

序章

情報処理は AI でもやれる簡単な仕事になる

2026.05.01

序章 — 事務処理はOffice、 業務ソフトはJava/C#、しか しAIはPythonとテキスト

事務処理はOffice。業務ソフトはJavaやC#。しかしAIは、Pythonとテキストでできている。

ここに、決定的な断絶がある。

道具を変える

OpenAIもAnthropicもPythonで動いている。SDKもPython。データはMarkdown、JSON、YAML。これは偶然ではなく、AIの構造そのものから来ている。

WordファイルもExcelシートもPDFも、AIに渡すにはテキストへの変換が要る。変換するたびに、書式が崩れ、構造が消える。JavaやC#のレガシーコードは、AIに読ませても助言の質が下がる。

AIネイティブな道具と、企業の標準的な道具のあいだに、決定的な断絶が走っている。

すぐ消す

Office、Java、C#は、すぐに消してしまう。

これらの道具で行っていた作業——書式を整える、表を作る、画面を作る、コードを書く——は、AIに任せればいい簡単な仕事になった。

AIでもやれる仕事のために、人間が重い道具を抱え込む。それが、いま多くの職場で起きていることだ。

道具を変えれば、思考が変わる。思考が変われば、人間にしかできない仕事に時間を使えるようになる。

全員に関わる

これは技術者だけの話ではない。

事務職のあなたへ。WordをMarkdownに、ExcelをCSVに変える。それだけでAIが直接相談相手になる。

営業のあなたへ。報告書を書式から構造に変える。AIが分析と提案を返してくれる。

現場のあなたへ。手順書をテキストで残す。AIが多言語化し、新人教育を支える。

個人事業主のあなたへ。請求書も契約書もブログもMarkdownで持つ。Claudeが事実上の従業員になる。

開発者のあなたへ。新しく作るものはPython。WebサイトはHTMLとCSSと必要最小限のJavaScript。Reactはいらない。

Pythonは全員のもの

「Pythonは技術者のもの」という偏見を捨てる。

Excelの変換、メールからの抽出、PDFの整理、ファイル形式の統一。これらは事務職や個人事業主の日常で頻発する作業だ。

Pythonなら数行で終わる。そして書く必要はない。**Claudeに日本語で頼めばコードが返ってくる**。実行するだけ。

書く能力ではなく、使う能力。これが新しいリテラシーである。

最小スタック

職種を問わない。



全部テキスト。AIがそのまま読み書きできる。十年後も読める。

道具は、思考をかたちづくる

Wordで書くと、書式に気を取られる。Markdownで書くと、構造が前になる。

Excelで考えると、表に収まる発想ばかりになる。JSONで持つと、関係が明示される。

Javaで設計すると、クラス階層を先に作りたくなる。Pythonで書くと、やりたいことを最短で書ける。

Reactで作ると、ビルド設定とバンドルサイズに悩む。HTMLで書くと、内容そのものに集中できる。

道具を変えることは、思考を変えることだ。

実例: 数字で見る

Word ファイル(50 KB、5,000 文字)を Claude に渡すと、約 8,000 トークン消費する。同じ内容を Markdown にすると 4,000 トークン。**ほぼ半減**。AI 利用料も同じ比率で下がる。

100 個の Word から「肥料」を含む段落を抽出する作業: VBA で 30 分かけてコードを書く。同じデータが Markdown なら、`grep -A 3 肥料 *.md` の 1 行で 0.1 秒。

Excel `.xlsx` 1.2 MB のファイル(10,000 行の売上データ)、CSV にすると 280 KB。**4分の1**。Claude に渡す時、転送も解析も速くなる。

Office、Java、C# を使い続けることは、毎日 AI 利用料を 2 倍以上払い続けることでもある。

実例: 生み出せるもの

Markdown 1 ファイルから、**印刷品質の PDF・美しい Web ページ・プレゼンスライド・EPUB 電子書籍・AI への入力**が同時に生成できる。同じ原稿が、全媒体に展開する。

`pandoc + xelatex` を使えば、Markdown から **書籍出版品質の PDF** が作れる。表紙、目次、ヘッダー、ページ番号、参考文献、図表番号—— 学術論文や商業出版の体裁が、コマンド 1 行で生成される。

過去 10 年の社内文書を Markdown 化すれば、Claude が **組織の意思決定パターン** を分析できる。「過去 5 年で同じ議論を何度繰り返したか」「どの方針が定着し、どれが消えたか」が定量化できる。**組織の集合知が、検索可能な財産になる。**

Python とテキストの組み合わせは、節約のためだけにあるのではない。**個人や小組織が、これまで大企業や専門家チームにしかできなかった仕事を生み出せるようにする道具立てだ。**

結びに

事務処理は Office、業務ソフトは Java や C#、しかし AI は Python とテキスト。

AI 時代に何が変わるか。**情報の処理は、AI でもやれる簡単な仕事になる。**

書式を整える、表を作る、メールを書く、コードを書く、報告書をまとめる——これらは、AI に任せればいい仕事になった。人間がやる必要はない。

Office、Java、C# は、人間が情報処理を担っていた時代の道具である。AI でもやれる仕事を、人間がわざわざ重い道具で抱え込む。これが、いま多くの職場で起きていることだ。

人間の側に残る仕事は、**何をするか、なぜするか、結果をどう判断するか**を決めることだけだ。そこに集中するために、情報処理は AI に渡す。そのための道具が、Python とテキストである。

すぐに消す。それだけで、AIが同僚になる。

次の章から、領域ごとに具体的な作法を見ていく。

関連記事

- [構造分析08: 企業ITの税を引く](#)
- [構造分析12: AIと個人事業](#)
- [それでも Windows と Office を使い続けますか?](#)
- [Claudeと一緒に学ぶDebian](#)