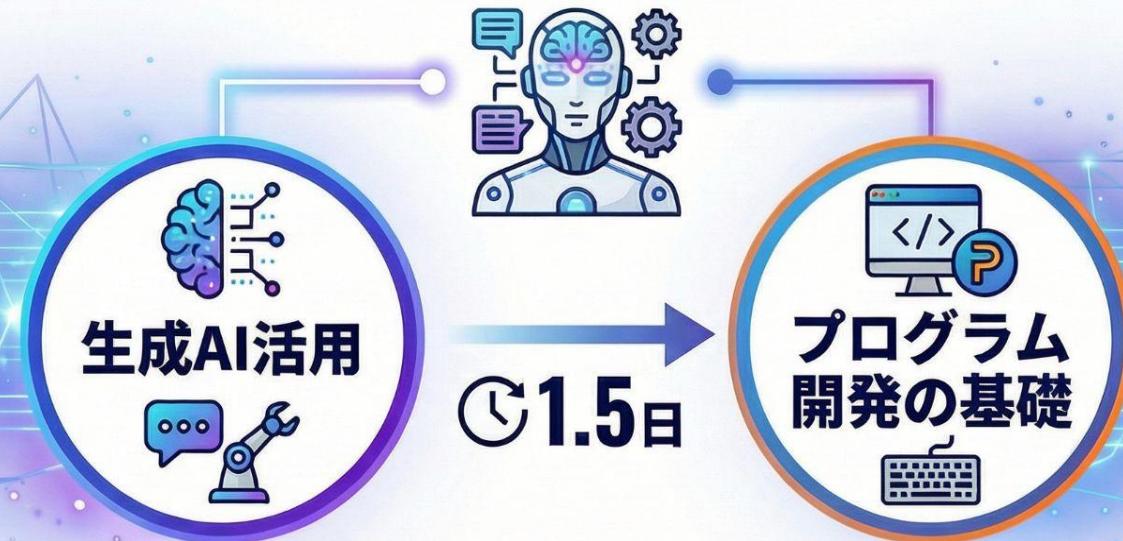


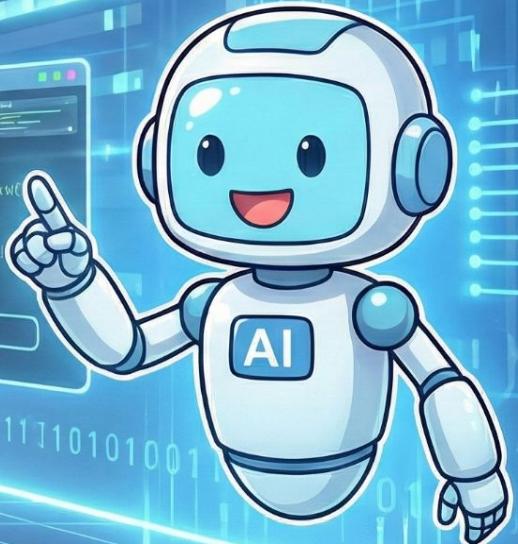
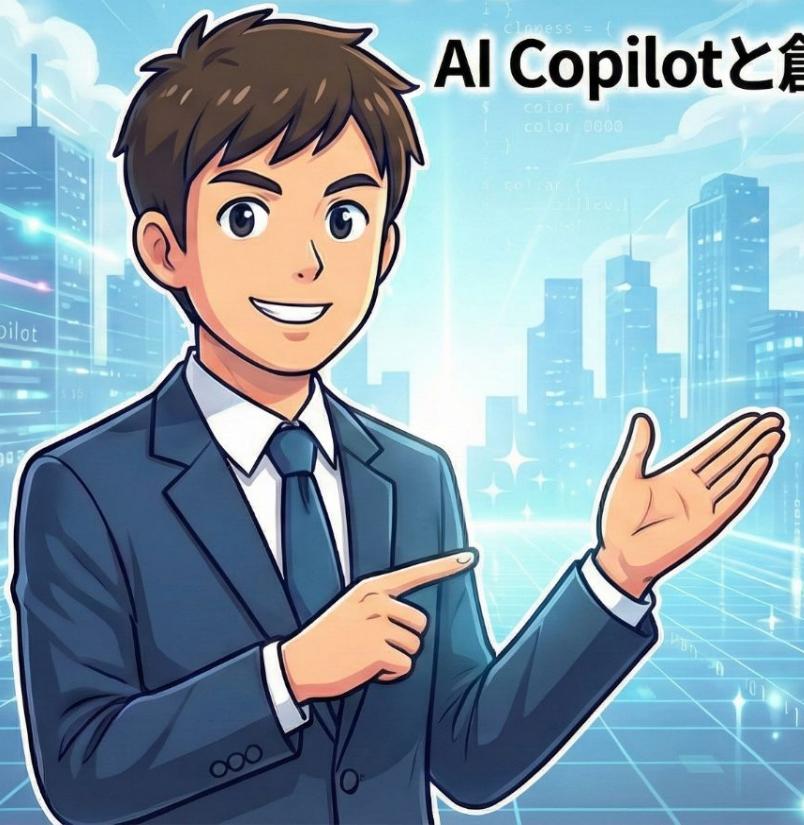
# 未来の技術者必修! 生成AI活用とプログラム開発の基礎を1.5日で完全習得



ナビゲーター：  
株式会社原子力エンジニアリング 解析サービス本部  
システム技術グループ 主幹技師長 翼 雅洋, 博士(工学)

# 生成AI駆動開發入門

# AI Copilotと創る、新しい開発体験

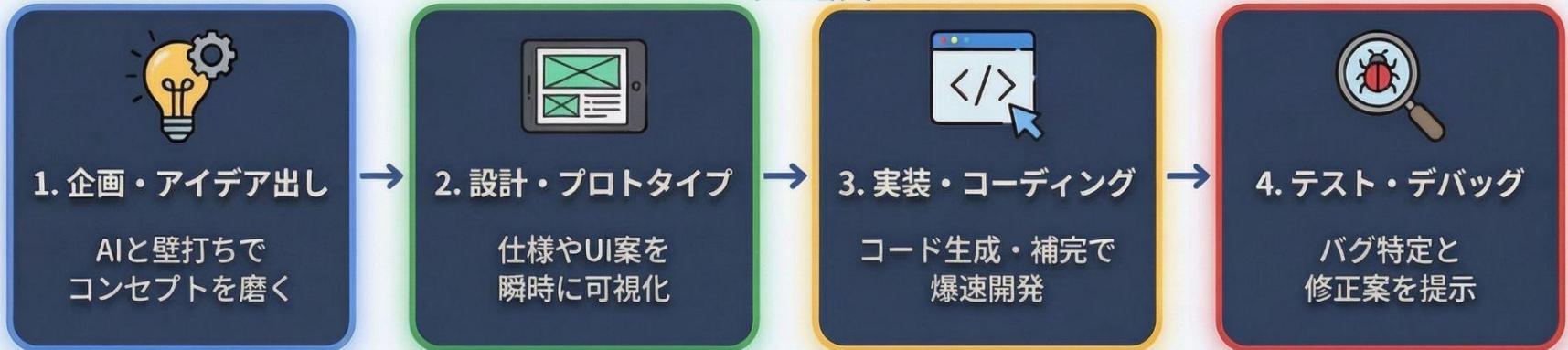


```
1 package com.wu;
2
3 class Segment {
4     tree
5     cones = &cones();
6     contle("Sounoneat! seeWc
7 },
8 muttete() {
9     // ignorance dawn slrat!
10    euPdatinttsrmenoe();
11 };
12 }
13 }
```



# 生成AI駆動開発とは？

開発プロセス全体をAIが加速させる新しいアプローチ



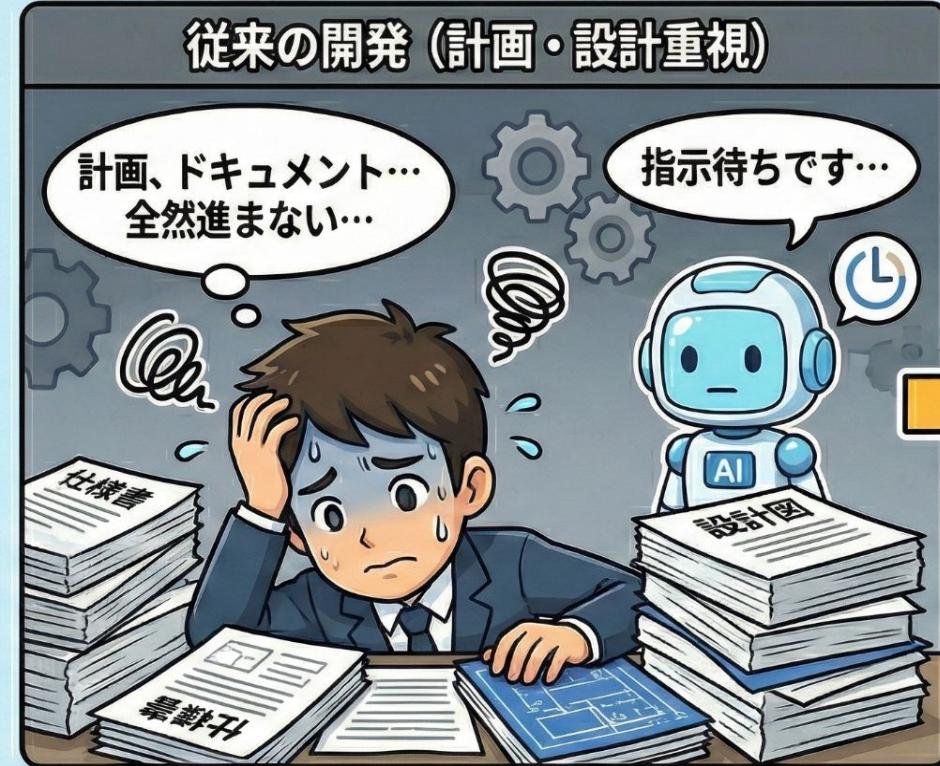
開発の全工程をAIが強力にサポート。  
人間は「創造」に集中できる新しい時代へ。

# バイブコーディングとは? (Vibe Coding) (直感とフローで創る新しい開発スタイル)

## 従来の開発 (計画・設計重視)

計画、ドキュメント…  
全然進まない…

指示待ちです…



## バイブコーディング (直感・試行錯誤重視)

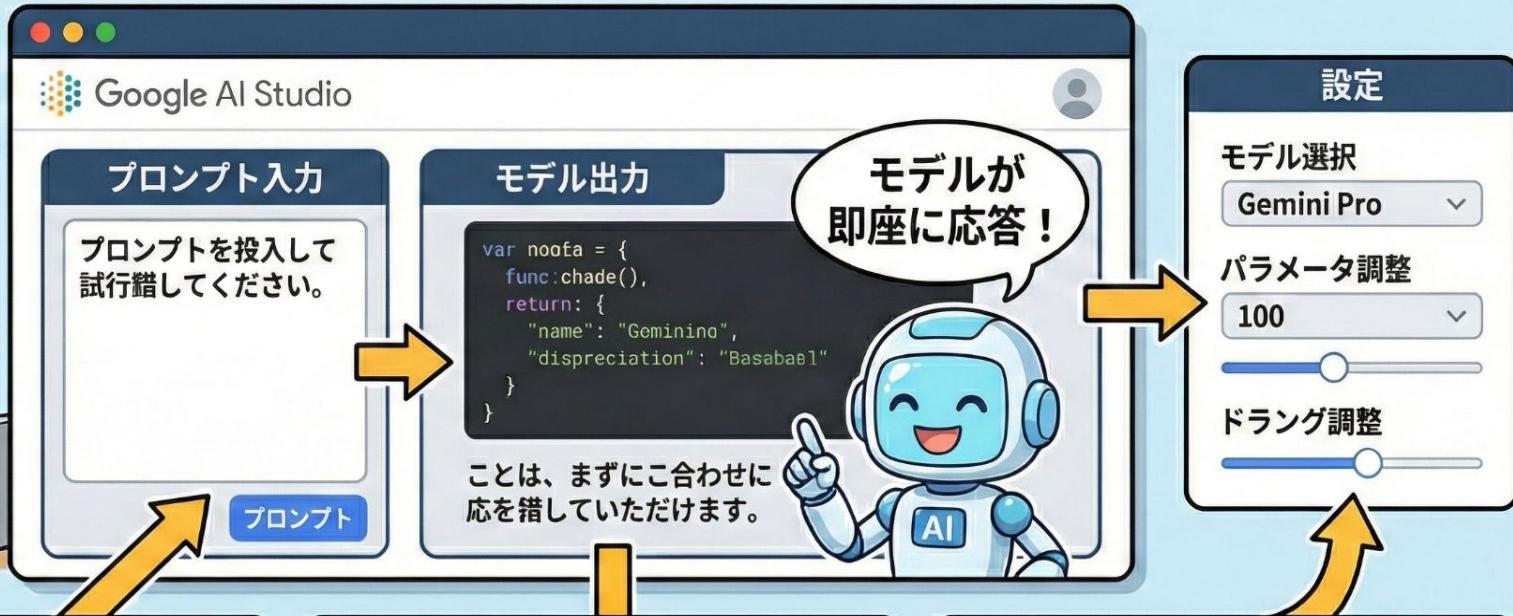
この『バイブス(直感)』  
を形にして！

了解！即座に  
コード化します！  
ジャムセッションですね！

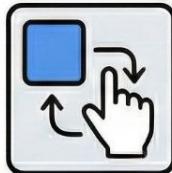


まとめ：計画よりも『試行錯誤』と『没入感（フロー）』を重視。  
AIがあなたの直感を即座に形にする、共創型の開発スタイル！

# Google AI Studio: 誰でも簡単にAI開発!



## ① 直感的なUI



ドラッグ&  
ドロップ

## ② 多彩なモデル



Gemini



PaLM

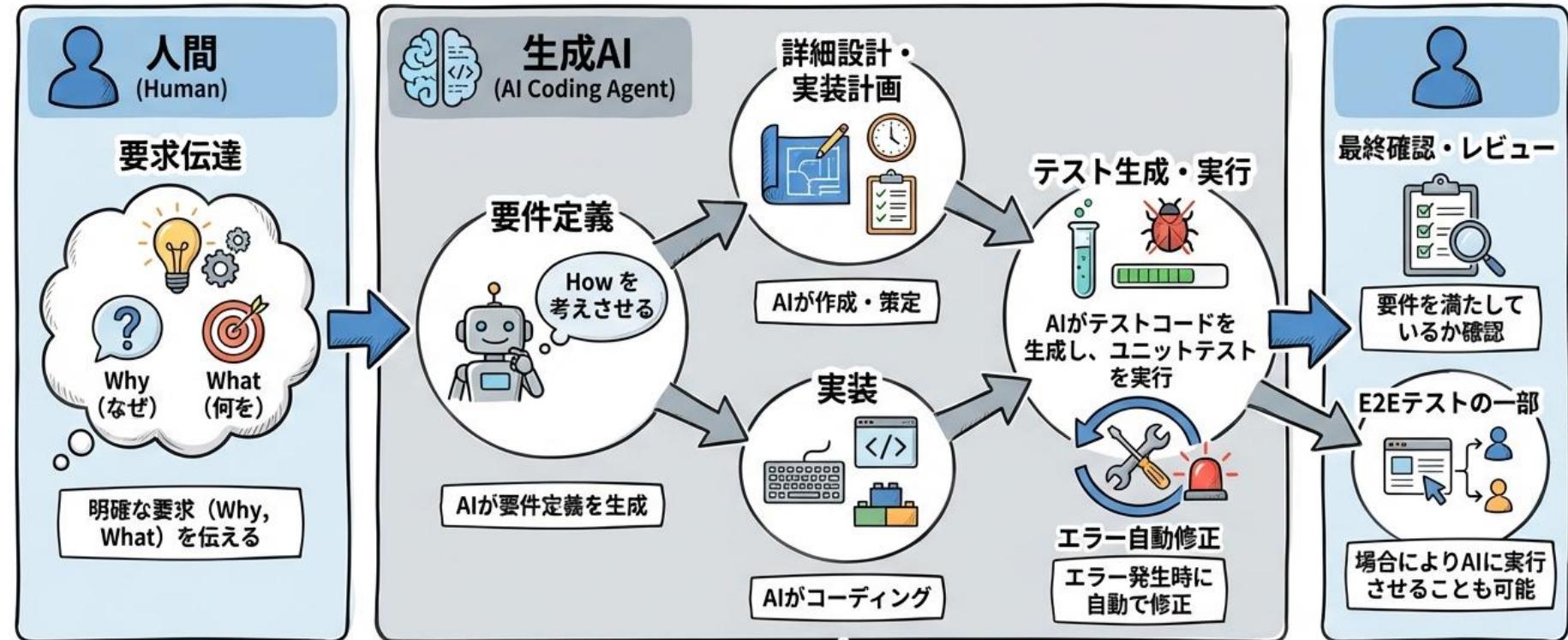
## ③ APIキー取得・コード出力



+



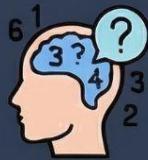
# バイブコーディング ワークフロー：人間とAIの協働



人間が本質的な価値創造 (Why/What) に集中し、AIが実行 (How) を担う、効率的で信頼性の高い開発プロセス。

# Google AI Studio でゲームを作ってみよう

AIの力で、あなたのアイデアを形に



## 数当てゲーム

AIが隠した数字を推理！



## オセロ風リバーシ

AIと対局！戦略を駆使。



## ブロック崩し

懐かしのアクション！  
AI生成ステージ。

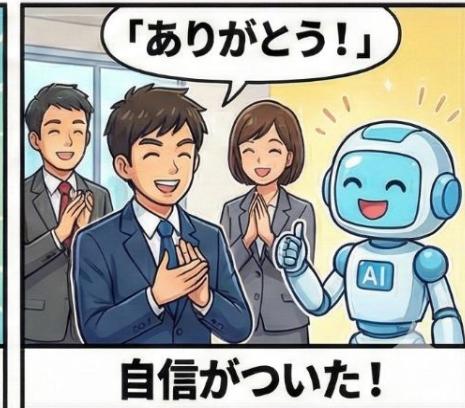
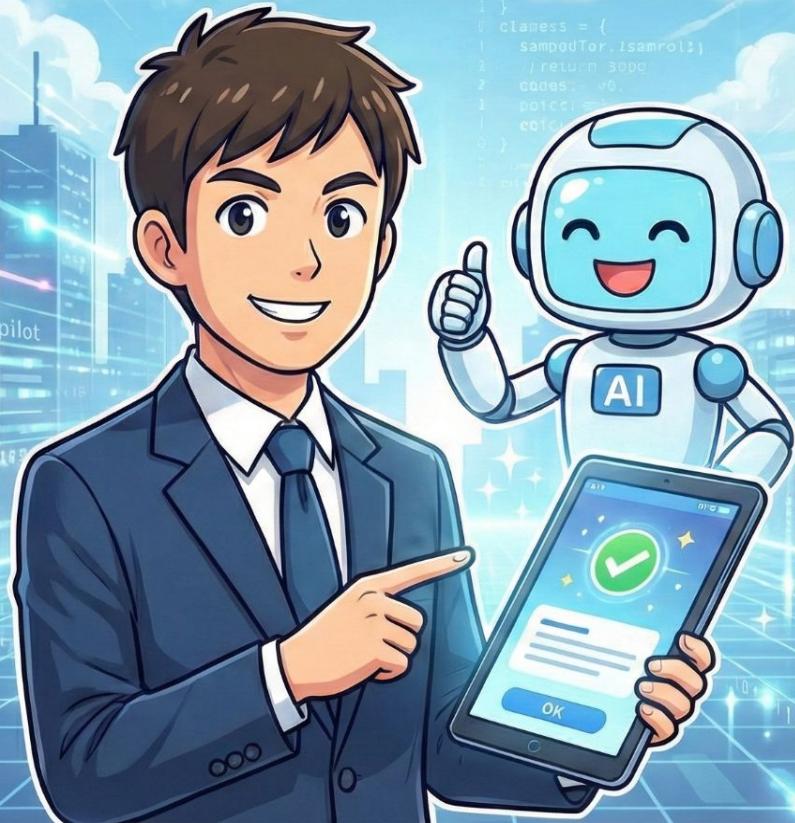


## インベーダーゲーム

迫りくるエイリアンを  
撃ち落とせ！AI制御。

Google AI Studio なら、コーディング未経験でも直感的に開発可能！

# 生成AI駆動開発ものがたり



# 要求仕様の定義

要求事項(Why/What):

- ・顧客の要望、  
自動化したい

要求事項  
(Why/What):  
顧客の要望、  
自動化したい

壁打ち  
(ラリー)

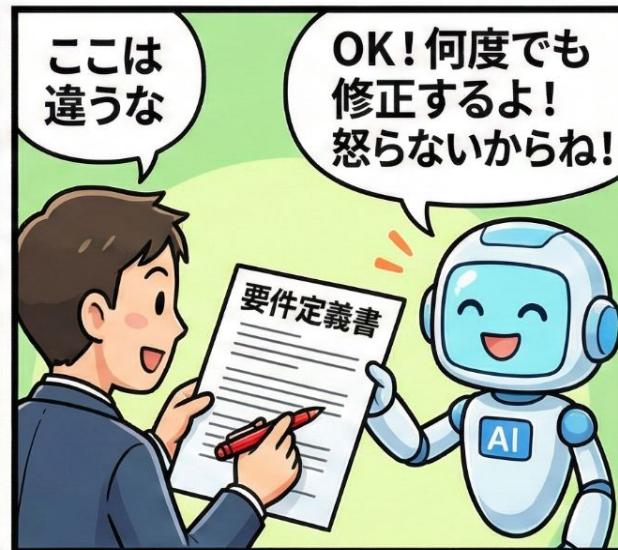
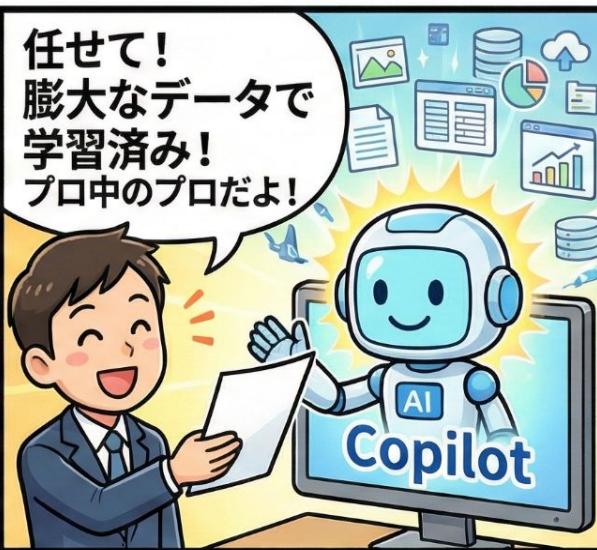


要件(How) 定義:  
API連携、DB設計、  
クラウド活用

コツが必要だが、慣れれば誰でもできる!  
現場を知るエンジニアが最強!



# 要件定義書、AIにおまかせ！



これが後のプロセスを  
成功させる  
**最重要ポイントだ！**



# 詳細設計と実装計画の作成ワークフロー (Copilot活用)

STEP 1:  
Copilotによるドラフト作成



STEP 2:  
自分でレビュー (必須)



STEP 3: エキスパート・他AIによるレビュー (任意・推奨)



STEP 1:  
Copilotによるドラフト作成

STEP 2:  
自分でレビュー (必須)

STEP 3: エキスパート・他AIによるレビュー (任意・推奨)

# Copilotとの実装・デバッグワークフロー

Copilot君、  
後はよろしく！

了解です！  
実装開始します！



STEP 1:  
Copilotによるドラフト作成

/review-security で  
情報漏洩チェック！



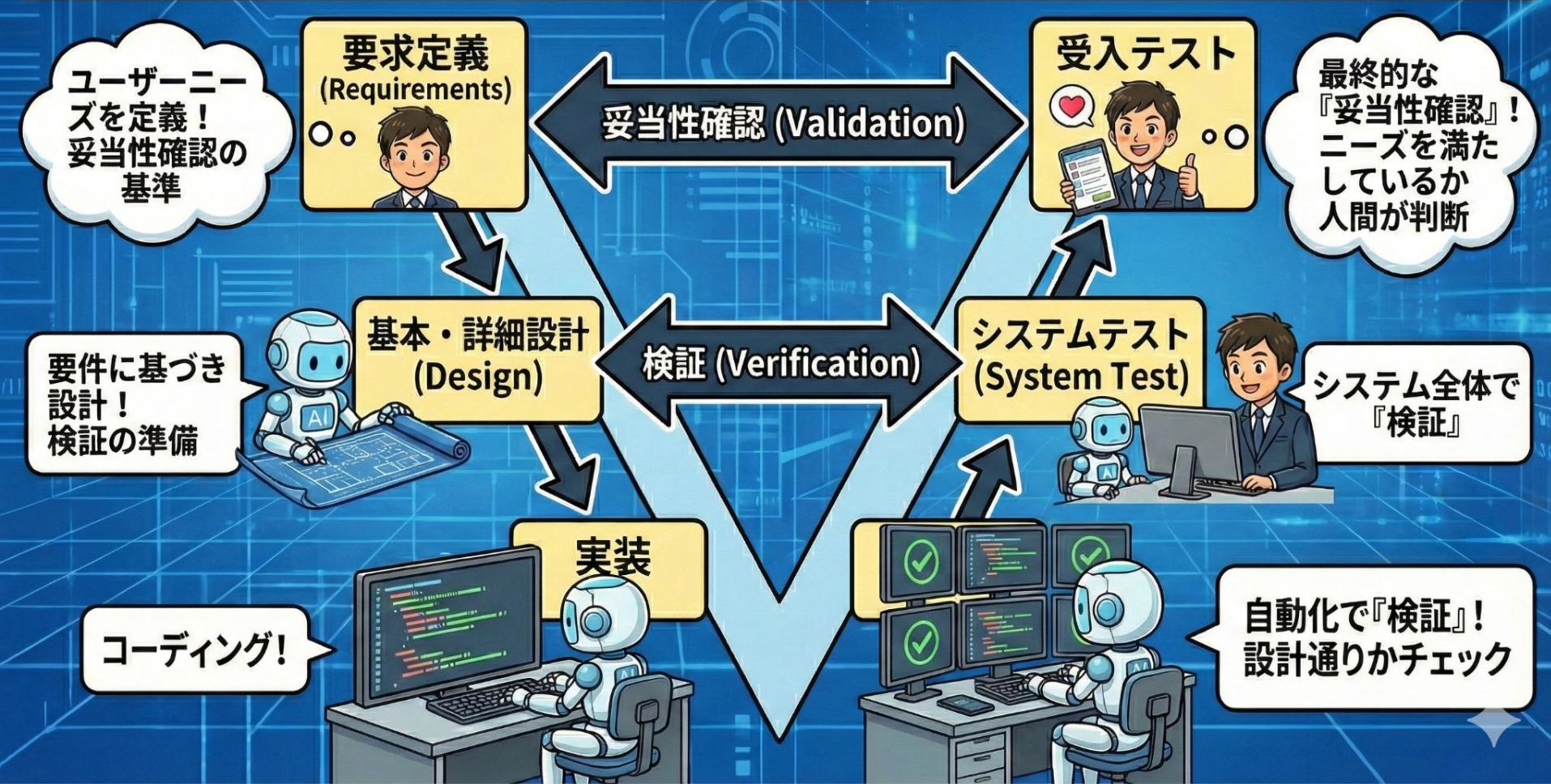
STEP 2:  
自分でのレビュー（必須）

エラーが出た…  
Copilot！



STEP 3: メッセージとスクショを  
送ってください！ 解決します！

# ソフトウェアV&Vのイメージ（人間とAIの協働）



# 重要なのは『検証』と『妥当性確認』の違い

## 検証 (Verification)

システムが要件通りか？



自動化(人間の責任)

検証は、要件に沿って作られているか  
システムがチェック！  
テストプログラムで自動化が  
効率的です！

## 妥当性確認 (Validation)

最終的な要求（ニーズ）を満たしているか？



妥当性確認は、最終的なニーズが満た  
されているか、人間が責任を持って  
確認しましょう！

# MiniConda Python環境構築ガイド (Python 3.12)

## ステップ1： MiniCondaのインストール



公式サイトからインストーラーを  
ダウンロードして実行します。



## ステップ2： 環境の作成

```
> conda create -n workshop python=3.12
Proceed ([y]/n)? y
```

ターミナル/Anaconda Promptを開きます。  
名前とバージョンを指定します。

## ステップ3： 環境の有効化

```
(base) > conda activate workshop
(workshop) >
```

activateコマンドを実行します。  
プロンプトの変化を確認します。



Python 3.12の環境「workshop」が使用可能になりました！

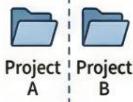
# Conda初心者卒業のための3つの基本コマンドセット

「とりあえずこれだけ出来ればOK」の最小フローをしっかり解説



## 1. conda create -n (環境を作る)

### 目的



プロジェクトごとに Python の世界を分けるためのコマンド。「このプロジェクトは Python 3.11+このライブラリ群」といった“箱”を作るイメージ。

### 使い方（実例）

```
$ conda create -n env-ml python=3.11
```

-n env-ml : 環境の名前（自分で決める）

python=3.11 : その環境で使う Python のバージョン

### 効用



プロジェクト A (py3.8)、プロジェクト B (py3.11) が共存できる。



ライブラリのバージョン違いで他プロジェクトが壊れる事故を防げる。



## 2. conda info -e (環境一覧を確認)

### 目的



今、どんな環境がいくつあるかを確認するためのコマンド。

### 使い方

```
$ conda info -e
```

```
# conda environments:  
#  
base          * /opt/miniconda3  
env-ml        /opt/miniconda3/envs/env-ml  
env-web       /opt/miniconda3/envs/env-web
```

\* が付いている行が「今アクティブな環境」。

### 効用



「さっそく作った環境の名前なんだっけ？」を確認できる。



似た名前の環境を量産してカオスになるのを防ぎやすい。



## 3. conda activate (環境を使い始める)

### 目的



作った環境を“今使う Python 環境”として有効にするコマンド。

### 使い方

```
$ conda activate env-ml
```

### 有効化前：

```
user@machine % → (env-ml) user@machine %
```

### 有効化後：

こうなっていれば、「今は env-ml の Python とライブラリを使っている」という状態。

(元に戻す: conda deactivate)

### 効用



python や pip を打ったとき、どの Python / ライブラリセットが使われるかが明確になる。



「どの環境でインストールしたんだっけ？」と迷子になるのを防げる。

## 4. 3コマンドの流れ（これだけできれば充分スタート）



1. 新しい環境を作る（最初だけ）  
\$ conda create -n env-ml python=3.11



2. 環境一覧を確認（名前・存在確認）  
\$ conda info -e



3. その環境を使い始める  
\$ conda activate env-ml



4. あとはいつも通り  
\$ python  
\$ pip install numpy

まずは「create して、info で確認して、activate して使う」 この3ステップを体に染み込ませれば、Conda の入口としては十分です。



次の一步へのヒント：パッケージを入れるときは「conda install パッケージ名」だけを追加するくらいがちょうどいい。

# Node.js 最新安定版 インストール&セットアップ

## インストール方法を選択



### A. 【推奨】バージョン管理ツール(nvm)を使う

1. nvm/nvm-windowsをインストール
2. コマンド実行: nvm install lts
3. バージョン指定: nvm use lts

特徴: 複数バージョン切替、権限管理が楽



### B. 【簡単】公式サイトからインストール

1. 公式サイト(nodejs.org)へアクセス
2. 「LTS (推奨版)」をダウンロード
3. インストーラーを実行 (Next連打)

特徴: すぐ使える、単純明快



## ✓ セットアップ確認 (共通)

- ターミナルでコマンド確認
- node -v (例: v20.11.0)
  - npm -v (例: 10.2.4)

バージョンが表示されれば完了!



# Git インストールと初期設定ガイド (Windows / macOS)

公式ダウンロードから git config までの流れを解説

## STEP 1: 対象OSの確認



Windows  
(対象)

対応OS:  
Windows 10/11  
(64-bit推奨)



macOS  
(対象)

対応OS:  
macOS  
(Apple Silicon / Intel)



## STEP 2: 公式サイトからダウンロード



※ 公式サイト ([git-scm.com](https://git-scm.com)) から、ご使用のOSに合わせたインストーラを選択してダウンロードしてください。

## STEP 3: 初期設定 (git config)

```
ユーザー名の設定  
(コミット履歴に表示されます)  
> git config --global user.name  
"Your Name"  
  
> git config --global user.email  
"you@example.com"
```

メールアドレスの設定  
(連絡先として使用されます)



設定完了  
(利用開始準備OK!)

注: ターミナル (Mac) またはコマンドプロンプト/PowerShell (Windows) を開いて実行してください。

# Visual Studio Code の拡張機能をインストールしよう

## STEP 1: 拡張機能ビューを開く



サイドバーのアイコンをクリック  
または **Ctrl+Shift+X**

## STEP 2: Pythonを検索



検索バーに「Python」と入力

## STEP 3: インストール



これで準備OK！  
対象の拡張機能の  
「インストール」をクリック

# —バイブコーディングAIアベンジャーズ—

動画も画像も、  
全部見えてるよ

マルチモーダル  
司令塔

ライブ  
デバッグ

GEMINI 3 PRO



CLAUDE  
SONNET 4.5  
& HAIKU 4.5

君の意図、  
汲み取ったよ

文脈の  
建築家

Haiku:  
要約しました!

CLAUDE SONNET 4.5 & HAIKU 4.5



サクッと  
終わらせて  
きました!

俊足な弟子  
爆速ラリー

GPT-5-MINI



論理の大賢者  
& 深腕エンジニア

実装は  
任せろ

GPT-5 & CODEX



# VS Code と Copilot で行う開発ワークフロー（壁打ちから計画まで）

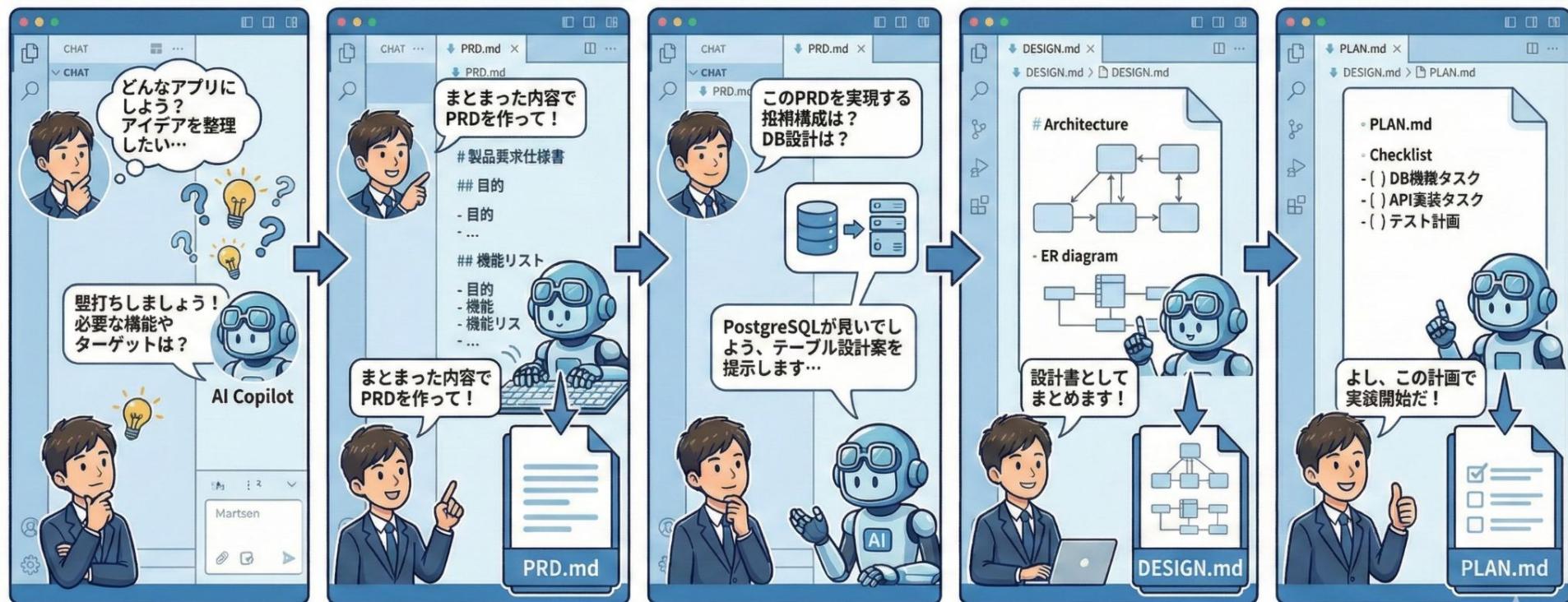
1. Askモード：  
要求仕様の壁打ち

2. Agentモード：  
PRD.md (要求仕様書) 作成

3. PRD.mdを元に  
詳細設計の壁打ち

4. DESIGN.md  
(詳細設計書) 作成

5. PLAN.md  
(実装・テスト計画) 作成



「対話（Ask）」で発散し、「指示（Agent）」でドキュメントに収束させる、効率的な共創フロー！

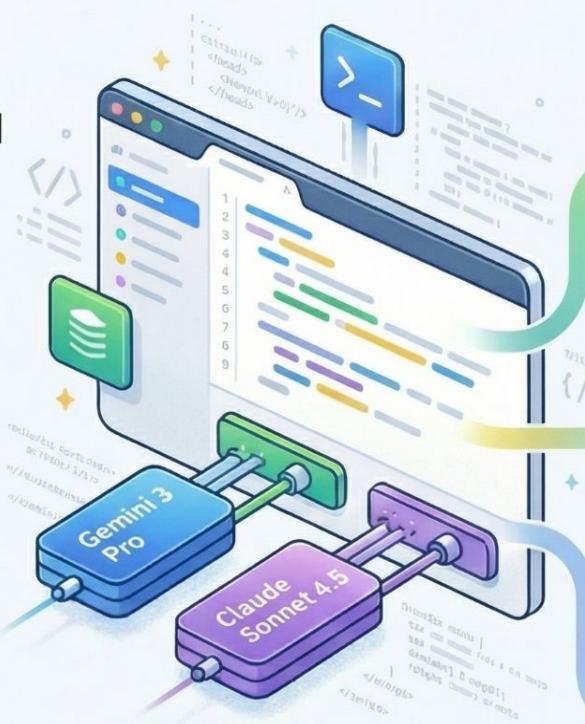
# 無料で使える！Googleの新AIコードエディタ「Antigravity」の魅力

## Google Antigravityとは？



### Google製の新しいAI搭載コードエディタ

開発を効率化する様々なAI機能が組み込まれています。



### 個人利用なら完全無料！

費用をかけずに強力なAI開発環境を試せます。



### 最新のAIモデルを複数利用可能

Gemini 3 ProやClaude Sonnet 4.5などを切り替えて使えます。

## 注目すべき主な機能



Fastモード Planningモード

### 2つのAI対話モード

高速な「Fastモード」と計画的な「Planningモード」を搭載しています。



### 無制限のコード補完機能

タブキーでの一括補完や、単語ごとの段階的な補完も可能です。



### 複数プロジェクトを同時管理

ワークスペース機能で複数のプロジェクトを切り替えながら作業できます。

<https://www.nstac.go.jp/use/literacy/ssdse/>