

TRAINING

AI協働開発ハンズオン講座

機電系トラック Day1

環境セットアップ・生成AIの基礎・リテラシー・GitHub Copilot操作



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。

本日のゴール

今日の終わりに、**自分のPCでCopilotが動く状態を作ります。**

自分のPCで Copilotが動く

VSCodeでサインインし、日本語で話しかけて返事が返るところまでを、全員そろって確認します

ゴールの判定条件

達成できたかは、この4つで自分で判定できます。

環境が動く

VSCodeの左下に自分のGitHubユーザー名が出て、チャット欄から日本語の返事が返る

生成AIの輪郭

午前の最後のクイズに、迷わず答えられる

守る線がわかる

午後のクイズで、客先の固有情報が入った相談文を選ばずに済む

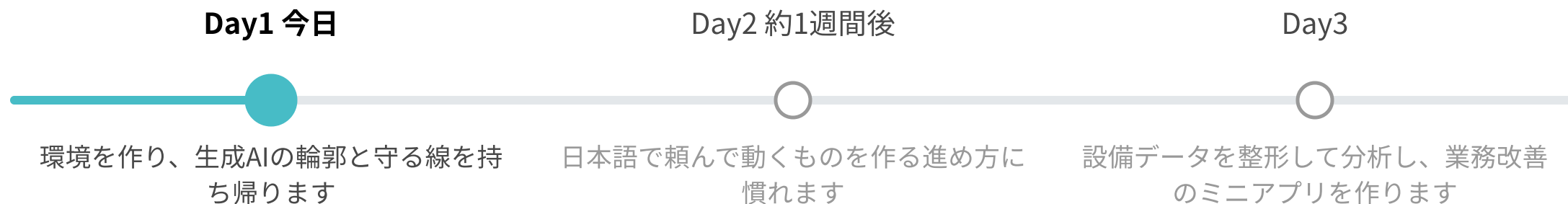
触った実感

午後の演習で作った画面が、自分のブラウザで動く

4つ全部を完璧にする必要はありません。1つ目が今日の山です。

3日間の位置

今日は3日間の1日目です。今日できなくても翌週取り返せます。



今日の流れ

午前は環境づくり、午後は考え方と操作に充てます。

区分	内容	時間
午前	環境セットアップ+GitHubアカウント作成+Copilot招待・動作確認	[110min]
午前	休憩	[10min]
午前	生成AIの基本	[60min]
午後	リスク・リテラシー・ガバナンス+機密情報の取扱い	[70min]
午後	休憩	[10min]

合計480min。上の[10min]は小休憩で、昼食は別枠です。

今日の流れ（続き）

区分	内容	時間
午後	VSCoideの使い方	[40min]
午後	GitHub Copilot基本操作 プチハンズオン5演習	[100min]
午後	理解度テスト	[40min]
午後	振り返り+翌週のテーマ案内	[40min]

環境づくりへ

ここまでにしたこと

ここまでで、今日1日でどこまでやるか、3日間のどこにいるかを共有しました。



これからやること

ここからの110分で、全員のPCでVSCodeとGitHub Copilotが動く状態を作ります。先に画面の名前とチャット欄の開き方を押さえてから、アカウント作成に入ります。



そのあとどこで効くか

この環境が動いていないと午後の5つの演習が全部止まります。翌週のDay2も同じ環境をそのまま使います。

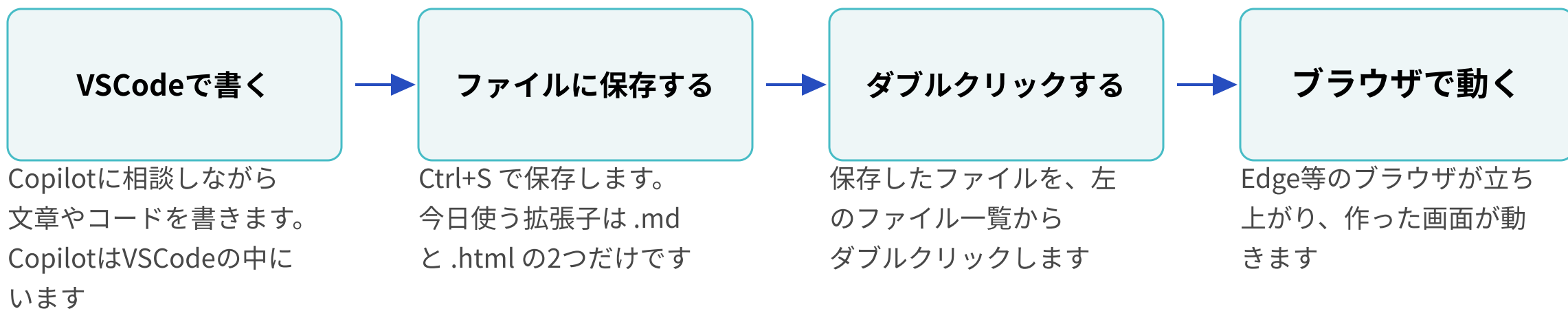
01

環境セットアップ

VSCodeとGitHub Copilotを、自分のPCで動く状態にします

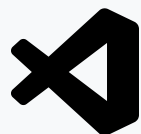
環境の全体像

VSCodeで書いて保存し、**ブラウザで開く**。今日はこれだけです。



Python も Node.js も出てきません。追加のインストールは今日ありません。

今日触るのは**この3つ**だけです。追加のインストールはありません。



VSCode

文章とコードを書く画面。
Copilotはこの中にいます



GitHub Copilot

日本語で相談する相手。
VSCodeの右側にチャット欄が
開きます

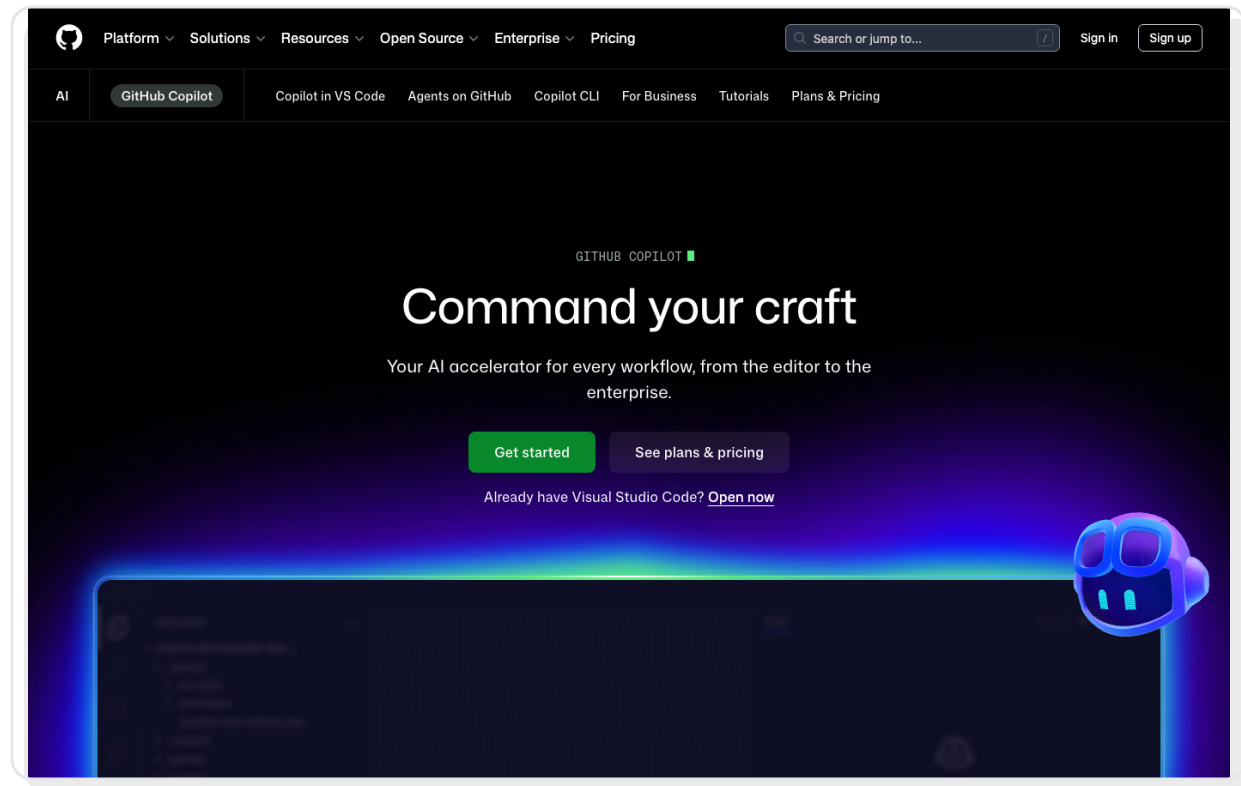


Edge

作ったファイルを開いて動か
すブラウザ。貸与PCに最初か
ら入っています

GitHub Copilotとは

Copilotは、GitHub社がVSCodeの中で提供している相談相手です。



GitHub Copilot の公式サイト

アカウントを作るのはGitHub、使うのはCopilotです。



GitHub

開発の現場で広く使われている場所。ここにアカウントを作ります

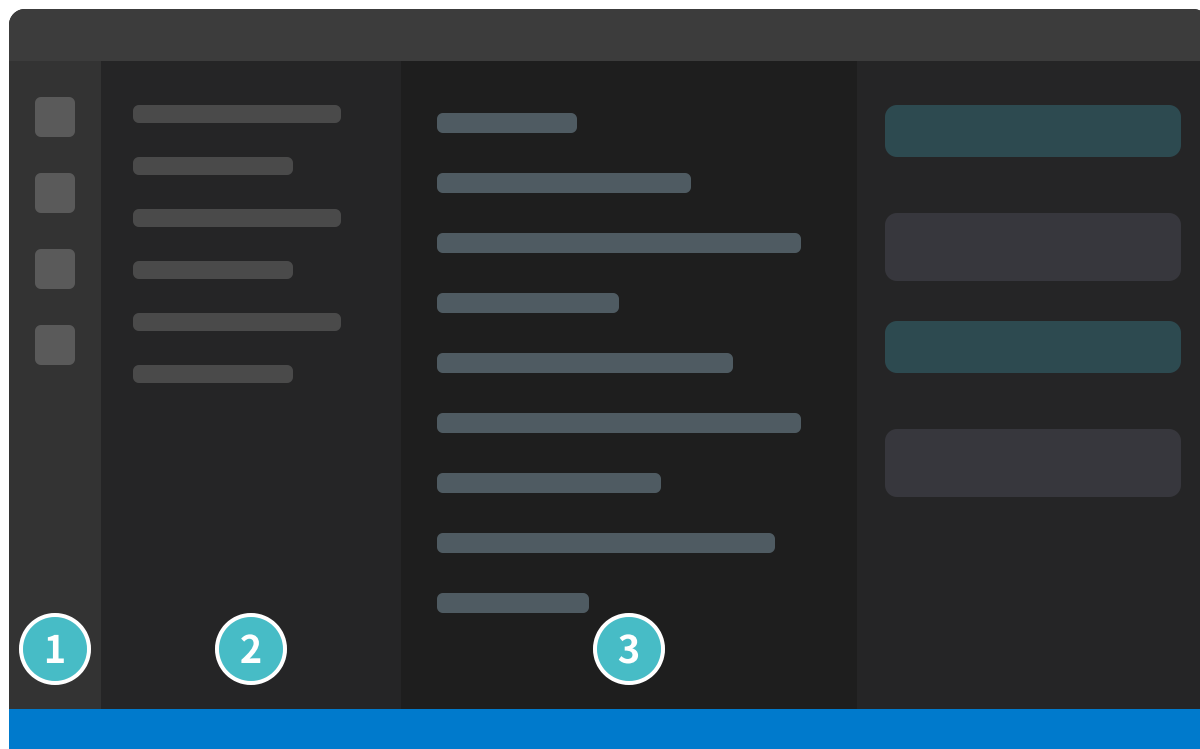


GitHub Copilot

そのアカウントで使える相談相手。VSCodeの中から呼び出します

VSCodeの画面

今日使うのは、この画面の中の3か所だけです。



1 アクティビティバー

左端の縦一列。書類が重なったアイコンを押すと
ファイル一覧が出ます

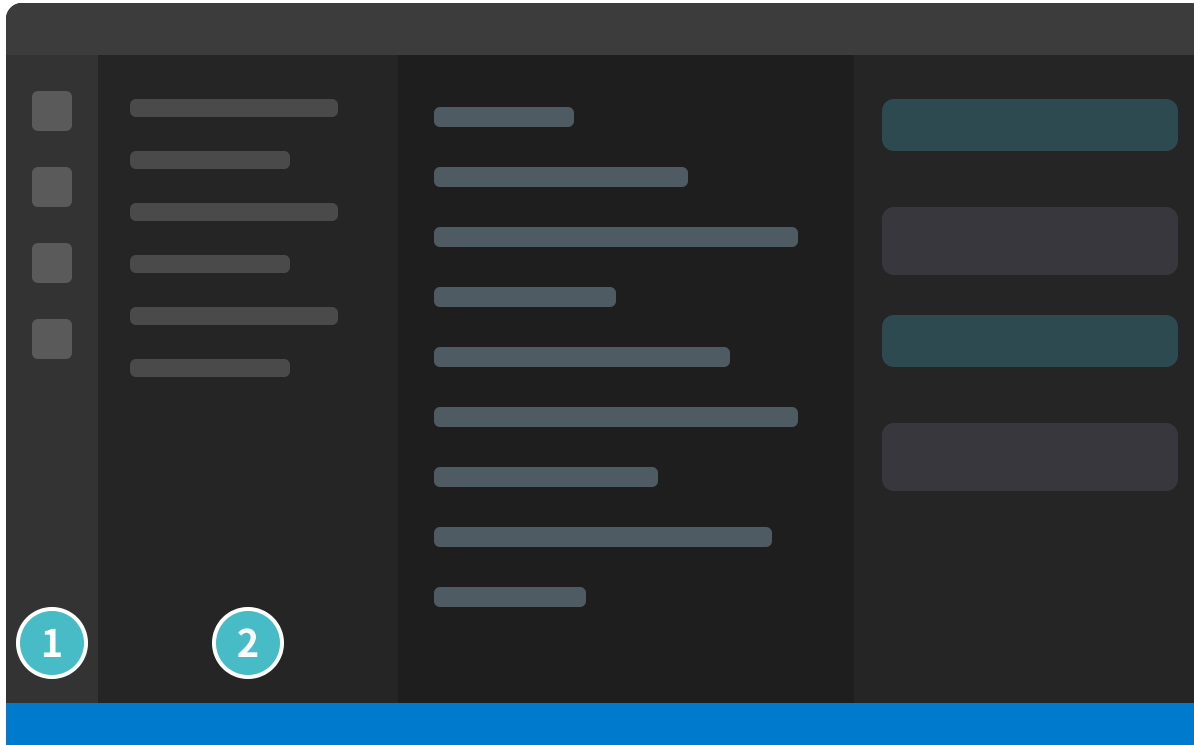
2 エクスプローラー

開いているフォルダの中のファイル一覧。クリックす
ると右に開きます

3 エディタ

文章やコードを書く場所。タブで複数開けて、×で閉
じます

VSCodeの画面（続き）



1 チャットパネル

Copilotに日本語で相談する場所。画面の右側に開きます

2 ステータスバー

下端の細い帯。左下にサインイン中のアカウントが出ます

チャット欄の開き方

右上のCopilotアイコンを押すと、右側にチャット欄が開きます。

1

右上のCopilotアイコンを押す

画面上端の右寄りにある、吹き出しの形をしたアイコンです

2

右側にパネルが開く

編集画面の右に縦長のパネルが出ます。閉じるときは同じアイコンをもう一度押します

3

下の入力欄に打ってEnter

日本語のまま入力します。数秒で返事が返ってきます

アイコンが見つからないときは、メニューの表示 から Copilot Chat を探してください。

4ステップを順番どおり進めます。飛ばさないでください。

どんな場面か

今日の午後の5つの演習も、翌週のDay2も、この環境をそのまま使います。ここで止まったまま午後に入ると、演習が1つも動きません。

やること

- 1 スタートメニューから VSCode を起動します。すでに開いている方はそのまま構いません
- 2 ブラウザで github.com を開き、会社のメールアドレスで新規アカウントを作成します。決めたユーザー名は手元にメモしてください
- 3 講師がこの場から Copilot の招待メールを送ります。メール内のリンクを開いて承認してください
- 4 VSCodeに戻り、左下のアカウントアイコンから GitHub にサインインします

できあがるもの

成果物のファイルはありません。GitHubユーザー名のメモと、VSCodeの左下にそのユーザー名が表示された状態が成果です。

セットアップの達成基準

この4つが揃ったら、手を挙げて教えてください。

達成の目安

- VSCodeの左下に、自分で作ったGitHubユーザー名が表示されている
- 右上のCopilotアイコンに斜線が入っていない
- GitHubユーザー名を手元にメモしてある（Day2でも使います）
- 配布ZIPをデスクトップに展開し、フォルダ名が 機電系_Day1_配布 のままになっている

つまづいたら

- 確認メールが届かない。迷惑メールフォルダを見て、それでも無ければ事務局へ連絡します
- サインインのブラウザが開いたまま戻らない。VSCodeを一度閉じて開き直すと、サインイン済みになっていることがあります
- アイコンに斜線が入ったまま。招待の承認がまだです。メールのリンクをもう一度開いてください
- そもそもVSCodeがどこにあるか分からない。スタートメニューで vs と打つと候補に出ます

時間が余ったら

- 終わった方は、配布ZIPの展開までここで済ませてください
- それも終わった方は、隣の方が1から4のどこで止まっているかを一緒に見てあげてください

アカウント作成の分岐

3つの分かれ道があります。当てはまる方は先に手を挙げてください。

起きること	どうするか
すでに個人のGitHubアカウントを持っている	招待は会社のメールアドレスに届きます。会社アドレスで新規に作るのが確実です
二要素認証の設定を求められた	認証アプリかSMSでの受信を選びます。会社端末で認証アプリが使えない方は手を挙げてください
確認メールが届かない	迷惑メールフォルダを確認し、それでも無ければ事務局に連絡します。5分待って来なければ次の対応に回ります

会社のメールアドレスで作ります。個人アドレスで作ると、招待が届きません。

サインインの確認点

この2か所だけ見れば、サインインできたか判定できます。

左下

アカウントアイコンの横に、自分のGitHubユーザー名が出ています

右上

Copilotアイコンに斜線が入っていません。斜線は未サインインか未承認の印です

2か所とも当てはまる方は手を挙げてください。1人ずつ画面を見て回るより速く終わります。

最初に打つ文

この2行のどちらかを、そのままチャット欄に打ってください。

日本語のまま打って構いません。返ってきた内容が正しいかどうかは、ここでは気にしません。

こんにちは。あなたは何ができますか。日本語で3行で教えてください。
設備の点検報告書を書くとき、あなたに手伝ってもらえることを3つ挙げてください。

打ち終わったら Enter を押します。文の途中で改行したいときは Shift+Enter です。

返事が返ってくることを確かめます。

どんな場面か

環境が壊れたまま午後に入ると、5つの演習が全部止まります。ここで疎通だけ確かめておけば、午後は中身に集中できます。

やること

- 1 右上のCopilotアイコンを押して、画面右側にチャット欄を開きます
- 2 前のページの文をそのまま入力して Enter を押します
- 3 数秒待って、パネルに日本語の文章が出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。チャット欄に日本語の返事が表示されていれば達成です。保存も不要です。

疎通確認の達成基準

日本語の返事が最後まで表示されたら達成です。

達成の目安

- チャット欄に日本語の文章が返ってきた
- 返事が途中で止まらず、最後まで表示された

つまずいたら

- パネルは開くが返事が来ない。社内ネットワークの制限の可能性があります。手を挙げてください、こちらで記録します
- 英語で返ってきた。文末に「日本語で教えてください」と足して、もう一度送ってください
- エラー表示が出る。サインインが切れています。左下のアカウントから入り直してください
- アイコンを押してもパネルが開かない。VSCodeを一度閉じて開き直します

時間が余ったら

- 返事が返った方は、続けて「もっと短く3行で」と頼んでみてください。会話が続くことを先に体験しておく、午後の演習が速くなります
- 配布ZIPを展開したフォルダの中にある 01_サンプル_点検記録.md を開いて、点検項目と困っていることの欄をざっと見ておいてください

直らないとき

今日中に直らなくても、翌週までに個別に対応します。

症状	確認すること
チャット欄が出ない	VSCodeを一度閉じて開き直します。それでも出なければ手を挙げてください
サインインできない	招待メールのリンクを承認済みか確認します。迷惑メールも見ます
返事が返ってこない	社内ネットワークの制限の可能性があります。この場では直せないなので記録します
画面が英語のまま	操作には支障ありません。日本語化は午後のVSCodeの章で扱います

ここで解決しなくても午前の後半と午後の座学は受けられます。焦らないでください。

仕組みの話へ

ここまでにしたこと

ここまでで、全員のPCで Copilotが動く状態を作り、チャット欄に日本語で話しかけて返事を受け取りました。



これからやること

ここからの60分は手を止めます。いま返事を返してきたものが何をしているのか、仕組み、得意と苦手、使う場面、国内での使われ方の順に押さえます。



そのあとどこで効くか

この線引きが、午後のリスクの話と、午後の5つの演習でどこまで任せるかの判断につながります。

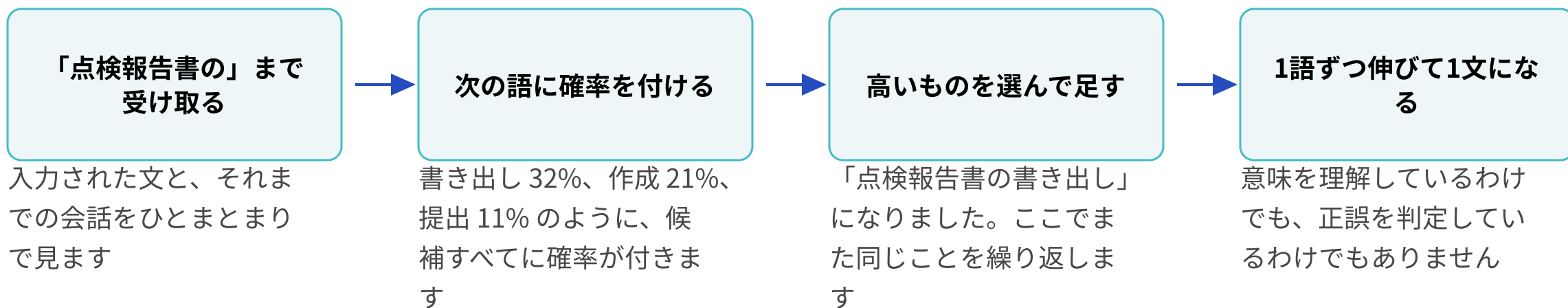
02

生成AIの基礎

何が得意で何が苦手か、任せる範囲の線引きを持って帰ります

生成AIの仕組み

生成AIは、次に来そうな言葉を**確率**で**選び**続けています。



確率の数値は説明のための例です。実際の値はモデルと文脈で変わります。

検索との違い

検索は探して返す、生成AIは作って返す。ここが違います。

検索エンジン

キーワードに合うページを探して、あるものをそのまま返します。出どころのURLが必ず残ります。同じ言葉で調べれば、だいたい同じ結果になります。

生成AI

学習した傾向から、その場で文章を作ります。同じ質問でも毎回違う文が返り、出どころは残りません。元のページを見せることはできません。

得意なこと・苦手なこと

形を整える仕事は任せられます。正しさを決める仕事は任せられません。

仕事の種類	任せ方	理由
報告書の下書き・要約	任せてよい	出てきた文を自分で読み直して直せます
用語の説明・書き方の相談	任せてよい	裏取りを自分でする前提なら、調べる時間が短くなります
体裁をそろえる作業	任せてよい	間違っても見ればわかり、その場で直せます
測定値の計算・件数の集計	そのままは使わない	数字の正しさを保証しません。桁を間違えても同じ口調で返します
規格の適合判定	そのままは使わない	規格番号そのものを作ってしまうことがあります
安全に関わる判断	使わない	誤りが事故につながり、後から直せません

判断の物差し

迷ったら、間違っても取り返しがつくかで決めます。

すぐ直せる

報告書の下書き。読み返して直せば、誰にも迷惑がかりません

確かめれば直せる

用語や手順の下調べ。一次情報で裏を取る前提なら使えます

ここから先は下書きまで

客先へ出す文書。責任を負うのは自分なので、最終形は自分で作ります

直すのに人が動く

測定値の計算。数字が独り歩きすると、後工程の判断まで巻き込みます

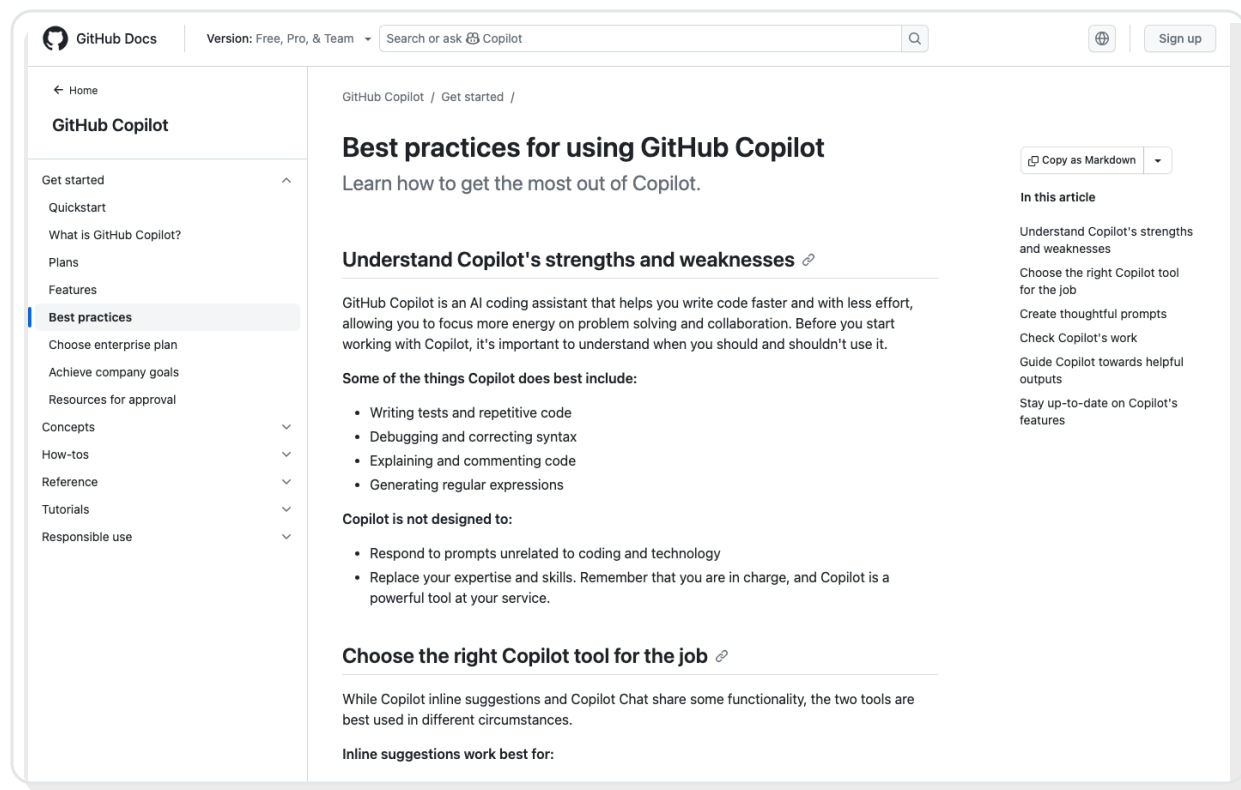
取り返しがつかない

安全に関わる判断。ここは使いません

覚えるのはこの1本の物差しだけです。表に無い場面もこれで判断できます。

提供元の説明

事実確認は利用者側の責任、という整理は提供元も同じです。



GitHub Copilot の利用上の注意に関する公式ドキュメント

国内での使われ方

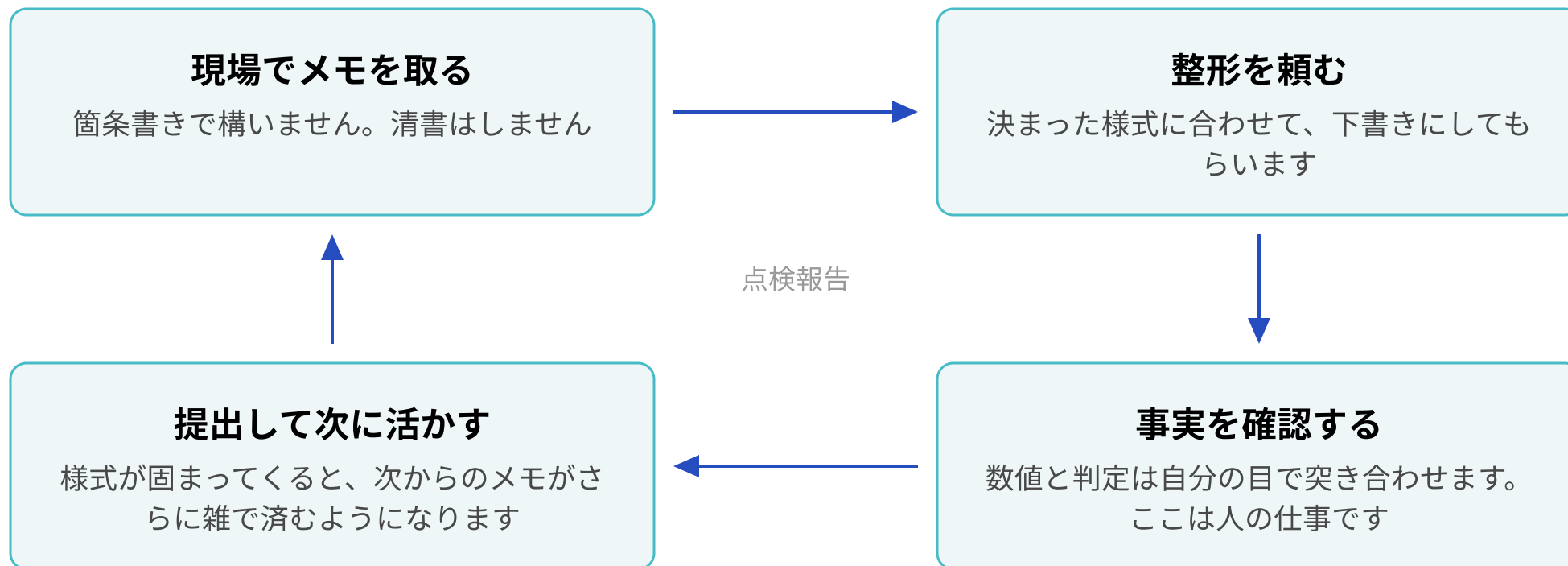
製造と保全の現場では、下書きと探し物から入るのが定番です。

工程	任せていること	人が残していること
設備の日常点検	現場メモから報告書の下書きを作らせる	数値と判定の確認、提出前の読み直し
設備マニュアルの参照	分厚い手順書から、該当しそうな箇所を探させる	手順書の原本での最終確認
不具合連絡の整理	バラバラの文面を、決まった項目に振り分けさせる	原因の切り分けと、対応の判断

業種と工程のレベルで書いています。個別の企業名と削減率は載せていません。

点検報告の回し方

現場メモを下書きに変える。ここが一番効きます。



メモに客先名や型番が入っていたら、貼る前に消します。午後の章で詳しく扱います。

探し物の変わり方

探す時間は短くなりますが、原本での確認は残ります。

これまで

分厚い設備マニュアルを、目次から当たりを付けて探します。目次に載っていない言い回しで書かれていると見つかりません。詳しい人に聞いて回ることもあります。

AIに聞く場合

この症状のときに見る章はどこですか、と日本語で聞きます。当たりを付ける時間が短くなり、詳しい人の手も止めません。

30秒、自分の業務に置き換えて考えてください。

**いま見た3つのうち
自分の仕事に一番近いのはどれですか**

30秒だけ考えてください。答えは集めません。午後の演習で、その場面を思い浮かべながら手を動かしてもらいます

Q. 生成AIの説明として、正しいものはどれですか。

- A インターネットを検索して、見つけた文章をそのまま返している
- B 質問の意味を理解したうえで、正解を計算して答えている
- C 次に来そうな言葉を確率で選び続けて、文章を作っている
- D あらかじめ用意された回答集から、近いものを探して返している

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。仕組みは確率による予測です。

C

次に来そうな言葉を確率で選び続けて、文章を作っている

次に来そうな言葉に確率を付けて、高いものを選んで1語ずつ足しています。意味の理解も正誤の判定もしていません。

A

インターネットを検索して、見つけた文章をそのまま返している

検索とは別の仕組みです。あるものを探すのではなく、その場で文を作ります。だから出どころのURLが残りません。

B

質問の意味を理解したうえで、正解を計算して答えている

正解を計算しているわけではありません。それらしい続きを選んでいるだけなので、間違いも同じ口調で返ってきます。

D

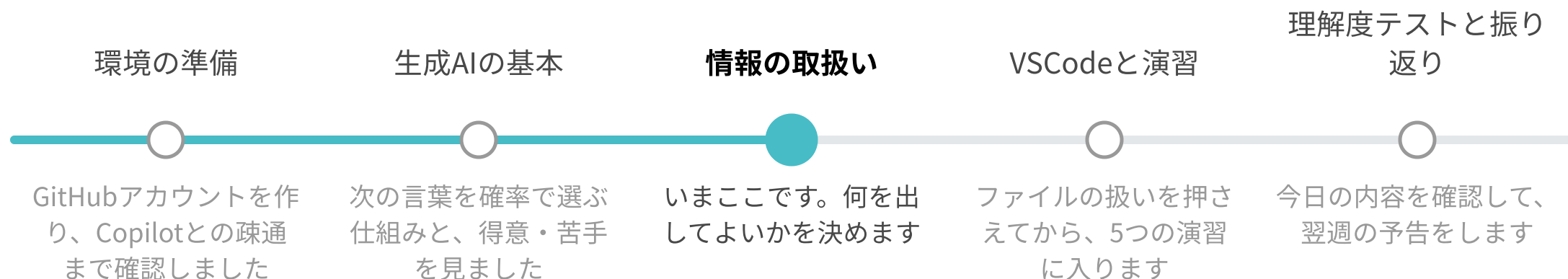
あらかじめ用意された回答集から、近いものを探して返している

決まった回答集は持っていません。同じ質問でも毎回違う文が返るのは、その場で作り直しているからです。

間違っている自信のある口調で返ってきます。内容の確認は人の仕事として残ります。

今日の現在地

午後の最初は、手を動かす前に**守る線を決める**時間です。



午後は演習が中心です。この章だけは手を止めて聞いてください。

基礎から実務の判断へ

苦手を知ることと、事故を避けることは別の話です。

ここまでにしたこと

午前の最後に、間違っているとしても自信のある口調で返ってくる、と一行だけ添えて締めました。あの一行がここからの出発点です。仕組みの話は午前で終わりで、次の言葉を確率で選んでいるという結論だけ持ってきてください。



これからやること

ここからは、苦手だと分かっているとしても事故になる場面を扱います。もっともらしい間違いの見分け方、入力した内容がどこへ行くか、客先情報の伏せ方4つ、判断に迷ったときの優先順位です。



そのあとどこで効くか

このあとの5つの演習は、ここで決めた線の内側だけを使って進めます。Day2の業務テーマ言語化も、この線の上で書きます。今日決めておけば、来週は迷わずに書き始められます。

03

リスクとリテラシー

客先の情報を扱う立場として、守る線を自分で引けるようになります

もっともらしい間違い

AIは間違っているときも、正しいときと同じ口調で答えます。

返ってきた回答をそのまま使う

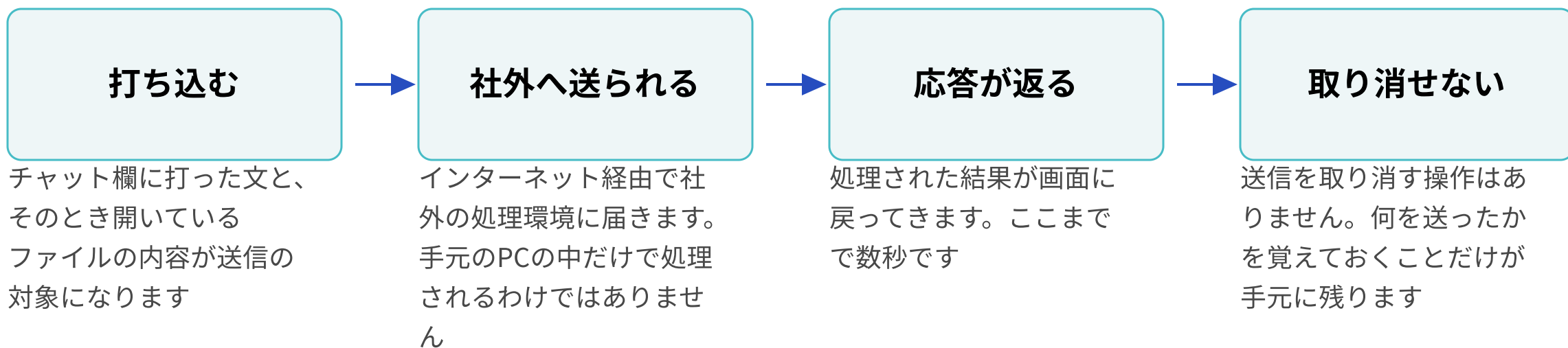
「その用途の締付トルクは JIS B 0000-0 に規定されています。表3の値をご確認ください」のように、規格番号も表番号も添えて返ってきます。番号が実在しないことがあります。文面からは判別できません。自信のなさそうな言い方はしてくれません。

番号と数値だけ引き直してから使う

文章の骨組みは使い、規格番号は日本産業標準調査会の検索など一次情報で引き直します。引けなければその回答は使いません。「どの規格に書いてあるか」ではなく「この番号は実在するか」を確かめます。

入力した内容の行き先

社外のサーバーで処理される点は、契約の内容にかかわらず変わりません。



学習に使われるかどうかは契約条件で決まります。今回のライセンスでの扱いは事務局に確認のうえお伝えします（要確認）。条件がどうであれ、客先の固有情報を出さない運用は変わりません。

送る前の確認

この4つを通したもののだけ送ります。迷ったら送れません。

1

客先が特定できないか

客先名・製品名・プロジェクト名・取引先名が入っていないかを見ます。入っていれば業界と工程のレベルに丸めます

2

実物が混じっていないか

測定データ、図面、社内の様式ファイル、画面のスクリーンショットは、そのままでは出しません。文章で構造を説明します

3

具体値が残っていないか

寸法・トルク・稼働時間などの数値は出しません。項目名だけを出すか、無関係な仮の値に置き換えます

4

自分で読み返したか

書いた文章を上から読み、社外に出ても困らないかを自分で判断します。判断がつかないものは送れません

業務の話を書くときは、**この4つ**を前提にしてください。

丸める

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

実物を出さない

実データ・実ファイル・スクリーンショットは使用しない

具体値を避ける

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

自分で確認する

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

この4つは、貴社との取り決めに基づく運用ルールです。今日以降の業務でも、この4つを前提にしてください。

同じ相談の書き換え

固有情報を落としても、相談の中身は落ちません。

そのまま書いた相談文

（客先名）向けの搬送装置（型番）で、駆動モーターの温度が82度まで上がります。添付の測定ログを見て、原因として考えられるものを教えてください。

丸めたあとの相談文

自動車部品の搬送装置で、駆動モーターの温度が想定より高い状態が続いています。設計と保全の両面から、確認すべき点を挙げてください。

そのまま使える相談文

丸めた言い方に、読み手と分量を足すだけです。

前のページの右側を、そのままチャット欄に打てる形にしたものです。1行目で対象、2行目で症状、3行目で頼みたいこと、4行目で読み手と分量、5行目で出せない情報を先に断っています。この5行の型は、業務のどの相談にも流用できます。

自動車部品の搬送装置について相談します。
駆動モーターの温度が、想定より高い状態が続いています。
設計と保全の両面から、確認すべき点を挙げてください。
読み手は同じ部署の技術者です。箇条書きで6項目以内、各項目40字以内をお願いします。
客先名と型番は出せないなので、一般的な範囲で教えてください。

5行目を入れておくと、AI側から「型番を教えてください」と聞き返される回数が減ります。

判断の優先順位

客先の規程が最優先です。4ルールはその下で使う書き方です。

1

派遣先の規程

客先の職場で働く場合、その職場の規程が最初に来ます。生成AIを使ってよいか、どの端末で使えるかは客先ごとに違います

2

会社の規程

次に、貴社の社内規程と申請手続きに従います。利用申請の可否と提出先は貴社の運用に従います（要確認）

3

今日の4ルール

上の2つに反しない範囲で、書き方の実務ルールとして4つを使います。4ルールが上の規程を上書きすることはありません

4

判断がつかないとき

送らずに、上長または貴社の相談窓口へ確認します。窓口の名称と連絡先は当日ご案内します（要確認）

客先の端末では使わない、客先から預かった資料は持ち出さない。この2つは、規程を確認するまでの既定の扱いにしてください。

判断が割れる3つを、先に自分で決めておきます。

どんな場面か

相談文を書くたびに「これは出していいのか」で手が止まります。判断が割れやすい3つを先に潰しておくと、このあとの演習とDay2で毎回悩まずに済みます。正解を当てる時間ではなく、自分の判断の癖を見つける時間です。

やること

- 1 手元のメモか配布資料の余白に、1・2・3と縦に番号を書いてください。
- 2 ケース1「客先名を伏せて、ある自動車部品メーカーの、と書いて相談する」を、○（出してよい）△（条件つき）×（出さない）のどれかで判定して書いてください。
- 3 ケース2「測定値をすべて10倍にして、桁を変えたデータを貼って相談する」を同じように判定してください。

判断が割れるケース（続き）

やること

- ① ケース3「社内の点検記録の様式を、中身を空欄にしてから貼って相談する」を同じように判定してください。
- ② 3つ書き終わったら、そう判断した理由を各1行だけ添えてください。4ルールの何番に当たるか、でも構いません。

できあがるもの

成果物の提出はありません。手元のメモに3つの判定と理由が書けていれば達成です。回収も発表もしません。

3ケースの判断

1は△、2は×、3は×。理由まで言えれば身についています。

達成の目安

- 3つとも○△×のどれかを自分で選び、理由を1行書けている
- ケース1を△と判断し、理由に「社名が推定できる書き方なら不可」の趣旨が入っている
- ケース2を×と判断し、理由に「値どうしの関係と傾向が残る」の趣旨が入っている
- ケース3を×と判断し、理由に「様式そのものが社外秘のことがある」の趣旨が入っている

つまずいたら

- 3つとも×にした場合。全部×にすると相談自体ができません。1は丸め方しだいで出せる、が答えです。どこまで丸めれば出せるかを考え直してください。
- ケース2を○にした場合。10倍でも値どうしの比率と変化の傾向はそのまま残ります。桁を変えても具体値を出したのと同じです。
- ケース1の△の条件が書けない場合。国内で数社しかない業種名を書くと社名が推定できます。自動車部品、産業機械のように、対象が十分に広い言い方かどうかで判断してください。
- 理由が思いつかない場合は、4ルールの何番に当たるかだけ番号で書いてください。それで達成です。

3ケースの判断（続き）

時間が余ったら

- 自分の担当業務で、判断が割れそうなケースを1つ書き出してください。Day2の業務テーマ言語化でそのまま使えます。
- ケース1を、社名が推定できない書き方に自分で直してみてください。業界名をどこまで広げれば安全かが見えます。

Q. 客先の装置について相談したいとき、適切な書き方はどれですか。

- A 客先名と型番を書き、測定ログのファイルを貼って相談する
- B 自動車部品の製造装置で温度が想定より高い場合、確認すべき点を相談する
- C 客先名は伏せたうえで、図面のスクリーンショットを貼って相談する
- D 相談内容を短くするため、実際の測定値をそのまま貼って相談する

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。固有情報を落としても、相談は成立します。

B

自動車部品の製造装置で温度が想定より高い場合、確認すべき点を相談する

業界と工程のレベルまで丸められており、客先名も実データも具体値も含まれていません。それでも装置の種類と症状は残っているので、確認すべき点は返ってきます。

A

客先名と型番を書き、測定ログを貼る

型番も測定ログも固有情報です。ルール1とルール2の両方に当たります

C

客先名を伏せて図面を貼る

図面そのものが機密です。名前を伏せても解決しません。構造は文章で説明します

D

実際の測定値をそのまま貼る

数値は具体値を避け、目的や課題の粒度で書きます。短くしたいなら項目名だけにします

迷ったときは「この文章が社外に出ても困らないか」を自分に聞いてください。困るなら書き直します。

判断の話から道具の話へ

ここから先は、画面の操作だけです。

ここまでにしたこと

もっともらしい間違いの見分け方、入力した内容の行き先、4ルール、判断の優先順位まで決めました。3つのケースも自分で判定しました。守る線はここで固定します。



これからやること

ここからは道具の話に戻ります。画面の名前は午前を押さえたので、残るのはフォルダとファイルの扱いだけです。開く、書く、名前を付けて保存する。この3つで足ります。



そのあとどこで効くか

このあとの5つの演習は、全部この画面の上で行います。最後の演習では、作ってもらった画面をファイルに保存してブラウザで開きます。Day2以降も同じ操作の繰り返しです。

04

VSCodeの使い方

フォルダを開き、ファイルを作って名前を付けて保存するまでを自分でできるようになります

フォルダとファイル

フォルダを開いておかないと、左に一覧が出ません。

ファイルだけを開いた場合

ファイル＞ファイルを開く で1つだけ開いた状態です。編集画面には中身が出ますが、左のエクスプローラーは空のままです。同じフォルダの他のファイルに移れず、保存するときも置き場所を毎回探すことになります。

フォルダを開いた場合

ファイル＞フォルダーを開く で 機電系_Day1_配布 を開いた状態です。左に中身が全部並び、クリックだけで行き来できます。保存先も最初からこのフォルダになるので、作ったファイルが迷子になりません。

ファイル操作の4つ

今日使うファイル操作は、この4つだけです。

1

フォルダを開く

上のメニューの ファイル から フォルダを開く を選び、デスクトップの 機電系_Day1_配布 を選びます。左のエクスプローラーに中身が並びます

2

ファイルを開く

エクスプローラーに並んだファイル名をクリックします。ダブルクリックは要りません

3

新しいファイルを作る

Ctrl+N を押すと空のタブが開きます。この時点では、名前も置き場所もまだ決まっていません

4

名前を付けて保存する

Ctrl+S を押し、ファイル名を拡張子まで入力して保存先を選びます。.md は文章、.html はブラウザで開ける画面になります

タブのファイル名の横に丸印が出ていたら、まだ保存されていません。Ctrl+S を押すと丸印が消えます。

覚えておく基本操作

文字が小さくて読みづらい方は、いま **Ctrl+Shift+=** を押してください。

やりたいこと	操作	使う場面
文字を大きくする	Ctrl+Shift+=	画面の文字が小さいとき。小さく戻すのは Ctrl+-
元に戻す	Ctrl+Z	貼り付けや削除を間違えたとき。続けて押せばさらに戻ります
やり直す	Ctrl+Y	戻しすぎたとき
ファイルの中を探す	Ctrl+F	開いているファイルから語を探すとき。右上に件数が出ます
画面を日本語にする	拡張機能から追加	左端の四角が4つ並んだアイコンから Japanese Language Pack を入れ、VSCodeを再起動します

日本語化の拡張機能は、社内から配布元のサイトに接続できる場合だけ追加できます。接続の可否は貴社の環境設定によります（要確認）。追加できなくても、今日の内容は英語表記のまま進められます。

演習で使う操作を、先に一度だけ通します。

どんな場面か

このあとの演習3では配布ファイルをCopilotに読ませ、演習5では作ってもらった画面をファイルに保存します。どちらもフォルダを開くところから始まります。本番でここが詰まると全員が止まるので、いま一度だけ通しておきます。

やること

- 1 上のメニューの ファイル から フォルダーを開く を選び、デスクトップに展開した 機電系_Day1_配布 フォルダを選んでください。
- 2 左のエクスプローラーに並んだ 01_サンプル_点検記録.md をクリックして開いてください。右側に中身が出ます。
- 3 Ctrl+N を押して新しいタブを開き、自分の名前と今日の日付を1行だけ打ってください。

ファイル操作の練習（続き）

やること

- 1 Ctrl+S を押し、ファイル名に メモ_午後.md と入力して、いま開いている 機電系_Day1_配布 フォルダの中に保存してください。
- 2 タブの丸印が消えたことと、左のエクスプローラーに メモ_午後.md が増えたことを確認してください。
- 3 最後に右上のCopilotアイコンを押してチャット欄を開き、もう一度同じアイコンを押して閉じてください。次の章の最初で使います。

できあがるもの

メモ_午後.md を、デスクトップの 機電系_Day1_配布 フォルダの中に保存します。中身は1行で構いません。提出はありません。

できていることの確認

左の一覧に **メモ_午後.md** が増え、丸印が消えていれば達成です。

達成の目安

- 左のエクスプローラーに並んだファイルの中に、01_サンプル_点検記録.md と メモ_午後.md の2つが見えている
- メモ_午後.md のタブに丸印が付いていない
- 01_サンプル_点検記録.md を開くと、点検項目の一覧が右側に読める
- 右上のCopilotアイコンで、チャット欄を開いて閉じる操作ができている

つまずいたら

- 配布フォルダが見つからない場合。ZIPを展開していない可能性があります。ダウンロードしたZIPを右クリックし、すべて展開を選んでからやり直してください。
- 保存先がドキュメントやダウンロードになってしまう場合。保存の画面の上部が 機電系_Day1_配布 になっているかを見てください。違っていれば、左の一覧から選び直します。
- ファイル名に .md を付け忘れた場合。エクスプローラーでそのファイルを右クリックし、名前の変更で .md を足してください。
- 左に一覧が出てこない場合。左端のいちばん上、書類が重なったアイコンを1回押してください。
- 保存の画面が英語で出る場合。Save as と出ていれば同じ画面です。ファイル名の欄に打って Save を押します。

できていることの確認（続き）

時間が余ったら

- メモ_午後.md に、午前の内容で覚えておきたいことを3行足して、もう一度 Ctrl+S で上書き保存してください。2回目の保存では名前を聞かれないことが確認できます。
- エディタのタブを×で閉じてから、エクスプローラーの メモ_午後.md をもう一度クリックして開き直してください。閉じてもファイルは消えないことが分かります。

操作の練習から実践へ

保存までできたので、ここからは**実際に頼んでみます**。

ここまでにしたこと

ここまでで、Copilotが使える状態を作り、生成AIが確率で言葉を選んでいることと、客先名や実データを出さない4つのルールを確認しました。直前ではフォルダを開いてメモ_午後.mdを保存するところまで、全員で通しました。



これからやること

ここからの100分は、その3か所を使って自分の手で5つの演習を回します。日本語で質問する、下書きを直してもらう、開いているファイルについて聞く、会話で寄せる、動く画面を作る。この5つです。



そのあとどこで効くか

5つ目でできた画面が、翌週のバイブコーディング入門と、3日目の業務改善ミニアプリの出発点になります。今日はここまで到達できれば十分です。

05

Copilotを触る

5つの小さな演習で、日本語で頼んで動かすまでを一通り体験します

Copilotの位置づけ

Copilotは別のアプリではなく、**書いている画面の右側**にいます。

ブラウザで使うチャット型AI

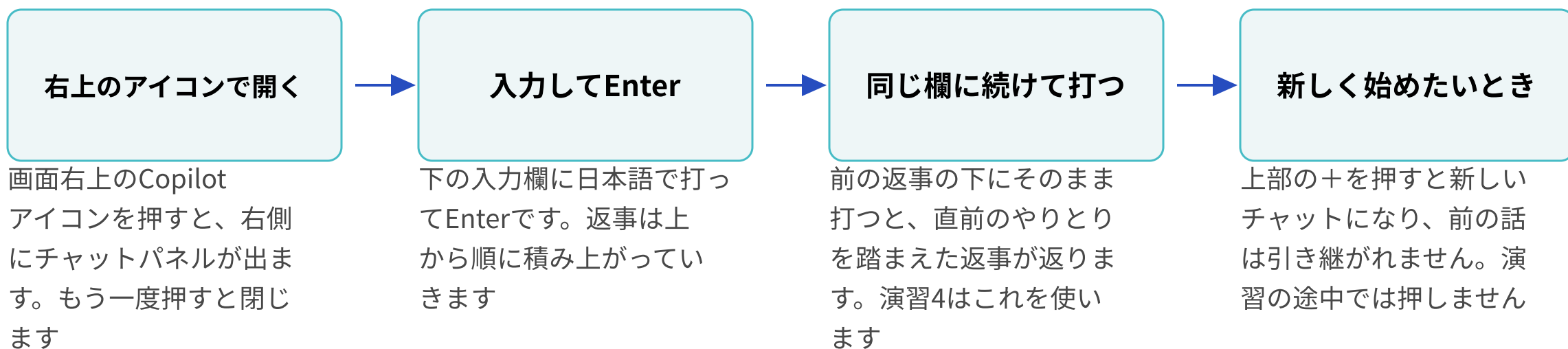
ブラウザのタブを開いて使います。渡せるのは自分がコピーして貼り付けた文字だけで、手元のファイルは見えていません。書きかけの資料に戻るにはタブを切り替えます。

VSCodeの中のGitHub Copilot

VSCodeの右側にチャット欄が開きます。いま開いているファイルの中身をそのまま材料にできます。書く場所と相談する場所が同じ画面にあるので、切り替えが発生しません。

チャット欄と会話の続け方

同じチャットの中で続けて打つと、前のやりとりが効きます。



アイコンが見つからないときは、上部メニューの表示からCopilot Chatを探してください。アイコンの位置とキーボードショートカットは当日のVSCodeの版で異なります（要確認）。

切り替えられますが、**今日は既定のままで進めます。**

どこで切り替えるか

チャット欄の入力ボックスの下にあるドロップダウンです。押すと選べるAIの名前が一覧で出ます。

選べる名前

貴社のライセンスで選択できるモデルの一覧は当日の画面でご確認ください（要確認）。名前は随時入れ替わります。

今日の演習での差

今日の5演習はどれを選んでも結果が大きく変わりません。触ってしまってもそのまま進めて支障ありません。

使い分けが要る場面

長いコードを扱うときや、手順の多い作業を任せるときです。翌週以降、必要になった段階で扱います。

頼み方の最低限

対象・読み手・形の3つを足すだけで、返ってくるものが変わります。

3つが抜けた頼み方

「点検報告書を書いて」。これだと、どの設備の、誰が読む、どのくらいの長さの報告書かわかりません。返ってくるのは教科書に載っているような一般的な書式の説明で、そのままでは自分の報告書に使えません。

3つを足した頼み方

「月次の設備点検報告書の書き出しを、上長向けに、200字以内で3パターン」。対象は月次の点検報告書の書き出し、読み手は上長、形は200字以内で3パターン。ここまで書けば、手直しして使える文章が返ってきます。

演習1で打つ文

考えずに済むよう、演習1で打つ文をそのまま出します。

次の1行をチャット欄にそのまま入力してEnterを押してください。自分の業務に置き換えられる方は、下の「対象」の部分だけ書き換えてください。客先名・製品名・プロジェクト名は入れません。

月次の設備点検報告書の書き出しを、上長向けに、200字以内で3パターン考えてください。

（書き換えるときの型）

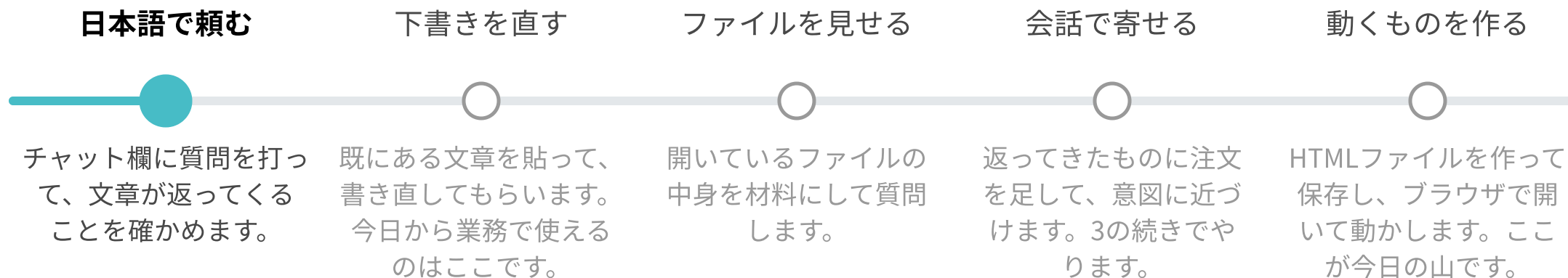
＜対象＞を、＜読み手＞向けに、＜形・字数・パターン数＞で作ってください。

（例：自分の業務に寄せる場合）

設備の異音に関する調査依頼の文面を、他部署の担当者向けに、150字以内で2パターン考えてください。

5つの演習の並び

できることを**1段ずつ積み上げて**、最後に動くものまで到達します。



1・2・3・5は単独で試せます。4だけは3の続きなので、3が動かなかった方は1の回答を起点にしてください。

現場で最初に使うのは、コードではなく文章の相談です。

どんな場面か

点検報告書や作業連絡の書き出しで手が止まる場面は、月に何度もあります。白紙から考える代わりに、たたき台を3つ出させて選ぶ。この使い方が、今日いちばん早く業務に持ち帰れます。

やること

- 1 右上のCopilotアイコンを押して、右側のチャット欄を開きます。
- 2 入力ボックスをクリックして、次の一文をそのまま入力します。「月次の設備点検報告書の書き出しを、上長向けに、200字以内で3パターン考えてください」
- 3 Enterを押して送信します。数秒から十数秒で文章が返ってきます。
- 4 返ってきた3つを読んで、手直しすれば使えるようなものが1つあるかを判断します。
- 5 気に入った回答があれば、その部分を選択してコピーし、Ctrl+N で新しいファイルを開いて貼り付け、Ctrl+S で メモ_演習1.md という名前で 機電系_Day1_配布 フォルダに保存します。

できあがるもの

メモ_演習1.md（機電系_Day1_配布 フォルダの直下）。保存まで進まなくても、3パターンが画面に出ているら達成です。

演習1の達成基準

3パターン返ってきて、1つは手直しで使えると**判断できたら**達成です。

達成の目安

- 書き出しの候補が3つ返ってきている。
- 3つのうち1つは、手直しすれば自分の報告書に使えると自分で言える。
- 客先名・製品名・プロジェクト名を1つも入れずに質問できている。

つまづいたら

- 一般論の説明しか返ってこない。対象・読み手・形の3条件を足して聞き直してください。
- 英語で返ってきた。続けて「日本語をお願いします」と1行送れば日本語に戻ります。
- Enterを押しても送信されず改行になる。入力欄の右にある送信ボタンを押してください（ボタンの有無は当日の画面で確認します。要確認）。
- 何も返らないまま止まる。左下のアカウントアイコンにGitHubユーザー名が出ているかを見てください。出ていなければサインインが切れています。

演習1の達成基準（続き）

時間が余ったら

- 同じ質問に「読み手は上長」「200字以内」「箇条書き禁止」の3条件を足して、出力がどう変わるかを見比べてください。
- 「もっと固い文体で」「もっとやわらかく」と2通り作らせて、文体の指定がどこまで効くかを確かめてください。

既にある文章を直す作業は、**今日から確実に効きます。**

どんな場面か

現場で書いた走り書きを、そのまま上長や他部署に出せる文体に整える。この手直しに毎回10分から20分かかっている人が多いはずです。ここを渡せるようになると、書く時間そのものが減ります。

やること

- 1 左のファイル一覧から 01_サンプル_点検記録.md を開きます。まだフォルダを開いていない方は、ファイル>フォルダーを開く からデスクトップの 機電系_Day1_配布 を選んでください。
- 2 ファイルのいちばん下にある「困っていること」の4行を選択して、Ctrl+C でコピーします。
- 3 チャット欄に貼り付けて、その下に次の一文を足します。「この4行を、上長への報告書向けの丁寧な文体に書き直してください。書かれている事実は落とさないでください」

演習2 下書きを直す（続き）

やること

- 1 Enterで送信します。返ってきた文章と元の4行を並べて、消えている項目がないかを目視で確かめます。
- 2 自分の文章で試したい方は、社外に出せる範囲のものに限って差し替えてください。客先名・製品名は入れません。

できあがるもの

メモ_演習2.md（機電系_Day1_配布 フォルダの直下）。保存しなくても、書き直された文章が画面に出ていれば達成です。

演習2の達成基準

元の事実が**1つも消えず**に、文体だけが変わっていれば達成です。

達成の目安

- 元の4行にあった内容が、書き直し後にも全部残っている。紙からExcelへの転記、記入漏れの発見の遅れ、過去の記録の検索、書き方のばらつきの4点です。
- 文体が報告書向けに変わっている。走り書きの調子が残っていない。
- 書き直し後の文章を、自分の判断で「これなら出せる」と言える。

つまづいたら

- 項目が1つ消えている。「元の4項目を全部残したまま、もう一度書き直してください」と送れば戻ります。
- 内容が勝手に膨らんで、元を書いていない対策案まで付いてくる。「書かれている内容だけで、付け足さないでください」と条件を足します。
- コピーがうまくいかない。ファイル上で4行をドラッグして選択し、選択部分が反転していることを確かめてから Ctrl+C を押してください。
- ファイルが開けない。ファイル>フォルダーを開く でデスクトップの 機電系_Day1_配布 を選ぶところからやり直します。

演習2の達成基準（続き）

時間が余ったら

- 同じ4行を「社内の口頭連絡向けに30字」と「客先提出の下書きとして丁寧に」の2通りに書き分けさせて、差を見てください。
- 残したい専門用語を2つ指定して、その語を使ったまま書き直させてください。用語の指定がどこまで効くかが分かります。

説明を全部打つ代わりに、**ファイルを開いておけば**読んでくれます。

どんな場面か

背景を全部チャットに打ち込むのは現実的ではありません。手元の資料を開いた状態にしておけば、それを材料にして答えが返ります。この差が、後の演習と実務での時短の正体です。

やること

- 1 VSCodeの上部メニューから **ファイル > フォルダを開く** を選びます。
- 2 デスクトップに展開した **機電系_Day1_配布** フォルダを選んで、**開く**を押します。フォルダ名は変えないでください。
- 3 左のファイル一覧に並んだ **01_サンプル_点検記録.md** をクリックして、編集画面に表示します。
- 4 チャット欄に「開いているこのファイルの内容を、5行で要約してください」と入力してEnterを押します。
- 5 返ってきた要約に、点検項目の4つと、転記にかかっている時間が入っているかを確認めます。

できあがるもの

メモ_演習3.md（**機電系_Day1_配布** フォルダの直下）。保存しなくても、要約が画面に出ていれば達成です。

演習3の達成基準

要約に**ファイル固有の中身**が入っていれば、読まれています。

達成の目安

- 要約の中に「点検項目が外観・異音・温度・振動の4つ」が入っている。
- 要約の中に「転記に2時間かかっている」という趣旨が入っている。
- 左のファイル一覧に、配布フォルダの中身が並んで見えている。

つまずいたら

- 点検の一般論しか返ってこない。ファイルが読まれていません。編集画面のタブが 01_サンプル_点検記録.md になっているかを確認してから、もう一度送ってください。
- 左のファイル一覧に何も出ない。フォルダを開かずにファイルだけ開いています。ファイル>フォルダーを開く からやり直します。
- デスクトップに配布フォルダが無い。ZIPが展開されていません。ZIPを右クリックして すべて展開 を選び、デスクトップに出してください。
- フォルダ名の末尾に (1) が付いている。二重に展開されています。中の階層をたどって、01_サンプル_点検記録.md が直下にあるフォルダを開いてください。

演習3の達成基準（続き）

時間が余ったら

- 「このファイルの困っていることのうち、AIで解決できそうなものはどれですか」と続けて聞いてみてください。
- 「この点検記録をExcelに転記するとしたら、どんな列構成にしますか」と聞いて、表の形まで提案させてください。

一度で完成させようとしなが、いちばんの近道です。

どんな場面か

最初に返ってきたものが的外れだったとき、頼み方を最初から書き直す人が多いのですが、続けて注文を足すほうが速く近づきます。この進め方は、演習5でも翌週のバイブコーディングでもそのまま使います。

やること

- 1 演習3の要約が画面に残っている状態から始めます。チャットを閉じてしまった方は、演習3をもう一度実行してください。
- 2 続けて「もっと短く、3行にしてください」と送ります。
- 3 返ってきたら、さらに「箇条書きにして、困っていることだけに絞ってください」と送ります。
- 4 3回目に「この中で、いちばん先に手を付けるべきものはどれですか。理由も1行で」と送ります。
- 5 演習3がうまくいかなかった方は、演習1で返ってきた3パターンの文章を起点にして、同じように注文を足してください。

できあがるもの

メモ_演習4.md（機電系_Day1_配布 フォルダの直下）。保存しなくても、3回のやりとりが画面に残っていれば達成です。

演習4の達成基準

3回以上やりとりして、**最初より意図に近づいていれば達成です。**

達成の目安

- 同じ話題について3回以上のやりとりが画面に残っている。
- 3回目の回答が、1回目より自分の意図に近いと自分で言える。
- 3回のうち、どの一言がいちばん出力を変えたかを1つ挙げられる。

つまずいたら

- 注文を足しても内容が変わらない。指示が抽象的すぎます。「短く」ではなく「3行で」のように数字を入れてください。
- 前のやりとりを覚えていない様子で、毎回ゼロから答えてくる。新しいチャットを開いていないか確認してください。同じチャットの中で続ける必要があります。
- 演習3が動かず起点が無い。演習1で返ってきた3パターンの文章をチャットに貼り直して、そこから注文を足してください。
- 回答がどんどん長くなる。「今の回答を半分の長さにしてください」と送れば戻せます。

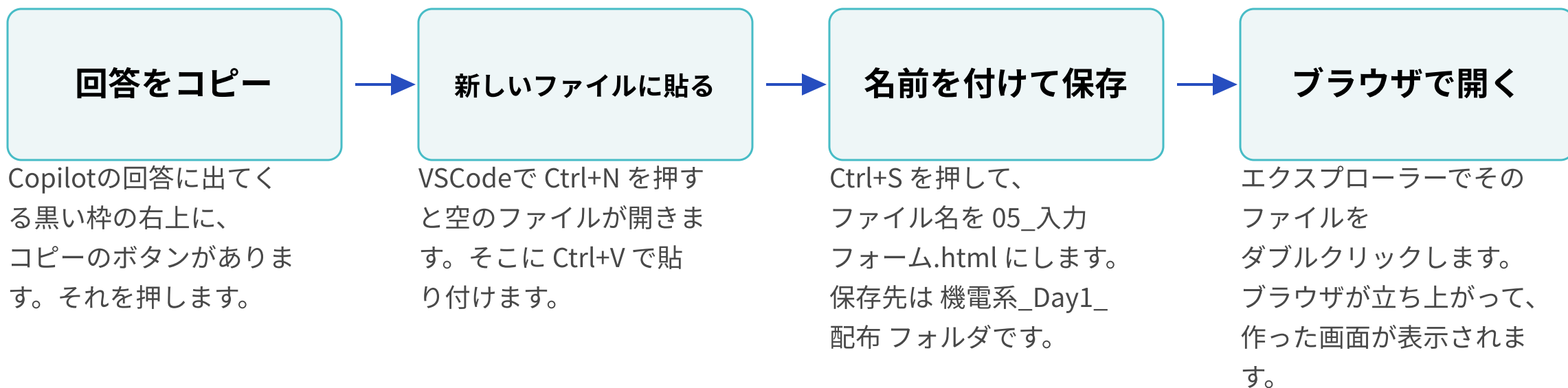
演習4の達成基準（続き）

時間が余ったら

- わざと「もっと専門的に、規格番号も添えて」と頼んで、内容が怪しくなるところまで押してみてください。午前に扱った、もっともらしい間違いが実際に出ます。
- 同じ要約を「新人向け」と「役員向け」に書き分けさせて、読み手の指定が文章のどこを変えるかを見てください。

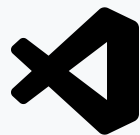
演習5の操作の流れ

回答をコピーして、**ファイルとして保存し**、ブラウザで開きます。



コピーのボタンの見た目と位置は、VSCodeとCopilotの版で変わります（要確認）。ダブルクリックでブラウザが開かない場合は、右クリック > プログラムから開く > Microsoft Edge を選んでください。

保存するのはVSCode、**開いて動かすのはブラウザ**です。



VSCode

回答を貼り付けて 05_入力
フォーム.html として保存す
る場所



Edge

保存したファイルを
ダブルクリックして動かす場
所

日本語で頼んだものが、ブラウザで動くところまで行きます。

どんな場面か

点検結果を紙に書いてExcelに転記する、という手間の入口には、たいてい入力画面がありません。名前と数量を入れて一覧に足すだけの画面を、日本語で頼んで作ります。3日目の業務改善ミニアプリは、この延長線上にあります。

やること

- 1 チャット欄に次の一文を入力してEnterを押します。「名前と数量を入力して追加ボタンを押すと、下の一覧に1行増えるHTMLファイルを作ってください。1つのファイルで完結させてください」
- 2 回答の中に、黒い枠に囲まれたコードが返ってきます。枠の右上のコピーボタンを押します。
- 3 VSCodeで Ctrl+N を押して新しいファイルを開き、Ctrl+V で貼り付けます。

演習5 動くものを作る（続き）

やること

- 1 Ctrl+S を押して保存します。ファイル名は 05_入力フォーム.html、保存先はデスクトップの 機電系_Day1_配布 フォルダです。.html まで含めて入力してください。
- 2 Windowsのエクスプローラーで 機電系_Day1_配布 フォルダを開き、05_入力フォーム.html をダブルクリックします。
- 3 ブラウザに表示された画面に、名前と数量を入れて追加ボタンを押します。下の一覧に1行増えれば完成です。

できあがるもの

05_入力フォーム.html（デスクトップの 機電系_Day1_配布 フォルダ直下）。ブラウザで開いて、入力した行が一覧に増えれば達成です。

演習5の達成基準

ブラウザで値を入れて、一覧が1行増えれば達成です。

達成の目安

- デスクトップの 機電系_Day1_配布 フォルダに 05_入力フォーム.html がある。
- ダブルクリックするとブラウザが立ち上がり、名前と数量の入力欄と追加ボタンが表示される。
- 値を入れて追加を押すと、下の一覧に1行増える。
- ページを再読み込みすると入れた行が消えるのは正常です。保存する仕組みを入れていないためです。

つまづいたら

- ダブルクリックしても別のアプリで開く。ファイル名の末尾が .html になっていません。VSCodeで Ctrl+S し直して、名前を 05_入力フォーム.html にしてください。
- ブラウザは開いたが真っ白。貼り付けたコードが途中で切れています。回答の枠を最初から最後までコピーし直してください。
- 追加ボタンを押しても一覧が増えない。「追加ボタンを押しても一覧に増えません。1つのファイルで動くように直してください」とそのまま伝えて作り直させます。
- コードの枠が複数返ってきた。「1つのファイルにまとめて、そのまま保存できる形で出してください」と送り直します。
- 保存先が分からなくなった。VSCodeのタブを右クリックしてパスを確認するか、Ctrl+S でもう一度 機電系_Day1_配布 を選び直して保存します。

演習5の達成基準（続き）

時間が余ったら

- 動いたら「数量が空のときは追加できないようにしてください」と1つだけ注文を足して、同じファイルを上書き保存し、ブラウザを再読み込みしてください。
- 一覧の各行に削除ボタンを付けてください。動かなくても翌週以降に影響しません。
- 一覧の下に合計数量を表示させてください。

この進め方の呼び名

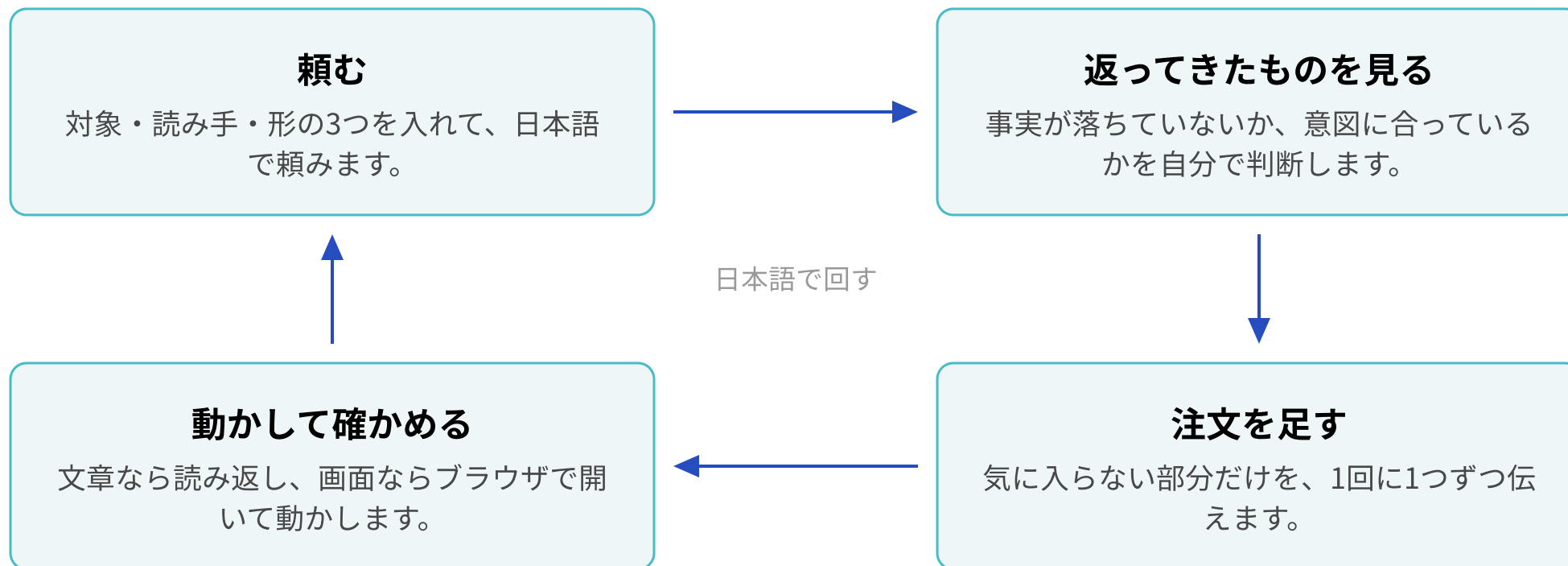
いまやったことには名前が付いています。

コードを書かずに 日本語で作らせる

この進め方をバンプコーディングと呼びます。演習5でやったことがそれです。翌週のDay2は、この進め方を1日かけて練習します。

5演習で身についたこと

この4手を回すのが、明日からの使い方です。



明日の業務で、この4手を回せそうな作業を1つだけ思い浮かべてください。書き出す必要はありません。



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

演習から確認へ

手を動かした内容を、**今日の成果物**として一度確認します。

ここまでにしたこと

5つの演習を回して、日本語で頼む、下書きを直す、ファイルを見せる、会話で寄せる、動くものを作る、の5つを一通り体験しました。午前には生成AIの仕組みと、客先情報の扱い方も確認しています。

これからやること

ここから40分で理解度テストを行います。今日扱った生成AIの基礎と、情報の取り扱いが範囲です。選択式で、今日の資料を見ながら解いて構いません。

そのあとどこで効くか

このテストの結果が、今日1日の成果物として記録に残ります。覚えているかではなく、必要なときに資料に戻って調べられるかを見ています。

06

理解度テスト

今日扱った内容を、資料を見ながら確認する復習型のテストです

理解度テストの概要

全員が満点を取れることを目指した、復習用のテストです。

項目	内容
出題範囲	今日扱った生成AIの基礎と、情報の取り扱いです。演習の操作手順は出ません。
設問数と形式	15問の選択式です。今日の資料を見ながら解いて構いません。
時間	30分です。早く終わった方も、その場でお待ちください。
相談	受講者どうしの相談はご遠慮ください。資料に戻るのは自由です。
結果	採点は弊社で行います。返却方法は貴社の運用に合わせます。

分からないところは資料に戻って確認してください。覚えているかではなく、調べて分かるかを見ています。

資料を見ながら、30分で15問に答えてください。

1

受け取る

理解度テストを受け取ってください。配布の形式は当日ご案内します。

2

解く

30分で15問に答えます。今日の資料に戻って確認しながら解いて構いません。

3

困ったら

設問の意味が分からない場合は手を挙げてください。答えは言えませんが、設問の読み方は説明します。

4

終わったら

早く終わった方も退出せず、その場でお待ちください。全員が終わってから次へ進みます。

5

提出する

終了の合図で提出してください。提出方法は当日ご案内します。

確認から締めへ

残り40分で、今日を振り返って翌週につなぎます。

ここまでにしたこと

今日は環境を整え、生成AIの仕組みと情報の扱い方を確認し、Copilotで5つの演習を回して、最後に理解度テストで内容を確認しました。動くものを作るところまで到達しています。



これからやること

ここからの40分は、自分の言葉で今日を整理する時間です。3つの問いに答える形で、各自メモに書き出してもらいます。書いたものは提出しません。



そのあとどこで効くか

書き出した内容が、翌週の業務テーマ言語化の材料になります。翌週の冒頭には復習の時間を40分取ってあるので、今日詰まったところは持ち越して構いません。

07

振り返りと次回

今日を自分の言葉で整理して、翌週に持っていくものを決めます

3つの問いに、自分の言葉でメモを書いてください。

どんな場面か

研修の内容は、聞いた直後がいちばん薄れやすい状態です。自分の言葉に置き換えたものだけが1週間後に残ります。ここで書いたメモが、翌週の業務テーマ言語化の出発点になります。

やること

- ① 手元のメモ用紙か、VSCodeで Ctrl+N を押して開いた新しいファイルを用意します。
- ② 1つ目の問いに答えます。「今日いちばん意外だったことは何ですか」。1行で構いません。
- ③ 2つ目の問いに答えます。「明日の業務で1つだけ試すなら、どの作業に使いますか」。作業名だけで構いません。客先名や製品名は書かないでください。
- ④ 3つ目の問いに答えます。「まだ不安なこと、分からないままのことは何ですか」。これは翌週の材料にします。
- ⑤ 書き終わったら、3つ目に書いた不安なことだけ、言える範囲で口頭で共有してもらいます。挙手制です。

できあがるもの

手元のメモ、またはVSCodeで作ったファイル。提出はしません。持ち帰って翌週に持参してください。

振り返りの達成基準

3つとも**1行ずつ書けていれば十分です。**

達成の目安

- 3つの問いに、それぞれ1行以上書けている。
- 2つ目の作業名に、客先名・製品名・プロジェクト名が入っていない。
- 3つ目に書いた不安が、翌週に聞ける形の質問になっている。

つまずいたら

- 1つ目が思い浮かばない。今日の演習で「へえ」と思った瞬間を1つだけ思い出してください。感想で構いません。
- 2つ目が大きすぎる。部署全体の課題ではなく、自分が明日やる1つの作業まで絞ってください。
- 3つ目に何も書けない。分からないことが無いのではなく、まだ言葉になっていないだけです。今日の中で「もう一度説明してほしい」ところを1つ挙げてください。
- 書く場所が無い。VSCodeで Ctrl+N を押して、そこに書いてください。保存は任意です。

振り返りの達成基準（続き）

時間が余ったら

- 2つ目に書いた作業について、Copilotに「この作業をAIに手伝ってもらおうとしたら、どこから頼めばよいですか」と聞いてみてください。
- 今日作った 05_入力フォーム.html を、自分の業務のどの入力作業に置き換えられるかを1行書き足してください。

今日のゴールの確認

朝に立てた4つのゴールを、**判定できる形で見直します。**

環境が動いた

左下にGitHubユーザー名が出ていて、チャット欄に日本語を打つと返事が返る。
この状態のまま翌週も使います。

仕組みが分かった

確率で次の言葉を選んでいるから、間違いも自信のある口調で返ってくる。午前のクイズに答えられれば達成です。

守る線が決まった

客先名・製品名・プロジェクト名は業界と工程まで丸める。実データと実ファイルは出さない。迷ったら送らない。

触った実感がある

演習5で作った 05_入力フォーム.html が、ブラウザで動いた。動かなかった方は翌週の冒頭で扱います。

今日の位置と残り2日

今日は3日間の**1日目**です。作るのは翌週からです。

Day1 今日

環境を整え、生成AIの仕組みと情報の扱いを確認し、Copilotで5つの演習を回しました。

Day2 翌週

バインドコーディング入門とプロンプトの基礎。崩れた表をAIに整えてもらう演習も入ります。

Day3 その次

ダミーデータの分析と、業務改善ミニアプリの作成。見本から1つ選んで、時間内に動くところまで作ります。

Day間はおおよそ1週間空きます。翌週の冒頭には復習と環境の再確認に40分取ってあります。

翌週の内容

翌週は、実際に手を動かして作る回です。

翌週やること	内容
Day1の復習と環境確認	1週間空くので、環境が動く状態に戻るところから始めます。今日詰まったところもここで扱います。
バイブコーディング入門	演習5でやった進め方を、130分かけて練習します。作るものは業務のものでもゲームでも構いません。
プロンプトの基礎	今日の対象・読み手・形の3つに、条件と例の足し方を加えます。
データの下処理	空白や結合セルで崩れた表を、AIに整えてもらいます。現場にありがちな形の表を使います。
業務テーマの言語化	自分の業務でAIが使えるような場面を書き出します。提出も発表ありません。

翌週までのお願い

宿題は1つだけです。書き出す必要はありません。

面倒だなと思う作業を 1つ思い浮かべておく

ご自身の業務で、毎回手間だと感じている作業を1つ。翌週のバイブコーディング入門で、それを題材にします。書き出さなくても、頭の片隅にあれば大丈夫です。

今日の振り返りで2つ目の問いに書いた作業が、そのまま使えます。メモを持ってきてください。