

TRAINING

# AI協働開発ハンズオン講座

## 機電系トラック Day3

---

データ分析の実践とレポート作成



## 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。

# 01

## Day2復習と環境確認

---

分析に入る前に、全員のVSCodeとCopilotとデータを同じ状態にそろえます [30min]

本日は午前で**分析の型**を作り、午後は自分の題材に使います

Day1 Copilotの操作

チャットで質問し、  
ファイルを添えて渡す動き  
を覚えました

Day2 下処理と  
バイブコーディング

空白行と結合セルと単位付  
きの文字列をそろえ、日本  
語の指示から動く画面を出  
しました

Day3午前 分析と報告

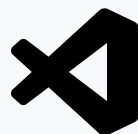
整形した数字を集計し、日  
本語の所見を書いて報告書  
にします

Day3午後 業務改善  
ミニアプリ

見本から1つ選び、時間内  
にブラウザで動くところま  
で作ります

午前の中心はデータ分析演習 [90min] です。演習のあとに休憩 [10min] を取ります

書くのはVSCode、頼むのはGitHub Copilot、**確かめるのはEdge**です



### VSCode

ファイルを置き、コードを書いて保存する場所です



### GitHub Copilot

VSCodeの中のチャット欄です。日本語の指示を書いて渡します



### Edge

作ったHTMLを開いて、動きを確かめるブラウザです

## 分析に入る前に、4点だけそろえます

### どんな場面か

Day2から1週間空いています。作業フォルダの場所が人によってばらばらのまま演習に入ると、下処理の途中で「ファイルが見つからない」が発生し、90分の演習時間がそこで削られます。置き場所を1つに固定しておきます。

### やること

- 1 デスクトップに day3 という名前のフォルダを作ります。
- 2 配布ZIPを右クリックしてすべて展開し、設備データ\_2026Q1.xlsx / xlsx.full.min.js / papaparse.min.js / chart.umd.js の4つを day3 フォルダに入れます。
- 3 VSCodeのファイルメニューからフォルダーを開くを選び、day3 フォルダを開きます。左のエクスプローラーに4つのファイルが並びます。
- 4 Copilot Chatの入力欄に「このフォルダにあるファイルを一覧で教えてください」と打ち、4つのファイル名が返ってくることを確認します。

### できあがるもの

成果物はありません。左のエクスプローラーの見出しが DAY3 になっていて、その下に4つのファイルが見えていれば達成です。本日作るファイルはすべてこの day3 フォルダに置きます。

# 環境確認の達成基準

## 達成の目安

- 左のエクスプローラーの見出しが DAY3 になっている
- その下に4つのファイルが並び、~\$ で始まる一時ファイルが増えていない
- Copilot Chatの返答に4つのファイル名がそろっている

## つまずいたら

- 左にファイル一覧が出ない。単一ファイルだけを開いています。ファイルメニューのフォルダーを開くから day3 を開き直してください
- Copilotが返事をしない。画面右下にサインインを促す表示が出ていれば、そこから入り直します
- xlsxが開けないと表示される。Excelで開いたままです。Excelを閉じてから読ませます
- ファイルが3つしか見えない。ZIPの中をダブルクリックで覗いているだけです。右クリックのすべて展開を使います

## 時間が余ったら

- Copilot Chatに「設備データ\_2026Q1.xlsx のシート名を全部教えて」と聞き、稼働実績・検査記録・不良内訳の3つが返るか見てください
- xlsx.full.min.js と chart.umd.js が何をするファイルかをCopilotに1行で説明させてみてください

Q. 「182.5 h」のように単位が付いた状態の稼働時間を、そのまま合計しようとするとうなりますか。

- A 単位が自動で外れて、正しく合計される
- B 小数点以下が切り捨てられて合計される
- C 文字として扱われ、計算できないか誤った結果になる
- D 数値に変換されるが、単位だけが末尾に残る

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C。正解は [C]。単位付きの文字列は、数値に直してから計算します

文字として扱われ、計算できないか誤った結果になる

C

「182.5 h」は数値ではなく文字列として保存されています。文字列は足し算の対象にならないため、計算できないか、その行が無視されて誤った合計になります。単位を外して数値だけの列にする下処理が先に要ります。

A

単位が自動で外れて、正しく合計される

単位を自動で外す仕組みはありません。表計算ソフトでも文字列のまま扱われます

B

小数点以下が切り捨てられて合計される

切り捨ては起きません。そもそも数値として読まれていないので、四捨五入の話に入る前で止まっています

D

数値に変換されるが、単位だけが末尾に残る

変換は自動では起こらず、単位だけが残ることもありません

セルの中身が左に寄っていれば文字列、右に寄っていれば数値です。表計算ソフトでの見分け方として現場でそのまま使えます

## ここから分析演習

準備は終わりです。ここから90分は手を動かします

### ここまでにしたこと

ここまで：day3 フォルダに4つのファイルをそろえ、Copilotがそのフォルダを読める状態にしました。単位付きの文字列がそのままでは計算できないことも確認しました。



### これからやること

これから：設備データ\_2026 Q1.xlsx を整形し、集計して、所見3行を書くところまでを90分で進めます。指示文はすべて画面に出すので、写して使ってかまいません。



### そのあとどこで効くか

そのあと：この90分で作る4つのファイルが、休憩明けの報告書とPDFの材料になります。午後の最終課題でも、この進め方をそのまま自分の題材に当てはめます。

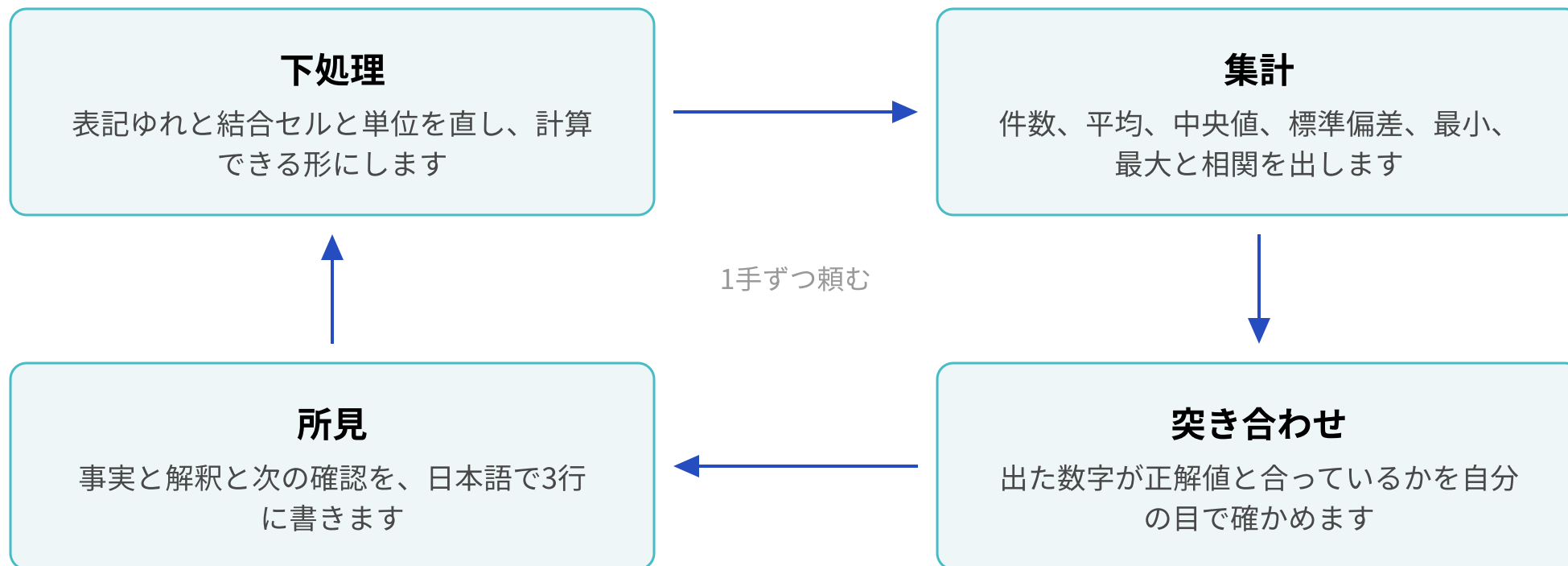
# 02

## データ分析演習

---

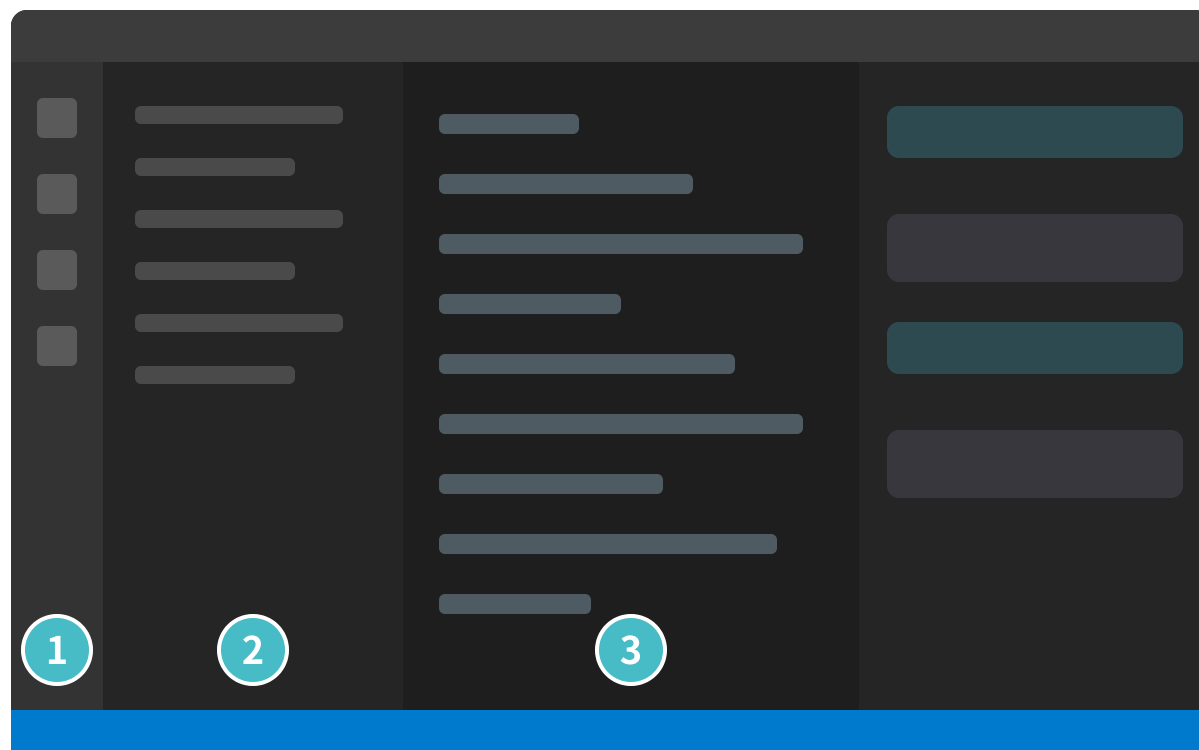
汚れた表を整形して、集計して、所見3行まで書きます [90min]

4つを一度に頼まず、**1手ずつ**出させて毎回目で見ます



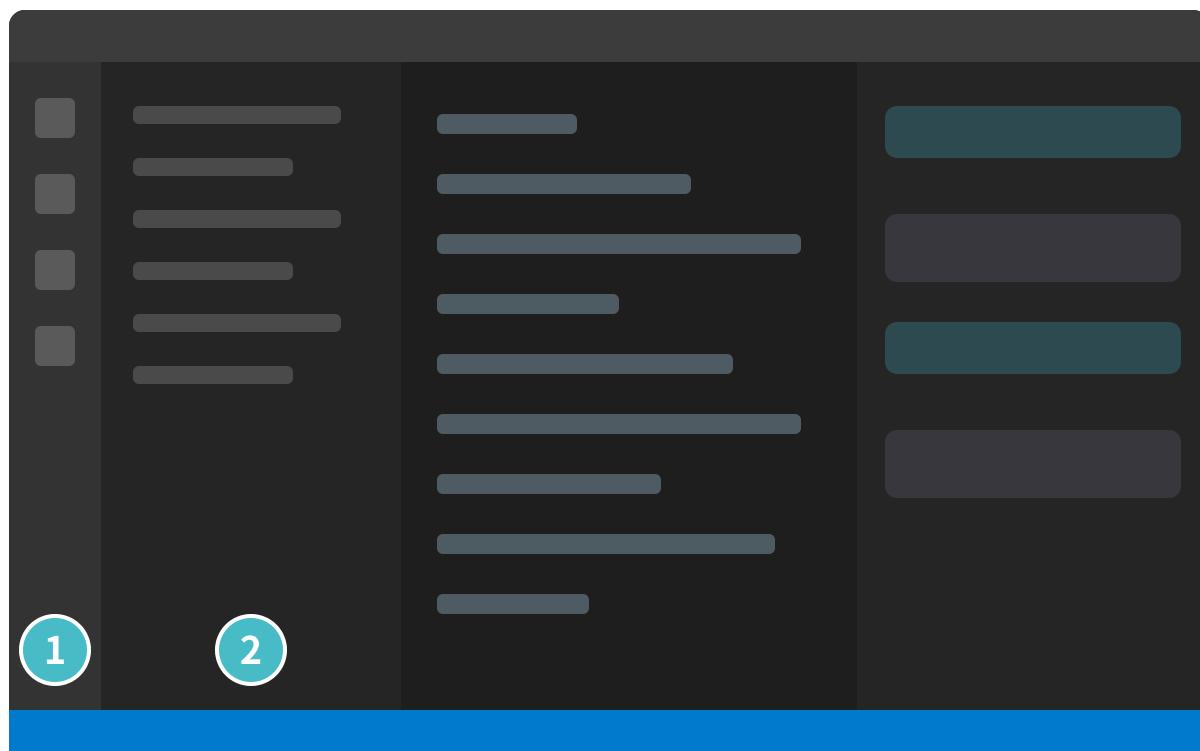
目安は下処理17分、集計14分、内訳と規格外15分、所見12分です。迷ったらこのページに戻ってください

## ファイルはチャットパネルの**入力欄**に落として渡します



- 1 アクティビティバー**  
いちばん左の縦帯。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**  
day3 フォルダの中身が並びます。ここから 設備データ\_2026Q1.xlsx をつまみます
- 3 エディタ**  
ここに落とすとファイルが開くだけです。渡したことにはなりません

## ファイルの渡し方（続き）



### 1 チャットパネル

右側。入力欄に落とすと、入力欄の上にファイル名のチップが付きます

### 2 ステータスバー

下端。Copilotのアイコンに斜線が入っていたら止まっています

## 稼働実績シートで直すのは2か所。欠損だけは埋めずに残します

### 渡したままの稼働実績シート

1行目にタイトルが入り、2行目は空行で、見出しは3行目にあります。稼働時間はすべて「182.5 h」のような単位付きの文字列です。表の途中に空白行が1行あり、稼働時間が1件、停止回数が1件だけ空いています。

### 整形後の表

見出しが1行目に来て、設備8行がそろいます。稼働時間は182.5という数値になります。空白行はなくなり、空いていた2つのセルは空のまま残ります。

## そのまま写して使える指示文です。見出しの位置を最初の1文に入れます

見出しが3行目にあることと、欠損を埋めないことの2つを先に書きます。ここを書かないと、AIは1行目のタイトルを見出しと誤認するか、空の行を勝手に消します。

設備データ\_2026Q1.xlsx の 稼働実績シート を整形してください。

見出しは3行目です。1行目のタイトルと2行目の空行は読み飛ばしてください。

直してほしいことは次の2点です。

1. 空白行を削除する
2. 「182.5 h」のような単位付きの文字列から単位を外し、数値にする

欠損は勝手に埋めず、何行目のどの項目が空かを一覧で報告してください。

整形後の表を表の形で出したうえで、同じ内容を cleaned\_operations.csv として保存してください。

全部を1回で送ります。1行ずつ送ると、AIが前の指示を忘れて別々の表を返してきます



## 整形後の表を8行そろえ、csvに残すところまでを進めます

### どんな場面か

現場から回ってくる表は、タイトル行と単位付きの文字列が入ったまま届きます。この形のまま平均を出すと、単位付きの行が計算から落ちて、実際より小さい数字が出ます。まず計算できる形に直します。

### やること

- 1 前のページの指示文をCopilot Chatに貼り、設備データ\_2026Q1.xlsx を入力欄に落としてから送ります。
- 2 返ってきた整形後の表を画面で数え、設備が8行そろっているかを自分の目で確認します。
- 3 欠損の報告に、稼働時間1件と停止回数1件の2件が挙がっているかを確認します。
- 4 csvが保存されていなければ「この表を cleaned\_operations.csv として保存してください」と追加で頼み、左のエクスプローラーにファイルが増えたことを見ます。

### できあがるもの

day3 フォルダの中の cleaned\_operations.csv。この後の集計とグラフは、すべてこのファイルを使います。

# 下処理の達成基準

## 達成の目安

- 整形後の表が8行ある
- 欠損の報告が 稼働時間1件・停止回数1件 の2件になっている
- 稼働時間の列に h や 時間 の文字が残っていない
- cleaned\_operations.csv が左のエクスプローラーに見えている

## つまずいたら

- 8行より多い。空白行が残っています。「空白行を削除してください」ともう一度頼みます
- 8行より少ない。欠損のある行を削除されています。「欠損のある行は削除せず、空のまま残してください」と頼み直します
- 稼働時間が「182.5 h」のまま返ってくる。単位が外れていません。「稼働時間の単位を外して数値にしてください」ともう一度頼みます
- 列名がおかしい。見出しの指定が抜けています。「見出しは3行目です」を先頭に足して送り直します

## 下処理の達成基準（続き）

---

### 時間が余ったら

- 稼働時間がいちばん長い設備といちばん短い設備を出させ、8台の間で何時間の開きがあるかを確認めます
- 検査記録シートを開いて、日付の書き方が何通りあるかと、全角の数字がどの行にあるかを目で数えます。午前の後半でそのまま効きます

## 2本に分けて頼みます。母数を書かせるのが要点です

1本目で稼働時間の6つの数字、2本目で相関係数を出します。どちらも「何件で計算したか」を書かせてください。母数が分からない平均は、報告書に載せられません。

### 【1本目 稼働時間の6つの数字】

cleaned\_operations.csv の稼働時間について、件数、平均、中央値、標準偏差、最小、最大を表の形で出してください。

標準偏差は、標本標準偏差（ $n-1$ で割る）と母標準偏差（ $n$ で割る）のどちらで計算したかも書いてください。

欠損の件数も別の行で出し、平均は欠損を除いた何件で割ったかを書いてください。

小数は第2位まで表示してください。

### 【2本目 相関係数】

cleaned\_operations.csv の稼働時間と不良数の相関係数を、小数第3位まで出してください。

計算に使った件数も一緒に書いてください。

## 6つの数字と相関係数を出し、**stats.txt**に残します

### どんな場面か

平均だけを渡された報告書は、受け取った側が判断できません。中央値とばらつきを並べて初めて、8台がそろっているのか片寄っているのかが読めます。相関係数は、次にどこを調べるかの当たりを付けるために出します。

### やること

- 1 欠損の1件を平均の母数に入れるか外すかを、自分で決めてから指示文に書きます。今回は外す前提で正解値を用意しています。
- 2 1本目の指示文を貼り、cleaned\_operations.csv を入力欄に落として送ります。表で返ってきたことを確認します。
- 3 続けて2本目の指示文を送り、相関係数と計算に使った件数を受け取ります。
- 4 2つの表をコピーし、day3 フォルダに stats.txt という名前のファイルを作って貼り付け、保存します。

### できあがるもの

day3 フォルダの stats.txt。この中身は休憩明けの報告書にそのまま載せます。

# 集計の達成基準

## 達成の目安

- 件数が7、欠損が1件と出ている。8台のうち稼働時間が空の1台を除いた母数です
- 平均183.64、中央値182.5、最小158.0、最大210.0
- 標準偏差が18.51か17.13のどちらかで出しており、標本と母集団のどちらで計算したかを自分の言葉で言える
- 稼働時間と不良数の相関係数が -0.965
- stats.txt が day3 フォルダにある

## つまづいたら

- 平均が183.64にならない。単位付きの文字列が残っています。稼働時間の列だけを貼り直し、数値になっているか確かめます
- 件数が8で出る。欠損を0として数えています。「空の行は件数に入れないでください」と頼み直します
- 相関係数が出ない。不良数の列が文字列です。「不良数を数値として扱ってください」を足します
- 小数の桁だけが違う。四捨五入の位をそろえれば同じ値です。やり直す必要はありません
- 標準偏差が講師の読み上げと違う場合、計算の種類が違うだけです。18.51なら標本（ $n-1$ で割る）、17.13なら母集団（ $n$ で割る）です。どちらかを確認できていれば達成です。

## 集計の達成基準（続き）

---

### 時間が余ったら

- 停止回数についても同じ6つの数字を出させ、稼働時間より散らばっているかを見ます
- 平均183.64と中央値182.5の差が1.14しかないことから何が言えるかを、隣を見ずに一言で書いてみてください

## 相関係数は向きと強さに分けて読み、**原因とは切り離します**

| 見る場所        | 意味                                  | 今回の値   |
|-------------|-------------------------------------|--------|
| 符号          | マイナスは片方が増えると他方が減る向き、プラスは同じ方向に動く向きです | マイナス   |
| 絶対値が0に近い    | 2つの数字の間に、直線的な並びが見えていない状態です          | 該当なし   |
| 絶対値が0.7を超える | 直線的な並びがはっきり見えている状態です                | 0.965  |
| 今回の値        | 稼働時間が長い設備ほど不良数が少ない、という並びです          | -0.965 |

「稼働時間を延ばせば不良が減る」とは書けません。状態のよい設備に長い仕事が回っているだけ、という読み方も同じデータで成り立ちます



## 相関の $-0.965$ を、**点の散らばり**として1枚の図にします

数字だけでは、外れた設備が1台あるのか全体が並んでいるのかが分かりません。相関を出したこの流れで、散布図を1枚だけ作ります。グラフの部品は同じフォルダの `chart.umd.js` を読ませます。

`cleaned_operations.csv` の稼働時間を横軸、不良数を縦軸にした散布図を、HTMLファイル1枚で作ってください。  
同じフォルダにある `chart.umd.js` を `script` タグで読み込んでください。インターネットからライブラリを読み込む書き方はしないでください。  
データはHTMLの中に直接書いてください。  
どの点がどの設備かが分かるように、設備IDを添えてください。  
ファイル名は `chart_scatter.html` にしてください。

作ったら必ずブラウザで開いて、点が見えるところまで確かめます。枠だけ出て中が白い場合は `chart.umd.js` の読み込み行が抜けています

## 残り3本。規格外だけは**全角数字の始末**を先に書きます

工程別の不良、判定の内訳、規格外のロット。3本を続けて送ります。3本目は工程名の結合セル、5本目は全角数字が仕掛けです。

### 【3本目 工程別の不良】

設備データ\_2026Q1.xlsx の 不良内訳シート の工程別の不良件数を、件数の多い順に表で出してください。合計も出してください。

工程名は縦方向の結合セルでまとめられています。空欄の行には上と同じ工程名を補ってから集計してください。

### 【4本目 判定の内訳】

検査記録シートの判定列を集計して、OKとNGの件数を表で出してください。

### 【5本目 規格外のロット】

検査記録シートの測定値を確認してください。全角の数字が混ざっている場合は半角に直したうえで、9.90未満または10.10を超えるロット番号を表で出してください。

9.90 と 10.10 ちょうどは規格内として扱ってください。

## 工程別、判定、規格外の3つを出して stats.txt に追記します

### どんな場面か

「不良が多い」とだけ言われても現場は動きません。どの工程に何件かが出て、初めて次に見に行く先が決まります。規格外のロット番号は、そのまま現物の追跡に使えます。

### やること

- 1 設備データ\_2026Q1.xlsx を入力欄に落とし直してから、3本目と4本目の指示文を続けて送ります。
- 2 5本目の指示文を送る前に、検査記録シートの測定値の列を上から見て、全角の数字が混ざっている行を自分で見つけます。5本目の1文目は必ず入れます。
- 3 工程別の並び順を、件数の多い順にするか工程の順序どおりにするかを自分で決めて、指示文に足します。
- 4 返ってきた3つの表を stats.txt の下に追記して保存します。

### できあがるもの

stats.txt に3つの表が追記された状態。規格外のロット番号3件は、この後の所見でそのまま使います。

# 内訳と規格外の達成基準

## 達成の目安

- 工程別が 成形33、切削21、検査7 で、合計61
- 判定が OK6、NG3 で、合計9
- 規格外が L-1003、L-1006、L-1009 の3件

## つまずいたら

- 規格外が2件しか出ない。全角のままの L-1003 が数値として読まれず、抽出から漏れています。5本目の1文目を入れて送り直します
- 工程別の合計が61にならない。不良内訳シート of 工程名が縦の結合セルのままで、成形の下2行、切削の下1行、検査の下1行に工程名が入っていません。「空欄の行には上と同じ工程名を補ってから集計してください」を3本目に足して送り直します
- 判定の合計が9にならない。空欄の行を数え落としています。「空欄の行数も出してください」を足します
- 境界値で迷う。9.90 と 10.10 ちょうどは規格内です

## 内訳と規格外の達成基準（続き）

### 時間が余ったら

- 成形の33件が全61件の何パーセントかを出させ、報告書に載せる形にします
- 規格外の3件が上限側に外れているのか下限側なのかを分けて数え、片寄りがあるかを見ます
- 工程別の棒グラフを chart\_defect.html という名前で作らせます。グラフのライブラリは同じフォルダの chart.umd.js を script タグで読み込ませ、データはHTMLの中に直接書かせてください。インターネットから読み込む書き方にとすると、社内網では表示されません

## 事実、解釈、次の確認の順に3行で書きます

### どんな場面か

数字だけ渡すと「で、どうする」を聞き返されます。事実と解釈と次の確認の3行を添えると、受け取った側がその日のうちに動けます。派遣先で数字だけ渡す人と所見を添える人とは、次の依頼の来方が変わります。

### やること

- ① 3つのテーマから1つ選びます。工程別の偏り、稼働時間と不良数の関係、規格外の3ロット、のいずれかです。全部書こうとすると時間が足りません。
- ② 1行目に事実を書きます。数字を1つ以上入れます。たとえば「成形工程の不良が33件で、全61件の半分を超えた」です。
- ③ 2行目に解釈を書きます。原因は書きません。「不良は成形工程に偏っている」までにとどめます。「～のため」「～によって」と書いたら言い過ぎです。

## 所見3行を書く（続き）

### やること

- 1 3行目に次の確認を書きます。誰が何の記録を見るかを書きます。「成形の型交換の記録と突き合わせる」のような形です。
- 2 3行を day3 フォルダの findings.txt に保存し、Copilotに「言い過ぎている箇所を指摘してください」と頼んで自己点検します。

### できあがるもの

day3 フォルダの findings.txt。3行です。休憩明けの報告書にそのまま載せるので、消さずに残してください。

# 所見の達成基準

## 達成の目安

- 1行目に数字が1つ以上入っている
- 2行目に「ため」「から」「によって」が入っていない
- 3行目に、見に行く記録の名前が書いてある
- findings.txt が day3 フォルダにある

## つまづいたら

- 数字が入っていない。stats.txt から該当する数字を1つ持ってきて1行目に足します
- 2行目で原因を断定している。「～のため不良が増えた」を「～に偏っている」に書き換えます
- 3行目が「調査する」で止まっている。誰が何の記録を見るかまで書きます
- 1行も書けない。事実の1行だけ書きます。それでも成果物になります

## 時間が余ったら

- 残り2つのテーマについて、事実の1行だけ書き足します
- 書いた3行をCopilotに読ませ、「この所見を読んだ人が次を取る行動を1つ挙げてください」と聞き、意図どおりに伝わるかを試します



## 午前の締めと休憩

### 4つのファイルがそろっているかを確かめてから休憩に入ります

#### ここまでにしたこと

ここまで：整形後の cleaned\_operations.csv、集計と内訳を書いた stats.txt、散布図の chart\_scatter.html、所見3行の findings.txt。この4つが day3 フォルダにそろっている状態が、午前の到達点です。



#### これからやること

これから：休憩を10分取ります。戻ったら、この3つを1本の報告書にまとめ、ブラウザの印刷機能でPDFにするところまで進めます。新しく入れるソフトはありません。



#### そのあとどこで効くか

そのあと：午後の最終課題では、この進め方をそのまま自分の題材に当てはめます。分析の型が手元に残っていると、題材選びで迷いません。

## 午前からの受け渡し

出した数字と図を、ここから**報告書1本**にまとめます

### ここまでにやったこと

汚れた表を整形し、6つの数字を出し、工程別の棒グラフと稼働時間の散布図まで作りました。相関 -0.965 と規格外3ロットも手元に出ています。



### これからやること

この50分で、出した数字とグラフを report.html という1枚の報告書にまとめ、ブラウザの印刷機能でPDFにします。新しく入れるソフトはありません。



### そのあとどこで効くか

午後は自分で選んだミニアプリを作ります。今日の報告書の作り方は、そのまま職場に持ち帰って別のデータでも使えます。

# 03

## 分析結果のドキュメント化

---

報告書にまとめ、PDFで残すところまで [50min]

## 口頭報告と報告書

### 報告書に**下処理の内容**を残すと、読んだ人が検算できます

#### 口で伝えて終わる

「平均は183.64でした」と伝えても、その183.64がどのデータを、どう直して出した数字なのかは残りません。1週間後に同じ数字をもう一度出せるかも分かりません。聞いた側は結局、自分でデータを見直すことになります。

#### 読める形で残す

何を直して何を計算したかを一緒に書いておくと、読んだ人が自分の手で確かめられます。今日の50分は、この形を1本作ります。中身の分量よりも、下処理の欄が書いてあるかどうかで価値が決まります。

## 6項目をそのままCopilotに渡すと、構成ごと作ってくれます

| 項目      | 書く内容                                                                |
|---------|---------------------------------------------------------------------|
| 表題と日付   | 何についての報告かと、いつ時点のデータかを書きます                                           |
| 対象データ   | 設備データ_2026Q1.xlsx の3シート、設備8件と書きます                                   |
| 下処理の内容  | 単位を外した、日付をそろえた、欠損は埋めずに残した。直した内容を残します                                |
| 集計結果    | 稼働時間の平均183.64、中央値182.5、標準偏差を表で載せます。標準偏差は自分の画面の値をそのまま書き、標本か母集団かを添えます |
| グラフ     | 工程別の不良の棒グラフと、稼働時間と不良数の散布図を貼ります                                      |
| 所見と次の確認 | 事実、解釈、次の確認の3行を並べます                                                  |

## 午前の成果物と使い道

### 午前に作った**5つのファイル**が、そのまま報告書の材料になります

**cleaned\_operations.csv**

整形後の8件です。対象データの欄と下処理の欄の根拠になります

**stats.txt**

件数、平均、中央値、標準偏差、最小、最大の控えです。標準偏差は計算の種類も書き添えてあります。集計結果の表にそのまま貼ります

**chart\_defect.html**

工程別の不良の棒グラフです。成形33、切削21、検査7、合計61件が出ています

**chart\_scatter.html**

稼働時間と不良数の散布図です。相関 -0.965 と一緒に載せます

**findings.txt**

所見の3行です。報告書の最後に置きます

## この7行をそのまま貼れば、報告書のHTMLが1枚返ってきます

day3 フォルダを開いた状態で、Copilot Chat の入力欄に次の7行をそのまま貼ってください。書き写す必要はありません。

設備データの分析報告書を、HTMLファイル1枚で作ってください。

見出しは「設備データ分析報告 2026年 第1四半期」とします。

中身は次の順に入れてください。表題と日付、対象データ、下処理の内容、集計結果の表、グラフ2枚、所見3行。

集計結果は stats.txt の値を、所見は findings.txt の3行をそのまま使ってください。数字は作らないでください。

グラフは chart\_defect.html と chart\_scatter.html 中のコードを貼り込み、同じフォルダにある chart.umd.js を script タグで読み込んでください。

インターネットからライブラリを読み込む書き方はしないでください。

ファイル名は report.html にしてください。

## report.html を1枚作り、ブラウザで開いて中身を確認めます

### どんな場面か

分析した本人が異動しても、残った報告書だけで同じ結論にたどり着ける状態にします。設備の稼働データを見た人が、次にどの記録を確認すればよいかまで書いてあれば、読んだ人がそのまま動けます。

### やること

- 1 day3 フォルダを開いた状態で、前ページの7行を Copilot Chat の入力欄に貼り、送信します
- 2 返ってきたコードを新しいファイルに貼り、report.html という名前で day3 フォルダに保存します
- 3 エクスプローラーで report.html を右クリックし、プログラムから開くを選び、Microsoft Edge を選びます
- 4 表の数字が stats.txt の値と合っているかを目で確認めます。違っていたら「集計結果の表の平均を183.64に直してください」のように、1か所ずつ頼みます

### できあがるもの

day3 フォルダの report.html。ブラウザで開くと、表と2枚のグラフと所見3行が上から順に並んで見える状態にします。



# 報告書の達成基準

## 6つの見出しと3つの数字がそろっていれば達成です

### 達成の目安

- ブラウザで開いて、表題と日付、対象データ、下処理の内容、集計結果、グラフ、所見の6つが上から順に出ている
- 集計結果の表に平均183.64、中央値182.5と、標準偏差（18.51か17.13のいずれか。どちらの計算かを添える）が入っている
- グラフが2枚とも、枠だけでなく中身まで描かれている
- 所見の3行が、事実、解釈、次の確認の順に並んでいる

### つまずいたら

- グラフの枠だけ出て中が白い。同じフォルダの chart.umd.js を script タグで読み込む形に直してください、と頼み直します
- 画面が真っ白。F12を押し、コンソールに出た赤い行をそのままチャットに貼ります
- 数字が stats.txt と違う。Copilotが値を作っています。stats.txt の中身を貼り、この値に置き換えてくださいと頼みます
- 午前のファイルがそろっていない。あるものだけで作り、無い項目は見出しだけ残します

## 報告書の達成基準（続き）

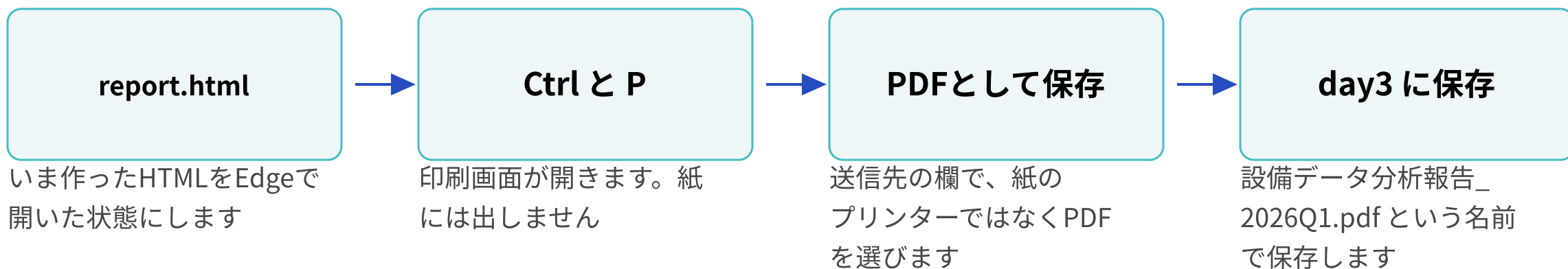
---

### 時間が余ったら

- 所見の3行の下に、次に確認する記録の名前を1行足します
- 同じ内容を、表紙なしで1ページに収まる短い版にするよう頼み、2つを見比べます

## PDF化までの経路

PDFは**ブラウザの印刷機能**で作ります。追加のソフトは使いません



## 印刷画面で3か所だけ設定を変えて保存します

### どんな場面か

報告書を相手に渡すとき、相手のPCにVSCodeが入っているとは限りません。開くだけで読めるPDFにしておくと、そのまま回覧できます。派遣先へ出す資料でも同じです。

### やること

- 1 report.html をEdgeで開いた状態で、Ctrlキーを押しながらPキーを押します
- 2 印刷画面の上のほうにある送信先の欄で、PDFとして保存を選びます。Edgeのバージョンによってはプリンターという名前の欄です
- 3 レイアウトの欄を横にします。表の右端が切れるのを防げます
- 4 その他の設定を開き、背景のグラフィックにチェックを入れます。ここを入れないとグラフの色が出ません
- 5 保存を押し、day3 フォルダに 設備データ分析報告\_2026Q1.pdf という名前で保存します

### できあがるもの

day3 フォルダの 設備データ分析報告\_2026Q1.pdf。ダブルクリックで開いて中身が読める状態にします。

# PDF保存の達成基準

## 2ページ以内で、グラフが色付きで入っていれば達成です

### 達成の目安

- PDFが2ページ以内に収まっている
- グラフが2枚とも色付きで入っている
- 所見の3行が最後のページに出ている
- 表の右端の列が切れていない

### つまずいたら

- グラフが白い枠だけ。その他の設定の、背景のグラフィックのチェックが外れています
- 表が右で切れる。レイアウトを横にします。それでも切れるなら、表の列を4つに減らしてくださいとCopilotに頼みます
- ページが5枚に増える。項目の途中で改ページが入っています。今日は直さなくて構いません
- 保存したPDFが見つからない。保存先がダウンロードフォルダになっています。もう一度CtrlとPからやり直します

## PDF保存の達成基準（続き）

---

### 時間が余ったら

- できたPDFの下処理の欄だけを読み、そこに書いてある内容だけで自分の手順を再現できるかを確認めます
- 表紙なしの1ページ版を作り、どちらが読みやすいかを見比べます

## 切れたPDFと収まったPDF

### 切れる原因は用紙の向きと背景の設定の2つだけです

#### そのまま印刷したPDF

用紙が縦のままだと、集計結果の表の右端が切れます。背景のグラフィックが外れていると、グラフは白い枠だけになり、棒も点も出ません。中身は正しいのに、渡された側は読めません。

#### 2か所直したPDF

レイアウトを横にすると表が収まります。背景のグラフィックにチェックを入れるとグラフの色が出ます。それでもはみ出すときは、表の列を4つに減らしてくださいとCopilotに頼めば済みます。

## 報告書を人に渡す前に、**伏せる4つ**を自分で確かめます

### 客先名と製品名

報告書に入っていないかを見ます。入っていたら、業種と工程の言い方に直します

### プロジェクト名と取引先名

同じく入れません。設備名は装置Aのような記号にそろえます

### 実データと画面の写し

今日のファイルは練習用です。実務では実データそのものや画面の写しを外に出しません

### 出す前の最終確認

社外に出してよい内容かどうかは、出す本人が確認します。判断を人に預けません



## 午後への受け渡し

午前は分析の型でした。午後は**自分で選んだもの**を作ります

### ここまでにしたこと

汚れた表を整形し、集計し、グラフにして、報告書のPDFまで通しました。Copilotへ出した指示は全部で7回ほどです。手順そのものは今日で身につけています。



### これからやること

この20分で、午後に作る業務改善ミニアプリの見本を5つ見て、1つ選びます。合わせて、どこまでできたら完成とするかを作業前に決めておきます。



### そのあとどこで効くか

選んだ見本を130分で作ります。何を作るか迷う時間をこの20分で使い切っておくと、130分が全部つくる時間になります。

# 04

## 業務改善ミニアプリ

---

見本を5つ見て、作るものを1つ決めます [20min]

## 午後の五つの区切り

午後の中心は**130分の開発**です。いまはその手前にいます



## 見本から選び、ブラウザで動かし、今日のうちに動く状態にします

### 見本集から1つ選ぶ

自分の業務そのものを作る回ではありません。前提が多すぎて130分では動くところまで届かないためです。Day2で書いた業務テーマは来週以降の材料に取っておきます

### ブラウザで動く形にする

HTMLファイル1枚で作ります。追加のインストールはしません。作ったファイルをブラウザで開けば、それだけで動く形にします

### 今日のうちに動かす

時間内にブラウザで動くところまで持っていくます。完成度で評価はしません。動いた時点で目標は達成です

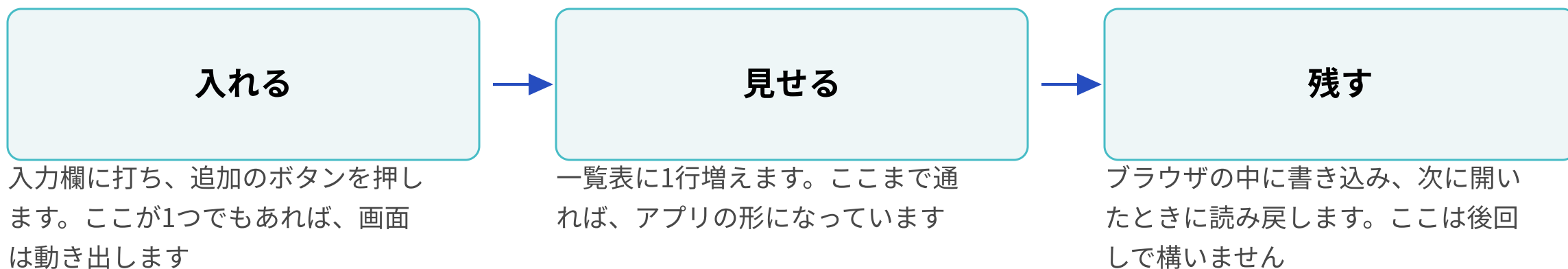
## 見本集の5つ

### 入力するものと画面に出るもので選んでください

| 見本               | 入力するもの           | 画面に出るもの               | つまり点                           |
|------------------|------------------|-----------------------|--------------------------------|
| ① 点検<br>チェックリスト  | 点検項目の名前と<br>チェック | 残りの項目一覧と未実施の<br>件数    | 項目の追加と保存を後から足すと崩れます            |
| ② 測定値の記録と規格判定    | 測定値と規格の上下<br>限   | OKかNGの判定と記録の一覧        | 上下限ちょうどの扱いを先に決めないと<br>件数が合いません |
| ③ 部品在庫の簡易管理      | 部品名と入庫数と出<br>庫数  | 残数の表と、下限を割った<br>行の色分け | 取り消し操作とマイナスの残数で数が合<br>わなくなります  |
| ④ 作業時間の記録と<br>集計 | 開始と終了、作業の<br>種類  | 種類ごとの合計時間と棒<br>グラフ    | 日をまたぐ記録と、分の丸め方でずれます            |
| ⑤ トラブル事例の検<br>索  | 検索する語句           | 一致した事例と対処の一覧          | 事例データを先に10件作らせないと画面<br>が空のままです |

## 三つの共通部品

5つとも、**入れる・見せる・残す**の3つでできています



入れると見せるを先に通します。残すと見た目は後です。この順番は開発の130分で何度も戻ってきます。

閉じて残るのは、**ブラウザの中の置き場**に書いているからです

**保存先**

ブラウザの中にある置き場です。localStorage と呼びます。ファイルとしてフォルダに出てくるわけではありません

**残る条件**

同じPCの同じブラウザで開けば残ります。Edgeで作ったものは、Edgeで開いたときに読み戻されます

**渡らない範囲**

他の人には渡りません。HTMLファイルをメールで送っても、入力した中身は付いていきません

**消える操作**

閲覧履歴の削除をすると消えます。PCの入れ替えでも消えます

**InPrivateの場合**

InPrivateウィンドウで開くと、閉じた時点で消えます

## 見本の選び方

迷ったら①か②にしてください。動きを見てから次を決められます

### 初めて自分で作る方

①点検チェックリストか②測定値の記録と規格判定にします。画面の部品が少なく、指示を1回投げた時点で何かが動きます。動いたものを見てから次の指示を考えられるので、止まりにくいです。②は午前の規格判定と地続きです

### 前回すぐ動いた方

③部品在庫の簡易管理か④作業時間の記録と集計にします。数字の足し引きと集計が入るぶん、指示の書き方で結果が変わります。⑤トラブル事例の検索は、ダミーの事例を10件作らせるところから始まります



## 完成と言える状態

---

完成の線引きは、**作業を始める前に**こちらで決めます

**入力したものが画面に出る  
閉じても消えない**

この2つで完成とします。色や配置の作り込みは完成の条件に入れません。整えるのは、この後の拡張80分の仕事です

## 開発への受け渡し

作るものと完成の線が決まりました。ここから手を動かします

### ここまでにしたこと

見本を5つ見て、入れる・見せる・残すという共通の骨組みと、閉じても残る仕組みを確認しました。完成の線引きも決まっています。



### これからやること

ここから130分で、選んだ見本を作ります。5行の仕様を書き、Copilotに1枚のHTMLを出させ、ブラウザで開いて1つずつ直していきます。



### そのあとどこで効くか

この後の拡張80分で、動いたものに機能を足します。いまは動くところまで手を止めて構いません。

# 05

## 選んだ見本の開発

---

ブラウザで動くところまで、130分で仕上げます [130min]

## 作るものを決め、day3 の中に **miniapp** フォルダを用意します

### どんな場面か

130分で動くところまで届くかどうかは、選ぶのに何分使ったかでほぼ決まります。迷ったまま20分過ぎた人は、この後も同じところで止まります。置き場所も先に固定しておく、自分のファイルを見失いません。

### やること

- 1 見本の表をもう一度見て、30秒で1つ決めます。決まらない人は①点検チェックリストにします
- 2 VSCodeの左のファイル一覧で day3 フォルダを選び、新しいフォルダのアイコンを押して miniapp という名前のフォルダを作ります
- 3 miniapp フォルダを選び、新しいファイルのアイコンを押して index.html という名前のファイルを作ります。中身は空のままで構いません
- 4 Copilot Chat の入力欄に、選んだ見本の番号と名前を1行だけ打っておきます

### できあがるもの

day3 の中の miniapp フォルダと、その中の index.html。中身は空で構いません。左のファイル一覧に2つが見えていれば達成です。

## 準備の達成基準

### 左のファイル一覧に miniapp と index.html が見えていれば達成です

#### 達成の目安

- 左のファイル一覧が day3 の下に miniapp、その下に index.html という並びになっている
- 自分が何番の見本を作るかを、口に出して言える

#### つまずいたら

- miniapp が day3 の外にできた。作ったフォルダを day3 の上へドラッグして移します
- index.html の右側に .txt が付いている。名前を右クリックして名前の変更を選び、.txt を消します
- どれを作るか決まらない。①にします。動いたものを見てから、拡張の時間に別のものへ移れます

#### 時間が余ったら

- 選んだ見本の画面を、手元のメモに手描きで書いてみます。入力欄と一覧表の位置が決まると、次の仕様が書きやすくなります

## 誰が、何を入れ、何が出て、どこに残るか。この5行から始めます

①点検チェックリストを選んだ場合の書き方です。自分が選んだ見本に合わせて、中身の語だけ入れ替えてください。

誰が使うか。設備の日常点検をする担当者が使います。

何を入力するか。点検項目の名前を入力し、確認できたらチェックを付けます。

画面に何が出るか。点検項目の一覧と、まだチェックの付いていない件数が出ます。

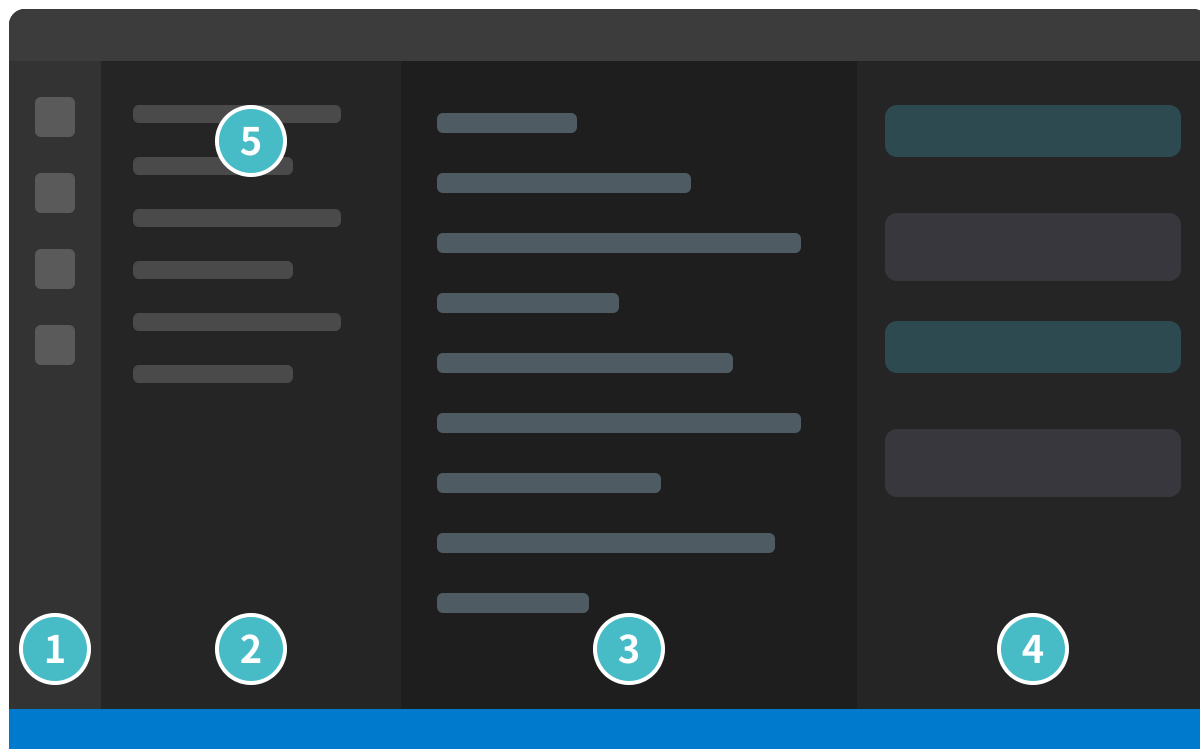
どこに保存するか。ブラウザの中に保存し、閉じて開き直しても残るようにしてください。

作り方の条件。HTMLファイル1枚にまとめ、名前は `index.html` にしてください。インターネットからライブラリを読み込む書き方はしないでください。

埋まらない行は仮の文言で構いません。5行がそろっていることのほうが、1行の精度より効きます。

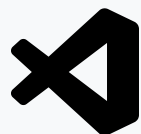
## VSCode画面の五つの部位

仕様を貼るのは**チャットの入力欄**。開くのは左のファイル一覧からです



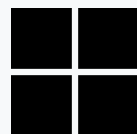
- 1 アクティビティバー**  
画面の左端の縦一列です。このアイコンで、左側に何を出すかを切り替えます
- 2 エクスプローラー**  
day3 と miniapp の中身が並びます。index.html を右クリックしてブラウザで開くのもここからです
- 3 エディタ**  
返ってきたコードを貼る場所です。貼ったら Ctrl と S で保存します。タブの白い丸が消えれば保存できています
- 4 チャットパネル**  
Copilot Chat です。5行の仕様を貼るのは、このパネルの一番下にある入力欄です
- 5 ステータスバー**  
画面の一番下の帯です。右のほうに Copilot のアイコンが出ていれば使える状態です

# VSCodeで書いて保存し、Edgeで開いて動かします



### VSCode

index.html を書いて、  
Ctrl と S で保存する場所  
です



### Edge

保存したファイルを開いて、  
動きを見る場所です



### GitHub Copilot

直したい1文を渡す相手で  
す。VSCodeの中にあります



## 保存しないと、ブラウザを更新しても画面は変わりません

1

### 保存する

エディタで Ctrl と S を押します。タブに出ている白い丸が消えれば保存できています。丸が残っているうちは、画面は変わりません

2

### ブラウザで開く

左のファイル一覧で index.html を右クリックし、プログラムから開くを選び、Microsoft Edge を選びます

3

### 開いたものを確かめる

アドレスバーに file:///C:/Users/ から始まり day3/miniapp/index.html で終わる文字列が出ていれば、いま編集しているファイルを見えています

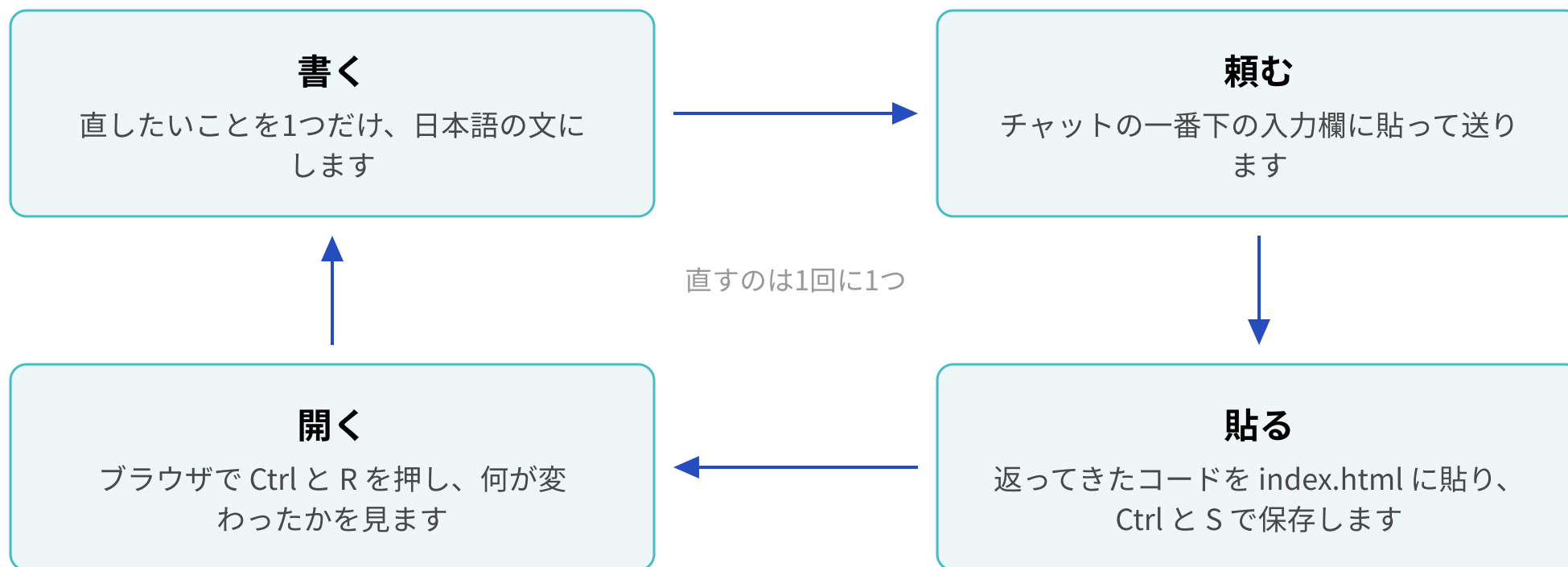
4

### 直したら読み込み直す

保存したあと、ブラウザで Ctrl と R を押します。押さないと前の画面のままです

## 直しの四手

まとめて5個頼まず、**1つ直しては開き直します**



## 50分で一覧が出ていれば、時間内に収まります



## 詰まったときの一手

### 原因は自分で探さず、そのままCopilotに渡してください

| 詰まりの種類        | 最初の一手                                                          |
|---------------|----------------------------------------------------------------|
| 画面が真っ白        | ブラウザで F12 を押し、赤い文字で出ている行をそのままチャットに貼ります。原因の見当は自分で付けなくて構いません     |
| 直したのに変わらない    | 保存できていません。エディタのタブの白い丸を見ます。丸が消えていたら、ブラウザで Ctrl と R を押します        |
| 直すほど壊れる       | index_ok1.html を index.html に上書きして戻します。戻ってから、足す機能を1つに絞って頼み直します |
| 何を頼めばいいかわからない | 今こうなっている、代わりにこうしたい。この2文を書いて渡します。直し方までは指定しなくて構いません              |

このページは作業中も投影したままにします。顔を上げたときに、次の一手が目に入る状態にしておきます。

## 選んだ見本を、入力が一覧に出て閉じてでも残る状態まで作ります

### どんな場面か

自分の手元の作業を、自分で作った画面で回せるようにします。ここから100分は個人作業です。完成度は問いません。入力したものが一覧に出て、閉じてでも残るところまでを目指します。

### やること

- 1 5行の仕様を書き、Copilot Chat の一番下の入力欄に貼って送ります
- 2 返ってきたコードを index.html に貼り、Ctrl と S で保存します
- 3 index.html を右クリックしてEdgeで開き、実際に文字を入力して確かめます
- 4 動いたら、左のファイル一覧で index.html をコピーして貼り付け、index\_ok1.html という名前にします。壊れたらここへ戻ります
- 5 直したいところを1つだけ文にしてチャットに貼り、返ってきたコードを貼り直して開き直します。これを繰り返します

### できあがるもの

day3 の miniapp フォルダの中の index.html と、動いた時点のコピー index\_ok1.html。

## 開発の達成基準

# 一覧に出る、閉じても残る、コピーがある。この3つを自分で確かめます

### 達成の目安

- 入力欄に文字を打ってボタンを押すと、一覧に1行増える
- ブラウザのタブを閉じて index.html を開き直しても、さっき入れた行が残っている
- 動いた時点のコピーが miniapp フォルダに1つ以上ある

### つまずいたら

- 25分たっても画面に何も出ない。仕様に「画面に表示する項目」の1行を足して頼み直します。それでも出なければ講師を呼びます
- 一覧には出るが、閉じると消える。「入力した内容をブラウザの中に保存し、開いたときに読み戻す形に直してください」と1回で頼みます
- 直すほど壊れる。index\_ok1.html を index.html に上書きして戻し、足す機能を1つに絞ります
- 記録する項目が多くて画面に入らない。3つに減らします。3項目あれば形になります

## 開発の達成基準（続き）

---

### 時間が余ったら

- 一覧の行を消せるようにします。消したあとに閉じて開き直し、消えたままかを確認めます
- 一覧の内容をCSVファイルとして書き出すボタンを足します
- 自分が選ばなかった見本の骨組みを、いまの index.html と見比べて、どこが同じかを探します

## 開発から拡張へ

### 動いているものに手を足す80分に入ります

#### ここまでにしたこと

開発の130分で、選んだ見本を1つ、ブラウザで動くところまで作りました。入力したものが一覧に出る状態まで来ています。保存まで届いた方もいます。



#### これからやること

この80分は、動いているものに足す時間です。入力、集計、グラフ、保存、見た目の5つから2つまで選びます。壊しても戻る地点があるので、遠慮なく足してください。



#### そのあとどこで効くか

足した1つが、この後に書くアクションプランの中身になります。来週の自分に渡す材料を、ここで1つ作っておきます。

戻る地点は day3/miniapp の index\_ok1.html です。開いたまま置いておいてください。



# 06

## 作ったものの拡張

---

5つの方向から2つまで選び、動く状態を保ったまま足します [80min]

## 午後の現在地

残りは拡張80分と、明日の計画に60分です



## 入力、集計、グラフ、保存、見た目の5つから2つまで選びます

### 入力を増やす

記録する項目に担当者名や設備名を足します。後から絞り込みたくなる項目を先に入れておくと、あとで作り直しになりません。5つの中でいちばん壊れにくい方向です

### 集計を足す

件数、合計、平均のどれかを画面に出します。午前に稼働時間の平均183.64を出したのと同じ計算を、アプリの中でやらせるだけです

### グラフを付ける

同梱の chart.umd.js を読み込んで、棒グラフを1つ出します。追加のインストールはしません。インターネットからライブラリを読ませないよう指示に書きます

3つ以上まとめて足すと、どの指示で壊れたのかが追えなくなります。2つで止めてください。

## 拡張の5つの方向（続き）

---

### 保存できるようにする

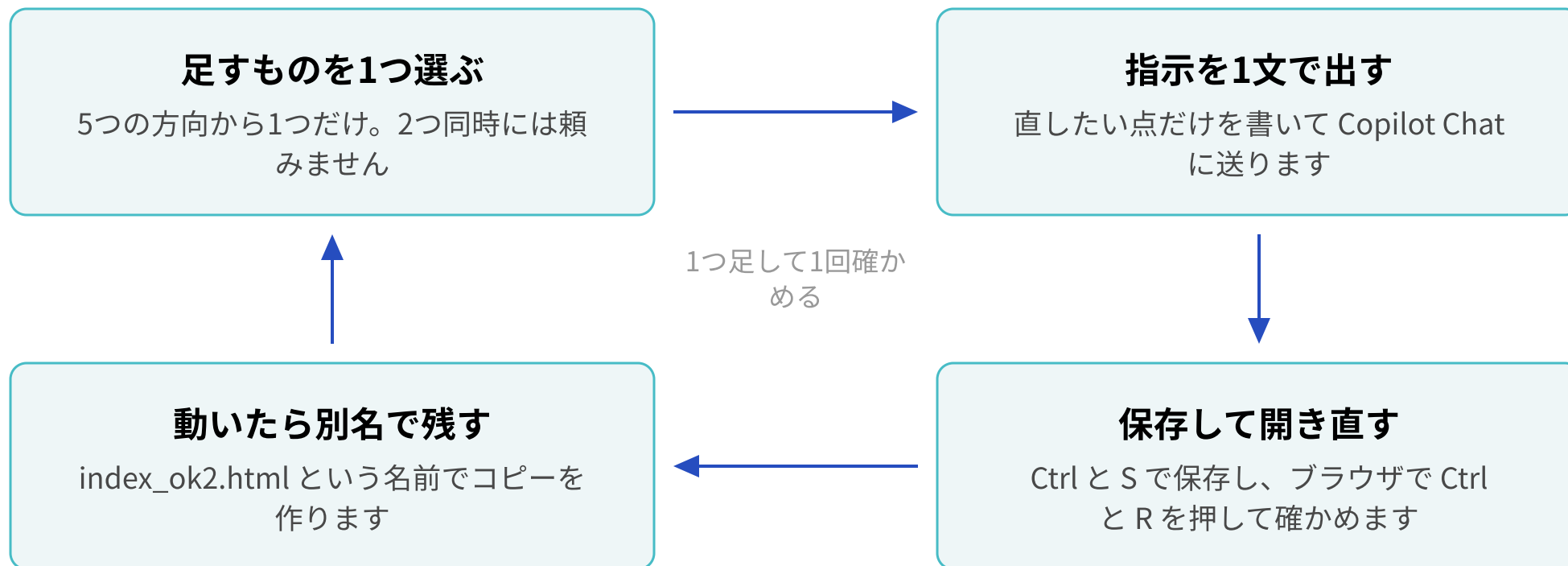
入力したものを、ブラウザを閉じてでも残るようにします。開発で保存を後回しにした方は、まずここを選んでください

### 見た目を整える

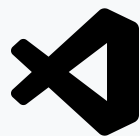
文字を大きくし、行の間隔を空けます。現場で使う画面は、離れて見て読めるかどうかで決まります

## 拡張の回し方

### 1つ足すごとに開き直して、動いたら名前を変えて残します

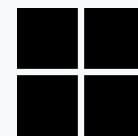


直すのはVSCode、**確かめるのはEdge**です



### VSCode

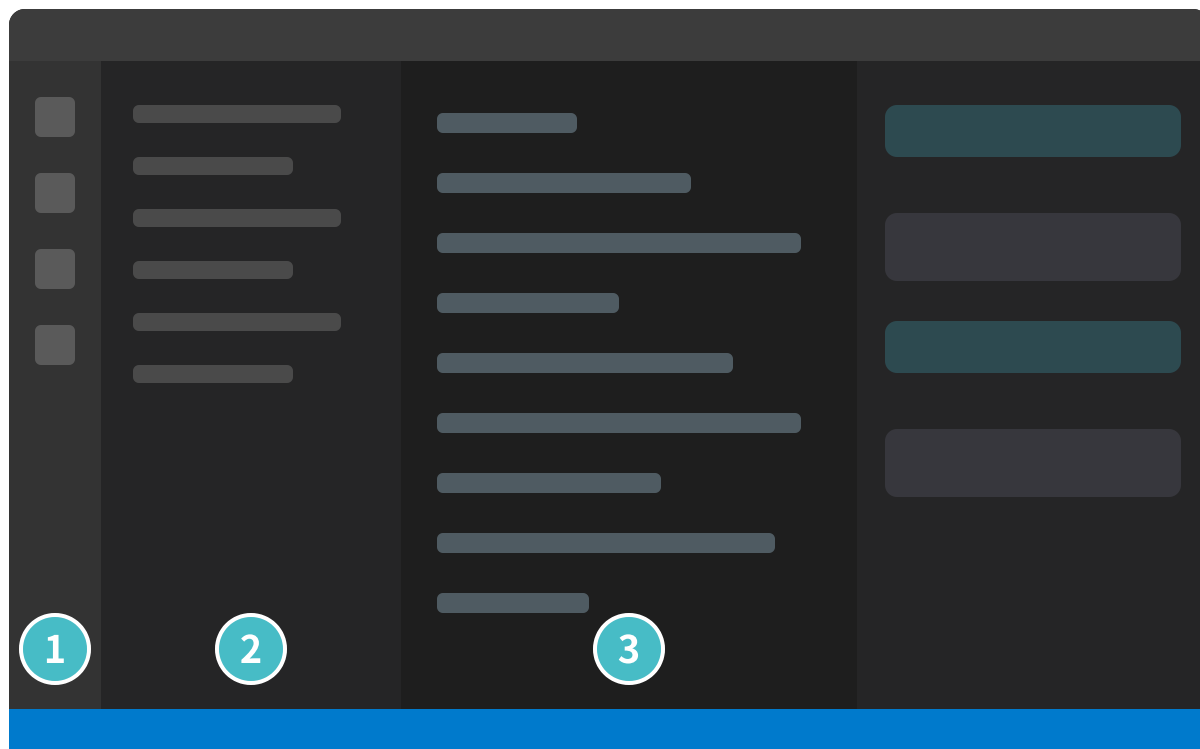
返ってきたコードを貼り、  
保存してコピーを残す場所  
です



### Edge

Ctrl と R で更新し、足し  
た1つが出ているかを見る  
場所です

## 見る場所は左のファイル一覧、タブの丸印、チャット欄の3つです



### 1 アクティビティバー

いちばん左の縦帯。ファイル一覧が消えたら、いちばん上の書類のアイコンで戻します

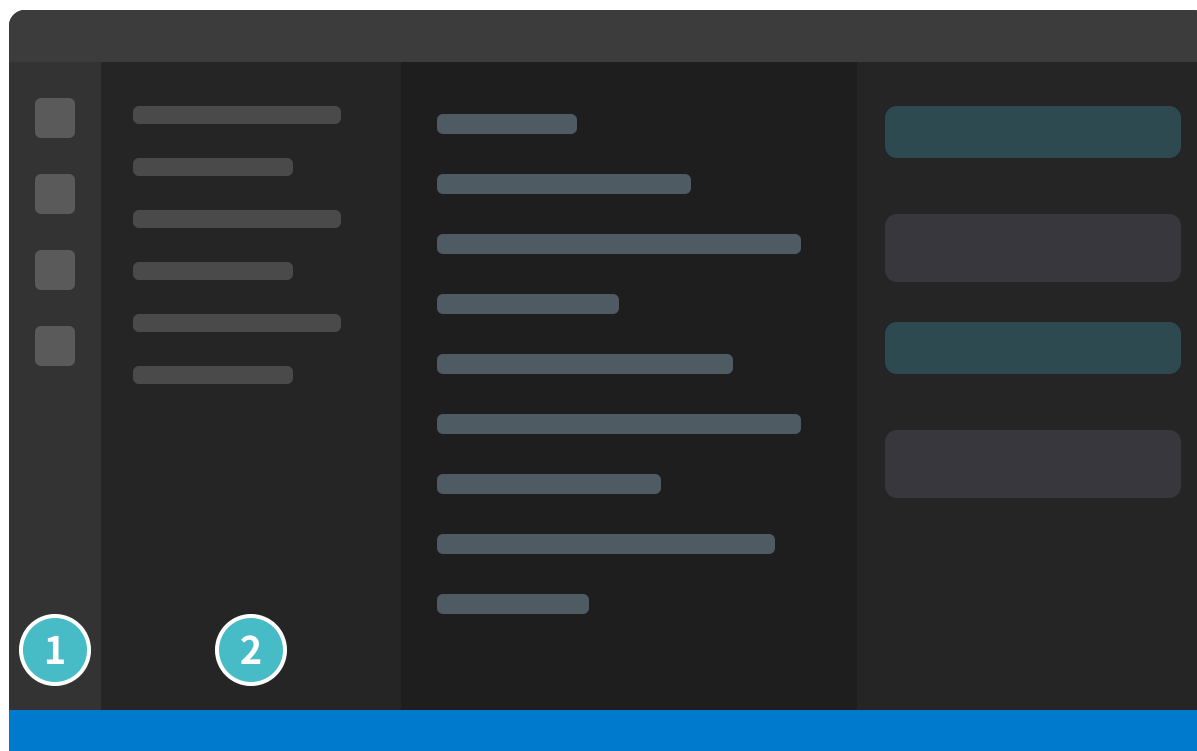
### 2 エクスプローラー

day3/miniapp の中身が並びます。index\_ok1.html が戻る地点です

### 3 エディタ

開いているファイル。タブに白い丸が残っていれば、まだ保存されていません

## 拡張中に見る画面（続き）



### 1 チャットパネル

直す指示を1文ずつ入れる場所。左の吹き出しアイコン、または Ctrl と Alt と I で開きます

### 2 ステータスバー

下の帯。ここは今日は使いません。表示が変わっても作業に影響しません



## 直したい点を**1つだけ**書いて送ります。下の3本は写して使えます

5つの方向のうち、入力、集計、グラフの3本を例文にしました。今ある動きは変えないでください、の一言を必ず付けます。この一言が無いと、直したところ以外まで作り直されて一覧が消えます。

入力欄に「担当者」を1つ足してください。一覧表にも担当者の列を足してください。今ある入力と一覧表の動きは変えないでください。

一覧の下に、登録件数と未実施の件数を表示してください。行が増えたときに自動で数え直してください。今ある一覧表の見え方は変えないでください。

同じフォルダにある `chart.umd.js` を `script` タグで読み込んで、作業の種類ごとの合計を棒グラフで表示してください。インターネットからの読み込みはしないでください。今ある一覧表はそのまま残してください。

保存と見た目を選んだ方は、この3本の書き方をまねて自分で1文にしてください。

## 5つの方向から2つまで選び、足すたびにブラウザで確かめます

### どんな場面か

開発の130分では、入力したものが一覧に出るところまでを目標にしました。ただ、自分が使う側になると足りないものが見えてきます。誰が入力したのか分からない、何件あるのか数えられない、閉じたら消える。現場に置く画面はここから先が要ります。この47分で、そのうち2つまでを埋めます。

### やること

- 1 day3/miniapp の index.html を VSCode で開き、ブラウザにも表示した状態にします
- 2 5つの方向から2つまで選び、選んだものを手元のメモに書きます。書いてから画面に向かいます
- 3 Copilot Chat に直す指示を1文で送ります。2つをまとめて頼みません
- 4 返ってきたコードを index.html に貼り、Ctrl と S で保存してから、ブラウザで Ctrl と R を押して確かめます
- 5 動いたら index\_ok2.html という名前でコピーを作ります。2つ目に進むのは、このコピーを作った後です

### できあがるもの

day3/miniapp の中に index.html と index\_ok2.html。2つ目まで足せた方は index\_ok3.html まで作ります。提出はしません。

## 足した1つが画面に見えて、元の動きが壊れていなければ達成です

### 達成の目安

- ブラウザで Ctrl と R を押しても、入力から一覧表示までが開発のときと同じに動く
- 足した1つが画面に見えている。列が増えた、件数が出た、グラフが出た、閉じても残った、文字が大きくなった、のいずれか1つ
- 左のファイル一覧に index\_ok2.html が見えている

### つまずいたら

- 画面が真っ白になった。index\_ok1.html の中身をすべてコピーして index.html に貼り戻します。そのうえで直前の指示を2文に分けて出し直します
- グラフが出ない。chart.umd.js が同じフォルダにあるか、script タグの指定がファイル名だけになっているかを見ます。インターネットのURLになっていたら消させます
- 一覧が消えた。今ある一覧表の動きは変えないでください、を指示に足してもう一度頼みます
- 何を足すか決まらない。入力を増やすを選びます。5つの中でいちばん壊れにくく、10分で1つ終わります

## 拡張の達成基準（続き）

### 時間が余ったら

- 入力を増やす方向の続き。日付の欄を足し、入力しなかった行には今日の日付が自動で入るようにします
- 集計を足す方向の続き。件数だけでなく平均も出し、小数第1位までにそろえます
- グラフを付ける方向の続き。棒を数の多い順に並べ替え、色は1色にします
- 保存できるようにする方向の続き。一覧を空にするボタンを足し、押す前に確認を出すようにします
- 見た目を整える方向の続き。文字を大きくしたうえで、自分が1歩下がって画面を見て読めるかを確認めます

## 同じ集計が、JavaScriptでは13～15行、pandasでは1行です

### 今日のブラウザ方式

工程ごとの不良数を数える処理は、JavaScriptで13～15行かかります。時間帯ごとの平均を出す処理は14行で、記録が無い時間帯は表から消えます。消えた行を残すには、時間帯の一覧を先に作って埋める処理を自分で足すことになります。

### Pythonがある場合

同じ集計は pandas で1行です。時間帯ごとの平均も1行で、記録が無い時間帯を残す指定も引数1つで済みます。表を扱う道具が最初から揃っているぶん、書く量が減ります。ただし導入には管理者権限とインストールの手続きが必要です。

速度は変わりません。差が出るのは扱える**処理の種類**です

| 観点             | ブラウザ方式        | Pythonがある場合       |
|----------------|---------------|-------------------|
| 表の集計           | 13～15行書けばできる  | 1行でできる            |
| 時刻がずれたログの突き合わせ | 相当する機能がない     | 時刻の近さで結合できる       |
| 分散分析・重回帰       | 相当する機能がない     | 統計の道具が揃っている       |
| 波形の信号処理・異常検知   | 相当する機能がない     | フィルタから学習まで一続きでできる |
| 5万行の処理速度       | 17ミリ秒         | 38ミリ秒             |
| 扱える行数の目安       | 20万行あたりから重くなる | 数十万行でも扱える         |

今日のやり方で足りるうちは、今日のやり方でいきます

## 今日の規模ならブラウザで足ります 数十万行になったらPythonの相談

設備8件の規模なら、どちらで書いても結果は同じです。行数が数十万に増える、複数のログを突き合わせる、統計で原因を切り分ける。このどれかが出てきたら、自社の情報システム部門に導入の相談を始めてください。管理者権限とインストールの手続きが要ります。

## 拡張から明日の計画へ

### 来週やることを**1つだけ**、日付を付けて書き出します

#### ここまでにしたこと

午前データを整理して所見を書き、午後にミニアプリを1つ動かして、拡張で2つまで足しました。今日作ったファイルは自分のPCの day3 フォルダに残っています。



#### これからやること

この40分で、来週やることを1つだけ書き出します。型を渡し、見本を見せ、書く言葉の伏せ方を確認してから、記入の時間を取ります。提出はありません。



#### そのあとどこで効くか

書いた1つを来週やるかどうかで、3日間が残るかどうかが決まります。今日の操作は1週間あれば手が忘れます。



# 07

## アクションプラン

---

来週やることを1つ、日付と終わりの合図まで決めます [40min]

## 下の5行を写して、右側を自分の言葉で埋めてください

day3 フォルダに action.txt という名前で保存するか、配布した 02\_アクションプラン記入シート.html をブラウザで開いて印刷し、その記入欄に書いてください。書く順番は上から。いつ、を先に埋めます。何をするかを先に考えると手が止まります。

いつ : 来週の ( ) 曜日、( ) 分使う  
何をするか : 見本①～⑤から1つ、または自分の業務から1つ  
終わりの合図 : これができたら終わり、と自分で言える状態を1行  
使う範囲 : まず自分の手元だけ。人に配るのは自分で1週間使ってから  
伏せるもの : 客先名、製品名、プロジェクト名、取引先名は業界と工程のレベルに丸める

action.txt は VSCode で新しいファイルを作って書けます。記入シートを使う方は、ブラウザで開いて Ctrl と P で印刷してください。書いた紙は写真を撮らずそのまま持ち帰ってください。



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

## 一歩は30分で終わる大きさにしてください

| いつ        | 何をするか                          | 終わりの合図                  |
|-----------|--------------------------------|-------------------------|
| 来週の月曜 30分 | 始業前点検のチェックリストを自分用に作る           | 自分の点検1回分が、その画面だけで終わる    |
| 来週の水曜 20分 | 毎週作っている集計表の手順を Copilot に説明してみる | 手作業のどこを任せられるかを3つ書き出せている |
| 来週中 15分   | 測定値の規格判定を1品目分だけ作る              | OKとNGが自分の目視と一致する        |
| 今月中 60分   | トラブル事例を10件だけ入力して検索できるようにする     | 過去の事例が語句で引ける            |

## 午前の報告書と同じ4ルールの中で、アクションプランも書きます

### 名前を丸める

客先名、製品名、プロジェクト名、取引先名は書きません。自動車部品の加工工程、樹脂成形の検査といった業界と工程のレベルに直します

### 実データを持ち込まない

実際の測定記録や業務ファイル、業務画面のスクリーンショットは使いません。桁数と項目の形だけをまねた練習用の値にします

### 具体値を避ける

規格の上下限や単価のような具体値は書きません。今日使った9.90から10.10のような練習用の値に置き換えます

### 出す前に自分で確認

社外に出しても問題ない内容かどうかは、書いた本人が確かめます。判断に迷うものは自社の窓口に聞きます

派遣先のPCでCopilotを使えるかどうかは、自社の窓口を確認してください。自分の判断で入れません。

## 日付から埋めて、来週やることを1つ書き上げます

### どんな場面か

3日間で操作が分かっていても、来週の予定表に入っていないものは実行されません。研修の翌週に自分の業務で1回でも使うかどうかで、ここから先が分かります。日付の入っていない計画は動かないので、日付から埋めます。

### やること

- 1 型の5行を action.txt に写します。紙で書く方は 02\_アクションプラン記入シート.html を印刷したものに書きます
- 2 いつ、から埋めます。来週の曜日と使う分数を決めます。何をするかより先に日付です
- 3 見本①～⑤のうち、自分の担当業務に一番近い1つを選んで、何をするかに書きます
- 4 終わりの合図を1行書きます。自分が見て、終わったと判断できる形にします
- 5 書き終わったら伏せ方4ルールで読み返し、名前が入っていたら業界と工程に直します

### できあがるもの

day3 フォルダの action.txt。02\_アクションプラン記入シート.html を印刷して書いた方は、その紙が成果物です。提出はしません。

## 日付、1つだけ、終わりの合図、名前なしの4点で確かめます

### 達成の目安

- いつ、に来週の曜日と使う分数が入っている。今月中や近いうちは書き直します
- 何をすることが1つだけになっている。2つ以上書いた方は、来週やる1つに丸を付けます
- 終わりの合図が、自分が見て判断できる形になっている。効率化する、慣れる、では判定できません
- 客先名、製品名、プロジェクト名、取引先名が1つも入っていない

### つまずいたら

- 何を書けばいいかわからない。来週の月曜、30分、から先に埋めます。日付が決まると中身は後から出てきます
- 業務全体をAIで効率化する、と書いた。見本の表に戻って、30分で終わる大きさに切ります
- 選んだ見本が自分の業務に合わない。見本①の点検チェックリストに寄せます。どの職種でも点検か確認の作業はあります
- 派遣先で使えるかわからず止まった。使える前提で中身を書き、来週の1歩目を自社の窓口を確認するにします

## 書けたかの基準（続き）

---

### 時間が余ったら

- 1つ書けた方は、その1つが終わった後にやることを2つ目として書きます。来週やるのは1つ目だけです
- 今日作ったミニアプリを来週どこまで足すかを、1行だけ書き足します
- 今日そのまま使えた指示文を1つ選び、action.txt の末尾に貼って持ち帰ります



## 計画から振り返りへ

### 3日間から**1つだけ**選んで持ち帰ります

#### ここまでにしたこと

拡張で2つまで足し、来週やることを1つ書きました。手元には day3 フォルダのファイルと、書いたアクションプランが残っています。



#### これからやること

残りの20分で3日間を振り返ります。一番使えると思った操作を1つ、手元に書き出してもらいます。発表はしません。最後に質疑と事務の連絡を置きます。



#### そのあとどこで効くか

書いた操作が、来週の1歩目で最初に手を動かす部分になります。3日分を全部覚えて帰るのは無理なので、1つに絞ります。

# 08

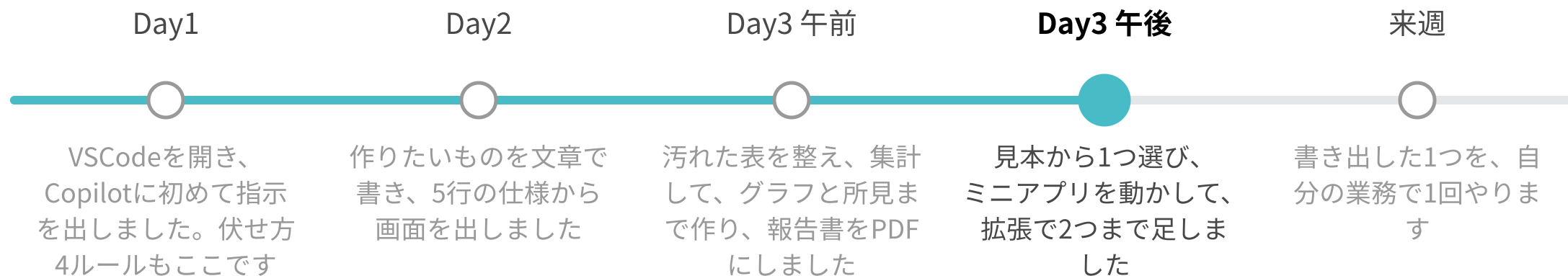
## 修了と振り返り

---

3日間で身に付いた操作を1つ選んで持ち帰ります [20min]

## 3日間の道筋

### 環境の入れ方から、動くアプリ1つまで来ました



## 一番使えると思った操作を、使う場面つきで1行書きます

### どんな場面か

3日分の操作を全部覚えて帰るのは無理です。1つに絞ると、来週それを使います。使わなかった残りは、必要になったときに資料を開けば戻せます。

### やること

- 1 手元のメモか action.txt の末尾に、3日間で一番使えると思った操作を1つ書きます
- 2 操作の名前だけでなく、どの場面で使うかを同じ行に書きます

### できあがるもの

成果物の提出はありません。1行が手元に書けていれば達成です。

## 明日1人で再現できる操作を選べていれば達成です

### 達成の目安

- 1行の中に、操作と使う場面の両方が入っている
- 書いた操作を、明日1人で再現できる。できないものは選び直します

### つまずいたら

- 3つ書いてしまった。来週1回目に使うものだけに丸を付けます
- 何も浮かばない。Copilot Chatを開いて指示を出す操作、作りたいものを5行で書く操作、表をチャットに読ませて整えさせる操作の3つから選びます
- 選んだ操作の手順が思い出せない。それは明日1人で再現できないので、もっと単純なものに変えます

### 時間が余ったら

- 選んだ操作を、隣の方に説明できる長さで2行に書き直します。声に出さなくて構いません
- 3日間で一度も使わなかった操作を1つ挙げ、来月試すものとして書き留めます

## 3日間の到達点

---

3日間で、環境の設定から動くアプリの完成まで来ました

### 3日前は環境の入れ方から 今日は動くアプリを1つ

Day1でVSCodeを開き、Day2で仕様を文章で書き、今日はデータを整えて分析し、ミニアプリを最後まで動かしました。手が止まったときにどう戻るかも一度は通しています。来週、書き出した1つをやってみてください。