kod szkolenia: Agentic Coding / PL DL 1 d

Agentic Coding w Claude Code – podstawy

Odbiorcy szkolenia

Developerzy (junior–senior), DevOps, team leaderzy techniczni, startupy oraz osoby automatyzujące procesy developerskie – każdy, kto chce wykorzystać AI do przyspieszenia pracy z kodem. Idealne dla osób znających podstawy pracy z terminalem i Git, chcących wdrożyć agentic coding w codziennej pracy

Korzyści

- Pełne wprowadzenie do Claude Code jako narzędzia CLI oraz rozszerzenia VS Code/JetBrains.
- Umiejętność delegowania zadań kodowania do AI: debugowanie, refactoring, pisanie testów, code review.
- Praktyczna znajomość MCP (Model Context Protocol) i integracji z zewnętrznymi narzędziami.
- Konfiguracja autonomicznych workflow: hooks, subagenci, checkpointy, custom commands.

Program szkolenia

Blok 1. Wprowadzenie do Claude Code & agentic coding

- Czym jest agentic coding i jak zmienia workflow developera.
- Instalacja Claude Code: npm, konfiguracja API key/login, pierwsze uruchomienie.
- Interfejs CLI vs VS Code extension – kiedy co używać.

Blok 2. Podstawowe komendy i nawigacja

- Struktura komend: /help, /model, /clear, /compact.
- Eksploracja codebase: jak Claude czyta i rozumie kod.
- Prompt engineering dla kodu, thinking budget hierarchy

Blok 3. Edycja kodu i debugging

- Edycja plików: tworzenie, modyfikacja, diff preview.
- Debugging z Claude: analiza błędów, stack trace, sugestie fix.

- Praktyczne ćwiczenie: fix buga w sample projekcie.

Blok 4. Refaktoring i testy

- Refaktoring kodu: ekstrakcja funkcji, optymalizacja, clean code.
- Generowanie testów jednostkowych i integracyjnych.
- TDD z Claude Code: test-first development workflow.

Blok 5. Git workflow i GitHub integration

- Commit, push, tworzenie branchy z poziomu Claude Code.
- Integracja z GitHub: issues, pull requests, code review.
- Custom commands: automatyzacja powtarzalnych zadań (/project:fix-issue).

Blok 6. MCP – Model Context Protocol

- Czym jest MCP i jak rozszerza możliwości Claude Code.
- Konfiguracja MCP servers: project config vs global config.
- Integracje: zewnętrzne API.

Blok 7. Autonomiczna praca: hooks, subagenci, checkpointy

- Hooks: automatyczne akcje (linting, testy po zmianach).
- Subagenci: delegowanie zadań równoległych.
- Checkpointy: bezpieczna praca autonomiczna z możliwością rollback.

Blok 8. Best practices i zaawansowane scenariusze

- Bezpieczeństwo: sandboxing, permissions, sensitive data.
- Claude Code for Web: praca z przeglądarki, przenosiny do do CLI.
- Tips & tricks: efektywna praca, koszty, wybór modelu.

Oczekiwane przygotowanie uczestnika

Konieczna jest posiadana licencja i zainstalowane już środowisko Claude Code (wysyłamy instrukcję instalacji przed szkoleniem).
Konto Anthropic (API key) lub Claude Pro/Max. Node.js 18+ zainstalowany.

Czas trwania

1 dni / 7 godzin

Język

Język: szkolenie – polski lub angielski