

第6回 並列開発環境 Git Worktree × 複数エージェント

Phase 2 | 環境整備 / 講義2h + ワークショップ6h

この回のねらい

- 複数Issueを同時並行で進める環境 **Git Worktree** を構築する
- ブランチ競合なしに **複数エージェントを並列稼働** させる
- 並列作業の成果を main へ **統合** する流れを体験する

Git Worktreeとは

1つのリポジトリを **複数ディレクトリ**に展開し、各々で別ブランチを扱える仕組み。

```
project/  
├── main/           # メインブランチ  
├── worktree-issue-1/ # エージェントA: Issue #1  
├── worktree-issue-2/ # エージェントB: Issue #2  
└── worktree-issue-3/ # エージェントC: Issue #3
```

なぜ有効か

- 各エージェントが独立ディレクトリ=**ブランチ競合なく並列開発**
- 運用イメージ：**「夜にIssueを一斉割り当て、翌朝には対応済み」**

勘所

独立したタスクに分割／同じファイルを跨がない／最後に統合

基本コマンド

```
git worktree add ../worktree-issue-1 -b feature/issue-1  
git worktree list  
git worktree remove ../worktree-issue-1
```

ワークショップ (6h)

1. **Worktreeを2～3個作成** (別ブランチ)
2. 各ディレクトリで `claude` を起動し、**独立タスクを並列実行**
3. 各ブランチをPR化 → mainへ順次 **統合** (コンフリクト処理)
4. 自分の実務での並列化候補を洗い出し

回のゴール

Git Worktreeで **複数エージェント**を並列稼働させ、
成果を main へ統合できる。

OJT課題：実務で2件以上をWorktree並列で進め、効果を記録