

bkit (Vibecoding Kit) 사용 가이드

Claude Code에서 AI-Native 개발을 위한 bkit 플러그인 사용법

bkit v1.4.7 | Last Updated: 2026-02-01

1. bkit이란?

bkit (Vibecoding Kit)은 Claude Code에서 PDCA 방법론 기반의 체계적인 개발을 지원하는 플러그인입니다.

핵심 기능

- **PDCA Cycle:** Plan → Design → Do → Check → Act
- **Multi-Agent System:** 전문화된 AI 에이전트 자동 호출
- **Level-based Development:** Starter / Dynamic / Enterprise
- **자동화된 품질 관리:** Gap Analysis, Code Review

2. 시작하기

Claude Code 설치

```
# Claude Code CLI 설치
npm install -g @anthropic-ai/claude-code

# 프로젝트에서 실행
cd materials-data-portal
claude
```

bkit 시작 옵션

옵션	설명
Learn bkit	bkit 사용법 학습
Learn Claude Code	Claude Code 기능 학습
Start new project	새 프로젝트 시작
Start freely	자유 대화 모드

3. PDCA Cycle

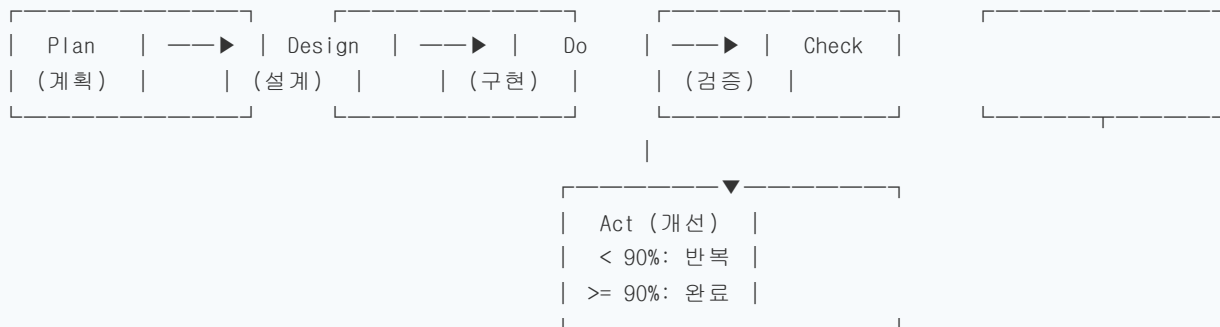
통합 명령어: /pdca

```

/pdca plan {feature}      # 계획 수립
/pdca design {feature}    # 설계 문서 작성
/pdca do {feature}        # 구현
/pdca analyze {feature}   # Gap 분석
/pdca iterate {feature}   # 자동 개선
/pdca report {feature}    # 완료 보고서
/pdca status              # 현재 상태 확인
/pdca next                # 다음 단계 안내

```

PDCA 흐름도



4. 전문 에이전트

bkit은 상황에 맞는 에이전트를 자동으로 호출합니다.

에이전트	역할	트리거 키워드
gap-detector	설계-구현 Gap 분석	verify, 검증, 확인
pdca-iterator	자동 반복 개선	improve, 개선, 고쳐줘
code-analyzer	코드 품질 분석	analyze, 분석, 품질
report-generator	완료 보고서 생성	report, 보고서
design-validator	설계 문서 검증	validate design, 설계 검증
starter-guide	초보자 가이드	help, 도움, 모르겠어
bkend-expert	BaaS 백엔드 전문가	login, auth, database
enterprise-expert	엔터프라이즈 아키텍처	microservices, k8s
infra-architect	인프라 설계	AWS, terraform, CI/CD

에이전트 자동 호출 예시

```

사용자: "이거 검증해줘"
→ gap-detector 자동 호출

사용자: "코드 분석해줘"
→ code-analyzer 자동 호출

사용자: "개선해줘"
→ pdca-iterator 자동 호출

```

5. 스킬 (Skills)

레벨별 스킬

스킬	레벨	용도
<code>/starter</code>	Starter	정적 웹사이트, 포트폴리오
<code>/dynamic</code>	Dynamic	풀스택 앱, 인증, DB
<code>/enterprise</code>	Enterprise	마이크로서비스, K8s

개발 파이프라인

<code>/development-pipeline</code>	# 9단계 파이프라인 안내
<code>/phase-1-schema</code>	# 스키마/용어 정의
<code>/phase-2-convention</code>	# 코딩 컨벤션
<code>/phase-3-mockup</code>	# UI 목업
<code>/phase-4-api</code>	# API 설계
<code>/phase-5-design-system</code>	# 디자인 시스템
<code>/phase-6-ui-integration</code>	# UI 통합
<code>/phase-7-seo-security</code>	# SEO/보안
<code>/phase-8-review</code>	# 코드 리뷰
<code>/phase-9-deployment</code>	# 배포

유틸리티 스킬

<code>/code-review</code>	# 코드 리뷰
<code>/zero-script-qa</code>	# 로그 기반 QA
<code>/claude-code-learning</code>	# Claude Code 학습
<code>/mobile-app</code>	# 모바일 앱 개발
<code>/desktop-app</code>	# 데스크톱 앱 개발

6. 실전 사용 예시

예시 1: 새 기능 개발

```
# 1. 계획 수립
사용자: "/pdca plan 사용자 인증 기능"

# 2. 설계 문서 작성
사용자: "/pdca design 사용자 인증 기능"

# 3. 구현
사용자: "로그인 기능 구현해줘"

# 4. Gap 분석
사용자: "검증해줘" 또는 "/pdca analyze 사용자 인증 기능"

# 5. 결과에 따라
# - Match Rate < 90%: 자동 개선
# - Match Rate >= 90%: 완료 보고서
```

예시 2: 코드 품질 확인

```
사용자: "이 코드 분석해줘"
→ code-analyzer 자동 호출
→ 품질, 보안, 성능 이슈 리포트
```

예시 3: 초보자 도움

```
사용자: "어떻게 시작해야 할지 모르겠어"
→ starter-guide 자동 호출
→ 단계별 안내 제공
```

7. Feature Usage Report

bkit은 매 응답마다 사용 현황을 보고합니다:

bkit Feature Usage

✔ Used: [사용된 기능]

▶▶ Not Used: [미사용 기능] (이유)

💡 Recommended: [다음 추천 기능]

8. 다국어 지원

bkit은 8개 언어를 자동 감지합니다:

us English	KR 한국어	JP 日本語	CN 中文
es Español	FR Français	DE Deutsch	IT Italiano

9. 주의사항

1. **AI는 완벽하지 않습니다** - 중요한 결정은 항상 검증하세요
2. **PDCA 순서 권장** - Plan → Design → Do → Check 순서로 진행
3. **Gap Analysis < 90%** - 자동 개선 반복 (최대 5회)
4. **문서 우선** - 구현 전 계획/설계 문서 작성 권장

Claude Code 문서: <https://docs.anthropic.com/claude-code>