読み方でそのまま使います
- 同じ依頼文を検査記録シートに当てて、7項目のどこを書き換えれば通るかを確かめてください

Q. 縦方向の結合セルでまとめられた工程名を、そのまま読み込んで集計するとどうなりますか

- A Excelの見た目と同じように、すべての行に工程名が入ります
- B 結合セルは読み込めないで、その行ごと消えます
- C 先頭の行だけが工程名として数えられ、残りは工程名なしになります
- D 工程名の空欄が数値の0として扱われます

正解は次のページで確認します。まず自分で考えてみてください。

正解はC。正解はC。空欄を埋めてから数えると成形は33件になります

先頭の行だけが工程名として数えられ、残りは工程名なしになります

C

結合セルは表示上まとまって見えているだけで、値が入っているのは範囲の先頭のセルだけです。読み込むと2行目以降は空になり、工程別に数えると成形は先頭の1行分しか残りません。空欄を直前の工程名で埋めてから集計すると、成形33件・切削21件・検査7件、合計61件になります。

A

すべての行に工程名が入ります

画面での見え方であって、セルの中身ではありません。クリックして数式バーを見ると空です

B

その行ごと消えます

行は残ります。空になるのは工程名の列だけで、不良項目と件数はそのまま入ってきます

D

空欄が数値の0として扱われます

空欄は0ではなく、空のまま入ってきます。0にするのは集計する側の判断で、元データにはありません

33件と合計61件は、次の章のグラフ演習でそのまま検算に使います。

整えた表の使い道

整えた表を、読める数字と1枚の図に変えます

ここまでにしたこと

汚れた3シートのうち、稼働実績シートを整えて clean.csv に保存しました。設備8件、平均183.64時間まで確認しました。



これからやること

ここからは残りの2シートを使います。検査記録から規格外のロットを抜き出し、不良内訳を工程別に集計して棒グラフまで持っていくます。全部ブラウザの中で動きます。



そのあとどこで効くか

グラフと検算のやり方は、Day3の分析レポートでそのまま使います。今日は「AIが出した数字を人がどう照らすか」の型を持ち帰ってもらいます。

05

表形式データの取り扱い

整えたデータを集計して、ブラウザでグラフまで出せるようになります

稼働時間の統計値

平均だけを見ると、**ばらつき**を見落とします

項目	値	読み方
平均	183.64 時間	欠損1件を除いた7台の平均です
中央値	182.5 時間	平均とほぼ同じです。極端に外れた設備はないと読めます
標準偏差	18.51 または 17.13 時間	計算の種類で変わります。標本なら18.51、母集団なら17.13。どちらもおよそ±18時間の幅という読み方です
最小	158.0 時間	他の設備より25時間ほど短い1台があります
最大	210.0 時間	最小との差は52時間です

この稼働時間は稼働実績シートの設備別の数字です。この後に出てくる工程別の不良件数とは、集計の単位が別です。

10件のロットから、規格を外れたものを条件で抜き出します

どんな場面か

寸法の測定値を目で追って規格外を探すと、桁を見誤ります。しかも測定値には全角の数字が混ざっています。人が見落とすところを、条件を書いて機械に探させます。

やること

- 1 01_データ読み込み.html に 設備データ_2026Q1.xlsx をドラッグし、検査記録シートを押して「この表をコピー」でコピーし、Copilot Chatに貼ります [2分]
- 2 「測定値の全角数字を半角に直し、日付を 2026-04-01 の形に揃えてください」と頼みます [2分]
- 3 続けて「寸法測定値の規格は下限9.9、上限10.1です。規格から外れたロットIDを一覧にしてください」と頼みます [2分]
- 4 出てきたロットIDを手元にメモします。元データの判定列がOK何件・NG何件かも数えておきます [2分]

できあがるもの

成果物のファイルはありません。規格から外れたロットIDが3件、画面に一覧で出ていれば達成です。

規格外抽出の達成基準

3件出て、合計が10ロットになれば正解です

達成の目安

- 規格から外れたロットIDが3件、一覧で出ています。L-1003・L-1006・L-1009 です
- 元データの判定列は、OKが6件、NGが3件です。AIが出した3件と、判定列のNG3件が同じロットIDになっています
- OK6件+NG3件+どちらにも数えない1件で、合計10ロットになります

つまずいたら

- 4件出て、うち1件は測定値が空欄。空欄のロットを規格外に数えています。測定値が無いものは判定できない、と伝え直します
- 1件も出ない、または件数が毎回変わる。全角の数字が残ったまま比べています。半角に直す手順を先にやり直します
- 件数が3件より多い。上限を「10.1以上」と書くと、10.1ちょうどの値まで規格外に数えます。上限10.1を超えたもの、に書き換えます
- 判定列と一致しない。判定列に全角ハイフンが入った行があります。その行を空欄と同じ扱いにします

規格外抽出の達成基準（続き）

時間が余ったら

- 下限9.9を下回ったロットが無いことも確かめてください。上限だけ見て終わる癖がつきやすい場所です
- 規格を9.95～10.05に狭めると何件になるかを試して、しきい値で結論が変わることを確かめてください

講師の画面で通します。手を動かすのはグラフ演習からです

1

規格を数値で伝える

「寸法測定値の規格は下限9.9、上限10.1です」と数値で渡します。規格という言葉だけでは判定できません

2

全角を直してから判定する

「1 0.0 5」を半角にしてから比べます。文字列のまま比べると、桁の大小が入れ替わります

3

外れたロットを一覧にする

L-1003、L-1006、L-1009 の3件が上限10.1を超えています

4

元の判定列と突き合わせる

元データの判定はOKが6件、NGが3件です。AIが出した3件と同じロットIDかを確認めます

5

合わないときは元を見る

測定値が空欄で、判定列が全角ハイフンの行が1件あります。この1件はどちらにも数えません

OK6件・NG3件・欠損1件で合計10ロットです。合計が10にならないときは、必ずどこかに数え落としがあります。

相関の読み方

相関 -0.965 は、**原因の証明ではありません**

-0.965 が示すこと

稼働時間が長い設備ほど不良数が少ない、という向きの関係が数字に出ています。
1に近いほど直線に並びます

示していないこと

長く動かせば不良が減る、とは言えません。よく動く設備がたまたま新しいだけ
かもしれません

母数を先に見る

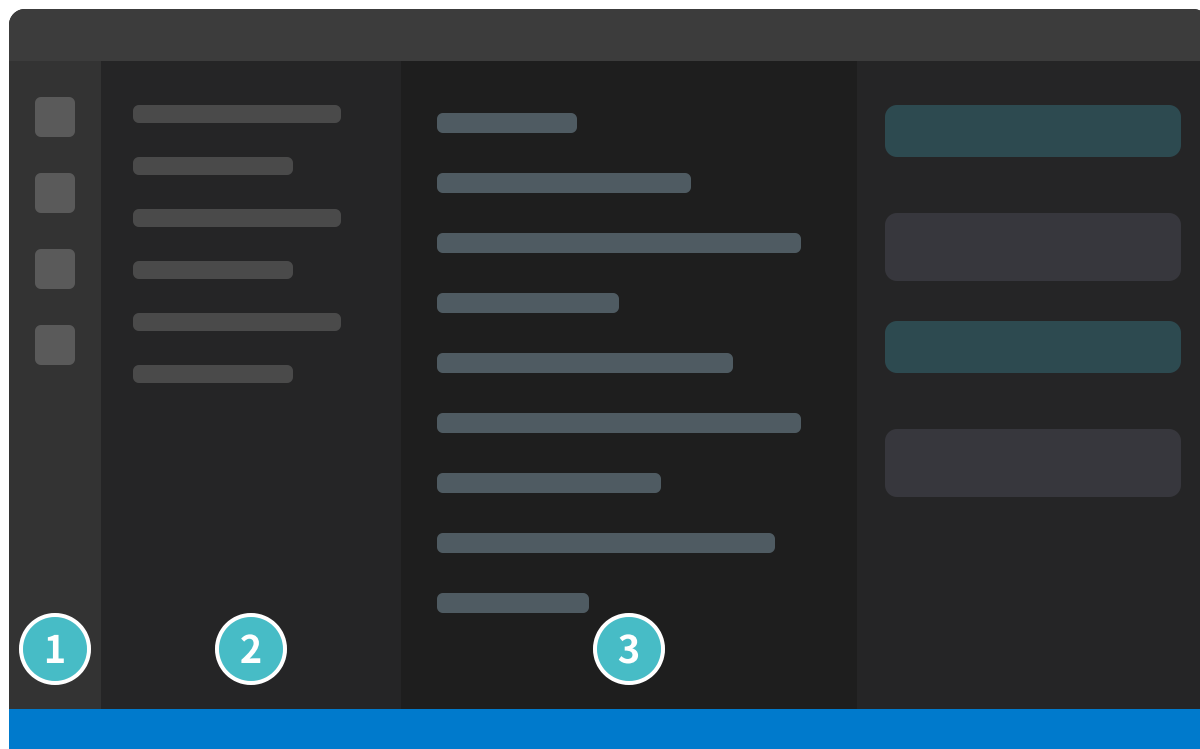
稼働時間が空欄の1台を除いた7台から出したものです。1台外すだけで値が動きます。
何台から出した相関かを必ず併記します

次に何を調べるか

設備の導入年や保全の頻度など、別の列を足して確かめます。相関は調べる先を決める材料として使います

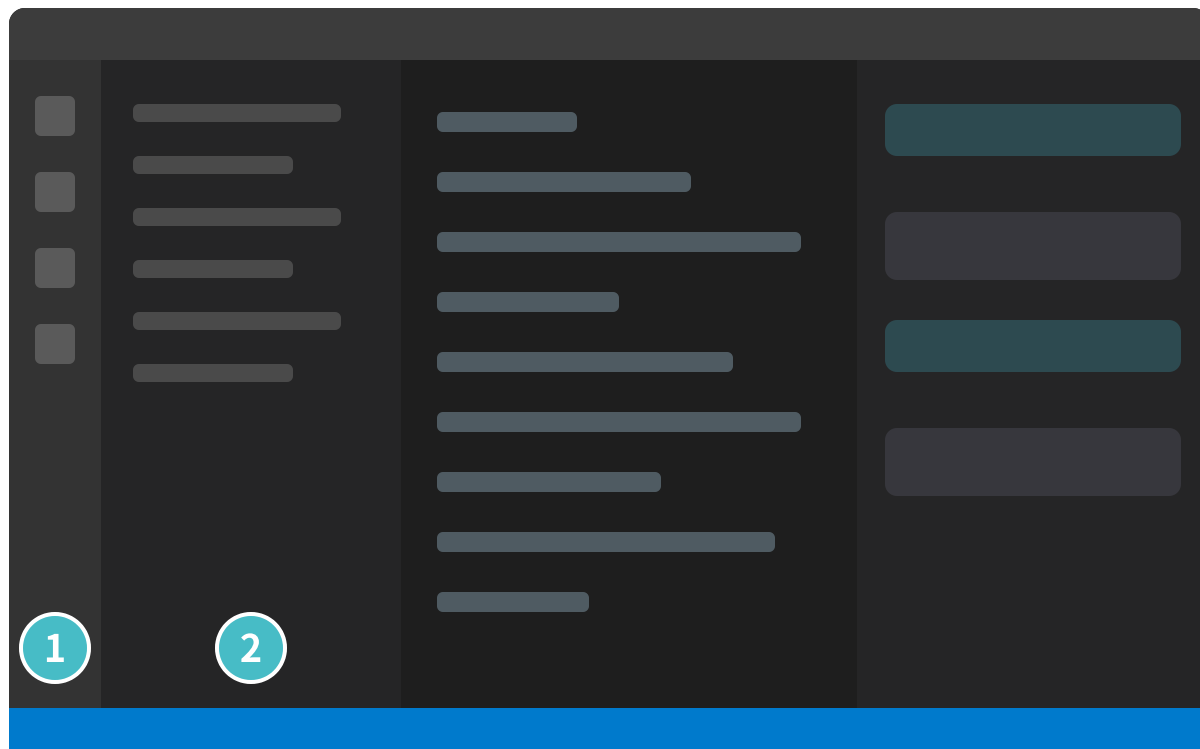
グラフ演習の準備

グラフの部品は **day2** フォルダにコピーしてから使います



- 1 アクティビティバー**
一番左の縦の並びです。一番上のファイルのアイコンを押すと、エクスプローラーが開きます
- 2 エクスプローラー**
day2 フォルダの中身が出ます。ここに `chart.umd.js` が並んで見えていれば、コピーは成功です
- 3 エディタ**
`chart.html` を開く場所です。タブの名前の横に丸印が出ていたら未保存です。Ctrl+S を押すと丸印が消えます

グラフ演習の準備（続き）



1 チャットパネル

右側です。ここに依頼文を貼ります。返ってきたコードは、コードの枠の右上にあるコピーボタンで写します

2 ステータスバー

一番下の帯です。文字コードが UTF-8 になっているかを見ます。ブラウザで日本語が化けるときはここを直します

グラフの依頼文

読み込み先を指定しないと、グラフは真っ白になります

集計結果を貼った後ろに、この5行を続けて送ります。3行目を書かないと、AIはインターネット上のURLから部品を読もうとして、社内の環境では表示されません。

この集計結果を棒グラフで表示するHTMLを1ファイルで作ってください。
同じフォルダにある `chart.umd.js` を `<script src="chart.umd.js"></script>` で読み込んでください。
インターネット上のURLからは読み込まないでください。
横軸は工程名、縦軸は不良件数です。棒の上に件数の数字を出してください。
HTMLの中身だけを返してください。説明の文章は不要です。

この依頼文も `prompt.txt` に足しておいてください。Day3で別のグラフを作るときに、横軸と縦軸の行だけ書き換えて使えます。



株式会社ギブリー (Givery, Inc.)

〒150-0036 東京都渋谷区南平台町15-13 帝都渋谷ビル8F

集計した数字を、会議で見せられる形まで一気に通します

どんな場面か

工程別の不良件数を口頭で報告しても、どの工程が多いのかは伝わりません。会議で1枚見せれば済む形まで、今日のうちに持っていくます。作るのはブラウザで開けるHTML1枚です。

やること

- 1 配布フォルダの lib フォルダを開き、chart.umd.js を day2 フォルダにコピーします。右クリック→コピー、day2 で右クリック→貼り付けです [3分]
- 2 01_データ読み込み.html で不良内訳シートを表示し、「この表をコピー」でコピーしてから、「工程の列が空の行は、直前の行の工程名で埋めてください」と頼みます [5分]
- 3 続けて「工程ごとの不良件数を数えてください」と頼みます。成形33件・切削21件・検査7件になれば正解です [5分]

手順2を飛ばして手順3だけを頼むと、件数が33・21・7になりません。結合セルの穴埋めが先です。

演習 集計してグラフにする（続き）

やること

- 1 前のページの依頼文5行を貼ってHTMLを作らせます。返ってきたコードをコピーし、day2フォルダに chart.html という名前で保存します [6分]
- 2 chart.html をダブルクリックしてブラウザで開き、棒の高さと棒の上の数字が集計結果と合っているかを見ます [5分]

できあがるもの

day2 フォルダの chart.html です。ダブルクリックすると、ブラウザに棒が3本出ます。

グラフ演習の達成基準

棒が3本出て、数字が33・21・7なら完成です

達成の目安

- day2 フォルダに chart.html があり、ダブルクリックするとブラウザに棒が3本出ます
- 棒の高さの順が 成形>切削>検査 になっています
- 棒上の数字が 33・21・7 で、足すと61件になります

つまずいたら

- ブラウザが真っ白。day2 フォルダに chart.umd.js が無いか、HTMLの中で http から始まるURLを読もうとしています
- 棒が1本しか出ない。工程の空欄が埋まっておらず、成形しか集計されていません。穴埋めの依頼からやり直します
- 件数が61件にならない。空欄のまま数えた行があります。埋めた後の表をもう一度見せて数え直させます
- ダブルクリックしてもVSCodeが開く。右クリックからプログラムを選んでブラウザで開きます
- 文字が化ける。ステータスバーの文字コードを UTF-8 にして保存し直します

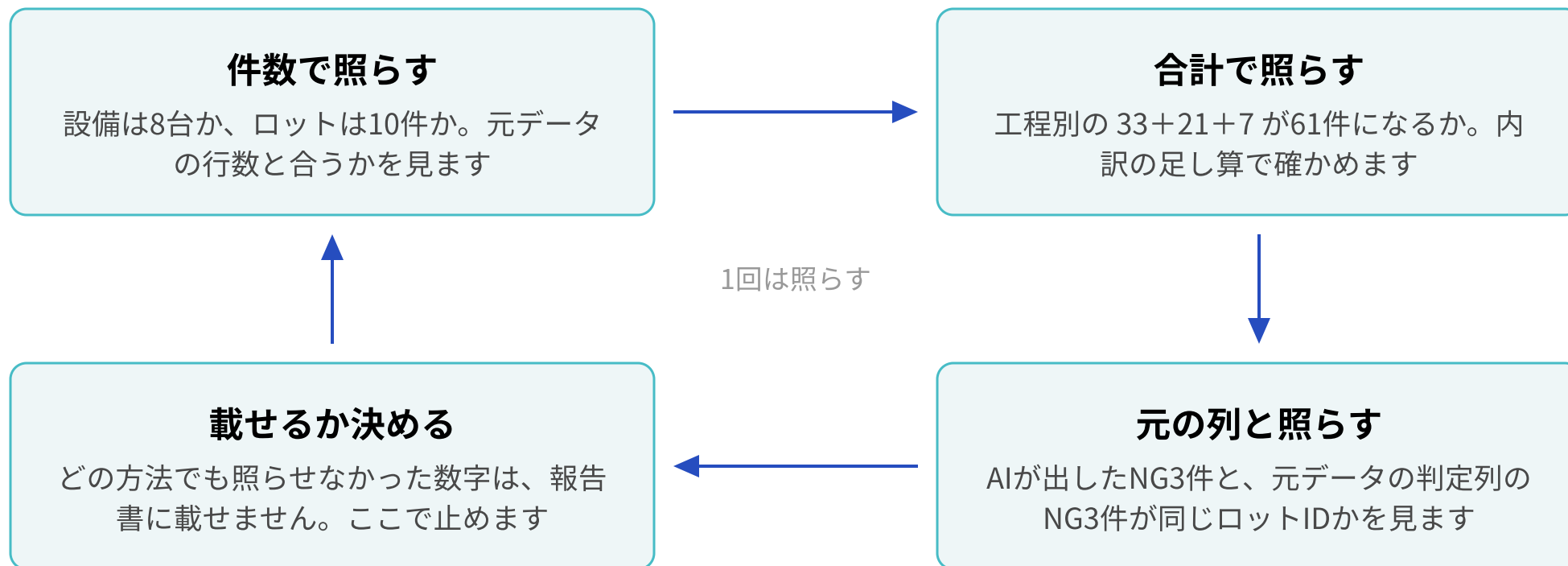
グラフ演習の達成基準（続き）

時間が余ったら

- 稼働時間と不良数を散布図にして、相関 -0.965 がどう見えるかを確認してください。稼働時間が空欄の1台は描けないので、点は7つしかありません
- 設備別の比較と工程別の内訳のどちらが自分の会議で使えるかを、chart.html の下に1行コメントで書き足してください

検算の型

AIが出した数字は、**必ず1回**どこかで照らします



今日この3つの照らし方を全部使いました。Day3のレポート作成でも同じ順番で使います。

配布データから自分の業務へ

ここからは配布データを離れて、**自分の現場の話**を書きます

ここまでにしたこと

ここまでは全員が同じ架空のExcelを扱いました。整形も抽出もグラフも、答えが決まっている題材でした。



これからやること

残りの30分は、自分の職場で手間になっている作業を1つ書き出します。提出も発表もありません。書いたものは自分の手元に残します。



そのあとどこで効くか

書いたテーマは、Day3の最後にある業務改善ミニアプリの題材になります。何も書かずに来ると、Day3の午後で題材探しから始めることになります。

06

業務テーマの言語化

自分の困りごとを、社外に出しても問題ない書き方で整理できるようになります

書くときの4つのルール

提出も発表ありません。手元の整理として書きます

ルール	具体的にどうするか
客先名・製品名・プロジェクト名・取引先名は業界・工程レベルに丸める	特定の会社名や装置の型番ではなく、自動車業界の設計工程、といった書き方にします
実データ・実ファイル・スクリーンショットは使用しない	手元のExcelや図面の画面は貼りません。今日の演習データのような架空の値に置き換えます
数値や仕様は具体値を避け、目的・課題・期待効果の粒度で記載する	公差の実数値ではなく、寸法の判定に時間がかかる、という書き方にします
提出前に受講者本人が社外に出ても問題ない内容かを確認する	書き終えたら自分でもう一度読み返します。判断がつかない部分は書かずに残します

この4つはテクノプロ様のルールをそのまま載せています。左の列は言い換えずに使ってください。

書き方の見本

この5行を埋めれば、それがそのままテーマになります

書けない例は「B社向けコントローラの試験成績書を、社内システムから出したCSVを手で貼り直している。1件40分」。客先名も製品名も社内システムの経路も入っています。同じ困りごとを、次の5行の形に置き換えます。

【業界と工程】 電機業界の検査工程
【作業の名前】 試験結果の一覧から報告書を作る
【今のやり方】 一覧を見ながら、報告書の様式に手で書き写している
【どのくらい手間か】 1件あたり数十分かかり、月に何十件もある
【AIに任せたい部分】 転記と体裁づくり

困りごとと期待は残り、客先名・製品名・具体的な数値は消えています。手間の書き方は「数十分」「何十件」までにとどめます。

12分で5行を埋めます。回収はしません

どんな場面か

Day3の午後に、業務改善ミニアプリを作る時間が130分あります。そこで題材から考え始めると作る時間が残りません。今日のうちに、自分の職場で手間になっている作業を1つ決めておきます。

やること

- 1 VSCodeの左側で右クリック→新しいファイル→theme.txt と入力して作ります [1分]
- 2 前のページの5つの見出しをそのまま写し、右側を自分の言葉で埋めます。手が止まったら、今週いちばん面倒だった作業から書きます [7分]
- 3 AIに任せたい部分を1つに絞ります。作業全体ではなく、転記だけ、集計だけ、のように部分を選びます [2分]
- 4 書いたものを自分で読み返し、客先名・製品名・具体的な数値が残っていないか探して消します [2分]

できあがるもの

day2 フォルダの theme.txt です。5行が埋まっていれば完成です。回収も発表もしません。

テーマ書き出しの達成基準

5つが1文ずつ入っていれば完成です

達成の目安

- 業界と工程／作業の名前／今のやり方／どのくらい手間か／AIに任せたい部分、の5つが1文ずつ入っています
- 手間の書き方が「数十分」「月に何十件」のような粒度になっていて、実数値が入っていません
- AIに任せたい部分が1つに絞られています。作業全体ではなく、その一部を指しています
- day2 フォルダに theme.txt として保存されています

つまずいたら

- テーマが大きすぎて書けない。部署の業務全体ではなく、自分が今週やった作業1つに絞ります
- 書ける作業が思いつかない。今日作った集計やグラフを、自分の業務の何に置き換えられるかから考えます
- 固有名詞を外すと何の話か分からなくなる。業界と工程の2行を先に埋めると、残りが書きやすくなります
- 書いてよいか判断がつかない。判断がつかないものは書きません。空欄のままで構いません

テーマ書き出しの達成基準（続き）

時間が余ったら

- その作業を、今の手順のまま3ステップに分けて書いてみてください。Day3でどこをAIに渡すか決めるときに使います
- 同じ作業を別の工程の人がやっている場合、どこが違うかを1行足してください

今日つくったもの

day2 フォルダにこの5つがそろっているか確認します

index.html

午前のバイブコーディング演習で作った、ブラウザで動くページです

prompt.txt

午後前半のプロンプト演習で作った定型文です。整形とグラフの依頼文も足しています

clean.csv

下処理演習で整えた稼働実績の表です。Day3の分析演習でこれを開きます

chart.html

グラフ演習で作った棒グラフのページです。同じフォルダの chart.umd.js とセットで動きます

theme.txt

自分の業務テーマです。Day3の業務改善ミニアプリの題材になります

Day3ですること

同じ環境、同じフォルダで来週の続きをやります

Day1



生成AIの基礎とリテラシー、VSCodeとGitHub Copilotの基本操作をやりました

Day2



日本語で頼んで動くページを作り、汚れたExcelを整えてグラフまで出しました。
自分の業務テーマも書きました

Day3



共通データの分析とレポート化、そして見本から選ぶ業務改善ミニアプリの作成をします。theme.txt がその題材になります