

2026.08.21 FRI

事前セットアップ

今回は Claude Code を使ってアプリを作ります。第1回・第2回にご参加いただいた方は、動作確認だけで終わります。第3回から参加される方は、Obsidian と Claude Code の2つを入れていただきます。前日までに終わらなかった場合も、当日その場で対応します。

[このページのPDF →](#)[教材ページに戻る →](#)

CONTENTS

1. どちらに当てはまるか
2. A 第1回・第2回にご参加の方
3. B 第3回から参加される方
4. うまくいかないとき
5. 当日お持ちいただくもの

どちらに当てはまるか

今回は Claude Code を使ってアプリを作ります。第1回（7月31日）と第2回（8月3日）で入れていただいた環境をそのまま使いますので、ご参加いただいた方は動作確認だけで終わります。第3回から参加される方は、Obsidian と Claude Code の2つを入れていただく必要があります。

項目	第1回・第2回にご参加の方	第3回から参加される方
Obsidian	入っています。保管庫もそのまま使います	インストールと保管庫の作成が必要です

Claude Code	入っています。起動するだけ確認します	インストールが必要です
ログイン	切れていないかを確認します	お送りするご案内に沿ってログインしてください
今回の配布フォルダ	両方とも、 前日までに このページの A-2 からダウンロードして展開しておいてください。アドインの部品の取り寄せ（手順4）に使います。当日は展開したフォルダをそのまま使います	
道具（Skill）	両方とも、配布フォルダの中の <code>SETUP-Mac.command</code> または <code>SETUP-Windows.bat</code> が自動で置きます	
所要の目安	[10min]	[30min]

前日までに終わらなかった場合も、当日その場で対応します。会場に講師とアシスタントがおりますので、うまくいかなかった画面のままお越しください。途中で止まったところのスクリーンショットがあると、早く済みます。当日お渡しできる USB メモリもご用意しています。

A 第1回・第2回にご参加の方

ほぼそのまま大丈夫です。ただし前回から3週間空いていますので、その間に環境が変わっていることがあります。次の4つだけ確認してください。

A-1 4つの確認

1 Claude Code が起動するか

Mac は「ターミナル」（Command + Space で `ターミナル` と入力）、Windows は「PowerShell」（Windows キーで `PowerShell` と入力）を開いて、次を入力し Enter を押してください。

```
claude --version
```

2.1.211 (Claude Code) のように、数字とカッコ書きが返ってくれば入っています。数字の細かい違いは気にしないで構いません。 `command not found` や `認識されません` と出た場合は、下の「うまくいかないとき」をご覧ください。

2 設定に壊れているところがないか

```
claude doctor
```

インストールの状態と設定ファイルを読み取り専用で点検して、問題があれば直し方まで表示します。何も起動しないので、気軽に打って構いません。

3 ログインが活着ているか

ターミナルまたは PowerShell で `claude` と入力して Enter を押すと、対話が始まります。「こんにちは」とだけ打って、返事が来ればログインは活着ています。終わるときは `/exit` と入力してください。

ログインを求められた場合は、画面の案内どおりにブラウザで承認してください。Claude Code は Pro・Max・Team・Enterprise のいずれかのプランが必要で、無料プランでは動きません（公式ドキュメント）。第1回でご案内したアカウントでログインしていただいた方は、そのまま使えます。

4 Obsidian の保管庫が開くか

Obsidian を起動して、前回作っていただいた保管庫が開くことを確認してください。午後の演習③で、過去の資料を保管庫にどう置いておくべきかを考えます。そのとき手元にあると比べやすくなります。保管庫が見つからない場合も、演習は配布フォルダの中だけで進みますので支障ありません。

A-2 今回だけ、先に入れていただくもの

この回の最後は、PowerPoint の右側に出るパネル（アドイン）を、ご自身の指示で作るところまで進みます。ここだけは当日その場で環境を作ると間に合いません。前日までに4つ済ませてください。所要は15分ほどです。

先に1つだけ確認してくださいPowerPoint が、ブラウザ版ではなく**アプリとして PC に入っていること**。アドインはアプリ版の右側に出ます。ブラウザ版しかない場合は当日その旨をお伝えください。見学の形でお席を用意します。

やること	やり方	確認
1 PowerPoint を確かめる	PowerPoint を開き、Mac は「PowerPoint」メニューの「PowerPoint について」、Windows は「ファイル」→「アカウント」を開きます	Mac は 15.24 以降、Windows は Microsoft 365 であれば動きます

2 Node.jsを入れる	nodejs.org を開き、緑色の「Node.js®を入手」を押して、指示どおりに進めます	ターミナル（Windows は PowerShell）で <code>node -v</code> と打ちます。v22 のような数字が出れば完了です
3 証明書を入れる	同じ画面で <code>npx office-addin-dev-certs install</code> と打ちます。パスワードを求められたら、PC のログインパスワードを入れてください	最後に installed と出れば完了です
4 部品を先に取り寄せる	下のボタンから配布フォルダをダウンロードして展開し、05_アドイン演習/9_9_完成品/ の中で <code>npm install</code> と打ちます	<code>node_modules</code> というフォルダができれば完了です

配布フォルダ（Windows の方） ↓

配布フォルダ（Mac の方） ↓

展開したフォルダは書類フォルダ（~/Documents）に置いてください。当日はこのフォルダをそのまま使います。落とし直すと、ここで取り寄せた部品が無い状態に戻ります。

The screenshot shows the Node.js website. At the top, there's a navigation bar with links like '学ぶ' (Learn), 'はじめに' (Getting Started), 'ダウンロード' (Download), 'ブログ' (Blog), 'ドキュメント' (Documentation), 'ベータ版ドキュメント' (Beta Documentation), '貢献' (Contribute), and 'コース' (Courses). Below this, the main heading is 'どこでもJavaScriptを使おう' (Wherever you use JavaScript). The text below the heading says: 'Node.js®はクロスプラットフォームに対応したフリーでオープンソースのJavaScript実行環境です。開発者にサーバー、ウェブアプリ、コマンドラインツール、スクリプトなどを開発する環境を提供します。' (Node.js® is a free, open-source JavaScript runtime environment that supports cross-platform. It provides an environment for developers to build servers, web apps, command-line tools, scripts, etc.). There is a prominent green button that says 'Node.js®を入手' (Get Node.js). Below the button, it says 'セキュリティサポート EOLバージョン用' (Security support for EOL versions). To the right, there's a code editor showing a simple HTTP server example in JavaScript.

手順2で開く画面 左下の緑のボタンです。押した先で、お使いの OS に合ったファイルが選ばれます。バージョンの選択肢が出たら、新しいほうで構いません。

4 が一番大事です。ここで取り寄せるファイルは 100MB 以上あり、当日会場で全員が同時に取りに行くと回線がふさがります。前日までをお願いします。

ターミナルの開き方

OS	開き方	フォルダへの移動
Mac	Spotlight（Command + Space）で「ターミナル」と打って Enter	<code>cd</code> と打って半角スペースを1つ空け、99_完成品 フォルダをウィンドウに

ドラッグして Enter		
Windows	エクスプローラーで 99_完成品 を開き、アドレス欄に powershell と打って Enter	上の方法なら、開いた時点でそのフォルダにいます

動くところまで見ておきたい方へ

ここは任意です。当日は全員で一緒に行いますので、飛ばしていただいても構いません。先に1回通しておく、当日の詰まりがなくなります。

PowerPoint をすべて閉じてから、 99_完成品 フォルダで次を打ちます。

```
npm start
```

PowerPoint がひとりでに開き、右側に「スライド下書き」というパネルが出れば成功です。終わるときは同じ画面で npm stop と打ちます。

ちなみに当日は、このコマンドを直接は打ちません。Claude Code に「アドインを起動してください」と頼む形で進めます。ここで試すのはその下ごしらえで、1回通っていれば当日は必ず動きます。

こうなったら	Mac	Windows
パネルが出てこない	Finder で Command + Shift + G を押し、 ~/Library/Containers/com.microsoft.Powerpoint/Data/Documents/wef を開きます。このフォルダが無ければ作り、 manifest.xml をコピーしてから PowerPoint を開き直します	PowerPoint の「ホーム」→「アドイン」→「詳細」に出ていないか見てください。出ていなければ PowerPoint をすべて閉じて npm start をやり直します
「安全ではない」と出る	手順3の証明書が入っていません。 npx office-addin-dev-certs install をもう一度打ってください	
PowerPoint が開かない	PowerPoint が起動したままだと差し替わりません。すべて閉じてから打ち直してください	

うまくいかない場合はそのままにしておいてください。当日の午前中に前に対応します。演習④は午後の最後なので、時間はあります。

A-3 画面のあるものを作るための確認

第1回・第2回は文章を扱う研修でしたが、今回は画面のあるものを作ります。使うものが少し変わりますので、当日つまづかないように先に触っておいてください。

確認すること	やり方	なぜ必要か
ブラウザで HTML が開くか	下の「見本アプリを開いてみる」から1本開いてみてください。 ボタンを押して動けば大丈夫です	午前に作るヒアリングシートは HTML ファイル1本です。 ダブルクリックでブラウザが開く状態になっている必要があります
スクリーンショットが撮れるか	Mac は Shift + Command + 4、 Windows は Windows + Shift + S。 範囲を選んで撮ってみてください	直したい画面を撮って Claude Code に渡します。言葉で説明するより速く、正確に伝わります
電源と通信	電源アダプタをお持ちください。 会場の Wi-Fi をお使いいただけます	Claude Code が長く動く時間帯があります。その間に画面が消えると、途中で止まります
ディスクの空き	2GB ほど空けておいてください	作ったファイルと、やり直したときの控えが溜まります

見本アプリを開いてみる →

A-4 1つだけ考えてきていただきたいこと

直近で担当された案件を1つ、思い浮かべておいてください。午前はその案件のヒアリングを題材にします。実名は使いませんので、業界と論点だけで構いません。

あわせて、その案件で「聞いておけばよかった」と思ったことがあれば、それも覚えておいてください。作るヒアリングツールの項目になります。

第2回でスライド生成キットをお使いいただいた方は、そのときの生成物があれば見返しておいてください。今回はその手前の工程を作ります。

B 第3回から参加される方

入れていただくのは Obsidian と Claude Code の2つです。順に進めて30分ほどです。うまくいかない箇所があっても、当日その場で対応しますので、途中で止まっても問題ありません。

2つが済んだら、上の A-2（今回だけ、先に入れていただくもの）もお願いします。午後の最後の演習で使います。

B-1 画面つきの手順書

第1回でお配りした環境構築手順です。1画面ずつ図が入っているので、こちらを開きながら進めていただくのが確実です。

PRE-SETUP

環境構築手順

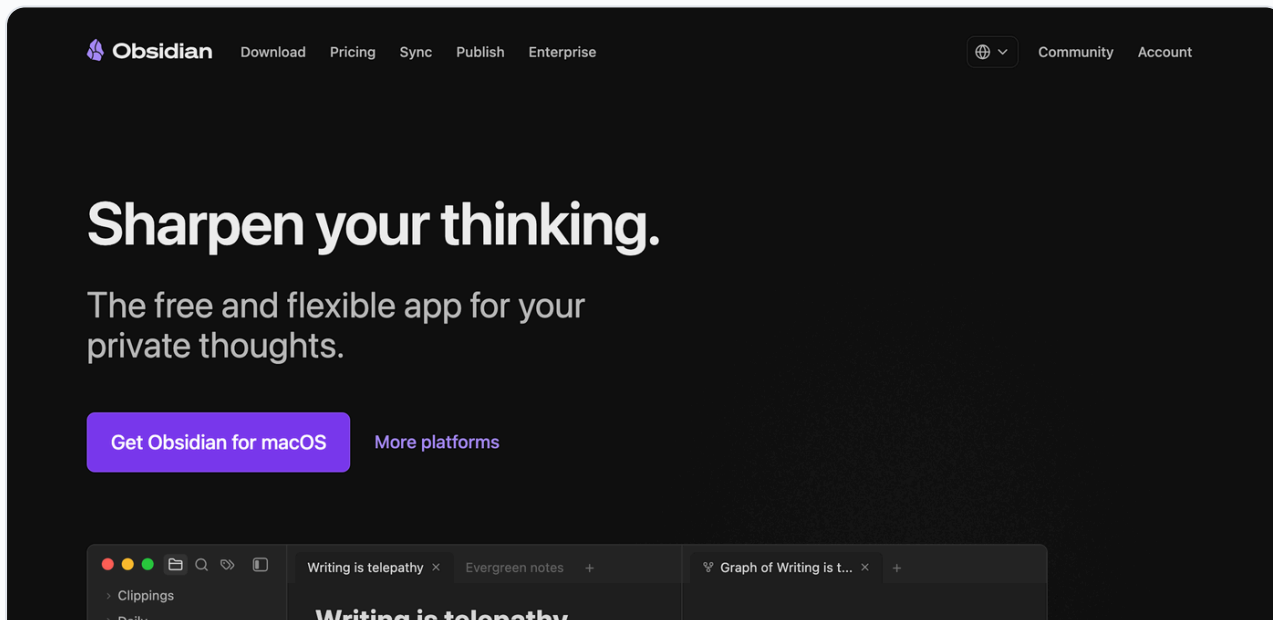
Obsidian のインストール、Claude Code のインストール、両者の連携までを画面つきで追えます。巻末に「よくあるエラーと対処法」とショートカット集が付いています。

Windows の方 →

Mac の方 →

この手順書は第1回・第2回向けに作ったものです。今回に必要なのは **Obsidian** と **Claude Code** の2つだけです。「Pythonパッケージ管理ツール」の節は、今回は飛ばしてください。Node.js は今回使いますが、手順書の節ではなく上の A-2 の方法で入れてください。

B-2 Obsidian



obsidian.md のトップページ 紫色のボタンから取得します。お使いの OS に合ったものが自動で選ばれます。

obsidian.md からインストーラーを取得して、画面の案内どおりに進めてください。起動したら「新規保管庫を作成」を選び、書類フォルダ（`~/Documents`）の下に任意の名前で作ります。保管庫が開けば完了です。

Macで「開発元を検証できません」と出た場合は、アプリのアイコンを右クリックして「開く」を選んでください。

B-3 Claude Code (Mac)

「ターミナル」を開きます。Command + Space を押して `ターミナル` と入力すると出てきます。開いた画面に次を入力して Enter を押してください。

```
curl -fsSL https://claude.ai/install.sh | bash
```

Mac のログインパスワードを求められた場合は入力して Enter を押します。入力しても画面には何も表示されませんが、それで正常です。 `Claude Code successfully installed!` と出れば完了です。

B-4 Claude Code (Windows)

Windows キーを押して `PowerShell` と入力し、Windows PowerShell を開きます。管理者として実行する必要はありません。開いた画面に次を入力して Enter を押してください。

```
irm https://claude.ai/install.ps1 | iex
```

`Claude Code successfully installed!` と出れば完了です。

画面の左端が `PS C:\` で始まっているれば PowerShell です。 `PS` が付かず `C:\` だけの場合はコマンドプロンプトなので、上のコマンドは動きません。PowerShell を開き直してください。

B-5 入ったことの確認

同じ画面で続けて次を入力してください。

```
claude --version
```

`2.1.211 (Claude Code)` のように数字が返れば入っています。何も返らない、あるいはエラーが出る場合は、下の「うまくいかないとき」をご覧ください。

B-6 ログイン

Claude Code は Pro ・ Max ・ Team ・ Enterprise のいずれかのプランが必要です。無料の Claude.ai プランでは使えません（公式ドキュメント）。

アカウントは Giverty で発行し、事前にメールでご案内します。届いたメールのリンクから承認してください。アカウントの選択画面が出た場合は「両方のアカウントを保持する」を選び、

そのまま進めます。チャット画面に切り替われば完了です。

ご案内が届いていない場合は、当日その場でお渡しできますので、そのままお越しください。

承認が済んだら、ターミナルまたは PowerShell で `claude` と入力して Enter を押します。ブラウザが開いたら案内どおりに承認してください。対話が始まったら「こんにちは」と打って返事が来れば、準備は終わりです。終わるときは `/exit` と入力します。

B-7 当日の朝にすること

ここから先は当日、全員一緒に進めます。事前にやっていただく必要はありません。

1. A-2 で展開した配布フォルダを開く（まだの方はこのページからダウンロードして展開する）
2. 展開したフォルダの中の `SETUP-Mac.command`（Windows の方は `SETUP-Windows.bat`）をダブルクリックする
3. 環境の確認と道具の配置が自動で進み、ハンズオン資料がブラウザで開く

うまくいかないとき

最初に `claude doctor` を試してください。インストールと設定を点検して、問題があれば直し方まで表示します。それでも分からない場合は、下の表から探してください。

出ている表示	何が起きているか	どうするか
<code>command not found: claude</code> (Mac)	入ってはいるが、場所が知られていない	ターミナルを一度閉じて開き直してください。それでも駄目なら <code>echo 'export PATH="\$HOME/.local/bin:\$PATH"' >> ~/.zshrc</code> を実行し、続けて <code>source ~/.zshrc</code> を実行します
<code>'claude'</code> は、内部コマンドまたは外部コマンド...として認識されていません (Windows)	同じく、場所が知られていない	PowerShell を一度閉じて開き直してください。 インストール先は <code>%USERPROFILE%\.local\bin</code> です
<code>'irm'</code> is not recognized as an internal or external command	PowerShell ではなく コマンドプロンプトを開いている	Windows キーから PowerShell を検索して開き直してください

The token '&&' is not a valid statement separator	逆に、 コマンドプロンプト向けの書き方を PowerShell に 入れている	このページの PowerShell 用の コマンドをお使いください
ログイン画面から先に進めない	無料プランのアカウントで ログインしている	ご案内したアカウントで ログインしてください。ご案内が 届いていない場合は当日 お渡しします
ダウンロードしたファイルの名前が 読めない文字になる	ZIP の文字コードが OS と 合っていない	もう一方の ZIP（Windows 用と Mac 用があります）で 落とし直してください

これ以外のエラーは、画面に出ている文字をそのままコピーして、お使いの生成AIに貼り付けて聞いていただくのが早いです。送ったコマンド、途中で出た表示、最後のエラーの3つを揃えて貼ると、たいてい1往復で解決します。それでも解決しない場合は、当日お持ちください。

当日お持ちいただくもの

- ノートPC（第1回・第2回と同じもの）
- 電源アダプタ
- 直近1か月で「毎回これをやっているな」と思った作業を1つ
- 第2回で書いた AI活用ワークシート（お持ちの方のみ）

入れないもの

実データを入れないでください。クライアント名、案件名、未公開の数字、個人情報。この4つは、AIに渡す文章にもファイルにも書きません。演習の題材は配布フォルダの中にダミーで用意してあります。

判断に迷ったときは、入れない側が正解です。

