



Claude Code Quick Start Guide

Set-up Instructions and
Top 20 Essential Shortcuts

*As featured in “How to Train AI to
Coach Like You,” Corey Blake, 2025*



* Welcome to Claude Code

CLAUDE
CODE

Press **Enter** to continue

INTRODUCTION

Welcome to Claude Code!

This guide provides everything you need to get started with Claude Code, including setup instructions and the 20 most essential shortcuts to become productive quickly.



What You'll Find Inside

- Part 1: Installation.....3**
- Part 2: Creating Your Safe Workspace.....6**
- Part 3: The Absolute Basics.....7**
- Part 4: Managing Your Work.....8**
- Part 5: Getting Things Done.....9**
- Part 6: Configuration and Settings.....10**
- Part 7: Advanced But Essential.....11**
- Your First Claude Code Session.....13**
- Quick Reference: When to Use What.....14**
- Pro Tips for Beginners.....15**

PART 1

Installation

First, let's get Claude Code on your computer—it takes about 5 minutes.

FOR MAC USERS



Install Node.js first

- Go to nodejs.org
- Click the big green "LTS" button
- Open the downloaded file and click through the installer



Open Terminal

- Press **Cmd + Space**
- Type "Terminal"
- Press Enter



Install Claude Code

- Copy and paste this line, then press Enter:

```
npm install -g @anthropic-ai/claude-code
```



Sign In

- Type **claude** and press Enter
- A browser window opens - sign in with your Claude account
- Return to Terminal - you're ready!



PART 1

Installation

First, let's get Claude Code on your computer—it takes about 5 minutes.

FOR WINDOWS USERS

Choose the easiest option for you.

Option A: Simple Installation (Recommended for Beginners)



Install Git for Windows

- Go to git-scm.com
- Download for Windows
- Run installer (keep all default settings)



Open PowerShell

- Right-click Start button
- Select "Windows PowerShell"



Install Claude Code

- Copy and paste this, then press Enter:

```
irm https://claude.ai/install.ps1 | iex
```



Sign In

- Type **claude** and press Enter
- Sign in when your browser opens
- You're done!

Option B: If You Already Have Node.js



In PowerShell, paste:

```
npm install -g @anthropic-ai/claude-code
```



PART 1

Installation

First, let's get Claude Code on your computer—it takes about 5 minutes.

FOR LINUX USERS

➔ Open Terminal (Ctrl + Alt + T)

➔ Install Claude Code:

```
curl -fsSL https://claude.ai/install.sh | bash
```

➔ Type **claude** to start and sign in

AFTER INSTALLATION

Verify Everything Worked

In your terminal, type:

claude doctor

This checks that Claude Code is properly installed.

Need help? Visit docs.claude.com

PART 2

Creating Your Safe Workspace

Let's create a controlled environment for learning Claude Code

STEP 1: Create Your Practice Folder

- ➔ **Go to your desktop**
- ➔ **Create a new folder called "Claude Code Practice"**
 - Mac: Right-click on desktop → New Folder
 - Windows: Right-click on desktop → New → Folder

This will be your safe **sandbox**—Claude Code will only access files within this folder

STEP 2: Open Terminal in Your Practice Folder

- ➔ **Mac Users:**
 - Right-click on your "Claude Code Practice" folder
 - Select "Services" → "New Terminal at Folder"
- ➔ **Windows Users:**
 - Right-click on your "Claude Code Practice" folder
 - Select "Open in Terminal" or "Open PowerShell window here"
- ➔ **Alternative for Any System:**
 - Open your terminal/command prompt normally
 - Type: **cd Desktop/"Claude Code Practice"**

STEP 3: Launch Claude Code

In your terminal window, type:

```
claude
```

✓ **Important:** Claude Code now has access ONLY to files within this "Claude Code Practice" folder and its subfolders. Your other files remain private. This is your safe space to learn and experiment!

PART 3

The Absolute Basics

Start here—master these four commands first.

➔ Launch Claude Code

Command: `claude`

Opens the interactive interface where you can chat with Claude and get coding help.

```
$ claude
```

➔ See All Available Commands

Command: `/help`

Your safety net - shows every command available when you need guidance.

```
> /help
```



➔ Stop Claude Mid-Task

Shortcut: `ESC`

Changed your mind? Press Escape to interrupt and redirect.

- Press once: Stop current action
- Press twice: Jump back in conversation history

➔ Exit Claude Code

Command: `/exit` or `q`

Done for now? Exit cleanly

```
> /exit
```


PART 4

Managing Your Work

Essential commands for organizing your sessions

➔ Start Fresh While Keeping Context

Command: `/compact`

Running out of token space? This summarizes everything and starts a new session; *perfect when Claude says context is getting full—enables infinite sessions.*

```
> /compact
```

➔ Continue Where You Left Off

Command: `/resume`

Pick up your last conversation exactly where it ended.

```
$ claude  
> /resume
```

➔ Track Your Tasks

Command: `/todo`

Keep organized with built-in task management—great for multi-step projects.

```
> /todo
```

➔ Clear Everything For Fresh Start

Command: `/cle`

Need a complete fresh start? This wipes the slate clean (except `claude.md` files).

```
> /cle
```



PART 5

Getting Things Done

Power shortcuts for productive work

→ Plan Before Acting

Shortcut: **Shift+Tab**

Makes Claude analyze and plan before making changes. Use this for complex refactoring or when you want to review the plan first.

- Mac: Press Shift+Tab twice
- Windows: Press once or use Alt+M

→ Accept Suggestions

Shortcut: **Tab**

Claude suggesting something? Hit Tab to accept.

→ Deep-Thinking Mode

Command: **think hard**

Add this to your prompt for complex problems requiring extended analysis.

```
> think hard about the architecture  
for this feature
```

→ Command History

Shortcut: **↑ / ↓ arrows**

Reuse previous commands quickly with up/down arrows.



PART 6

Configuration and Settings

Customize Claude Code to your needs

➔ Check or Change AI Model

Command: `/model`

See which model you're using or switch between them.

```
> /model  
Currently using: claude-3-sonnet  
Available: sonnet, opus, opus-plan
```

➔ Access All Settings

Command: `/settings`

Configure Claude Code to your preferences.

```
> /settings
```

➔ Initialize Your Project

Command: `/init`

Creates a `claude.md` file for project-specific instructions that loads automatically every session.

```
> /init
```


PART 7

Advanced (But Essential)

Level up your Claude Code skills

➔ Headless Mode for Scripts

Command: `claude-b "task"`

Run Claude without the interactive interface—perfect for automation.

```
$ claude-b "write a README  
for this project"
```

➔ View Session History

Command: `/logs`

Debug issues or review what happened in your session.

```
> /logs
```

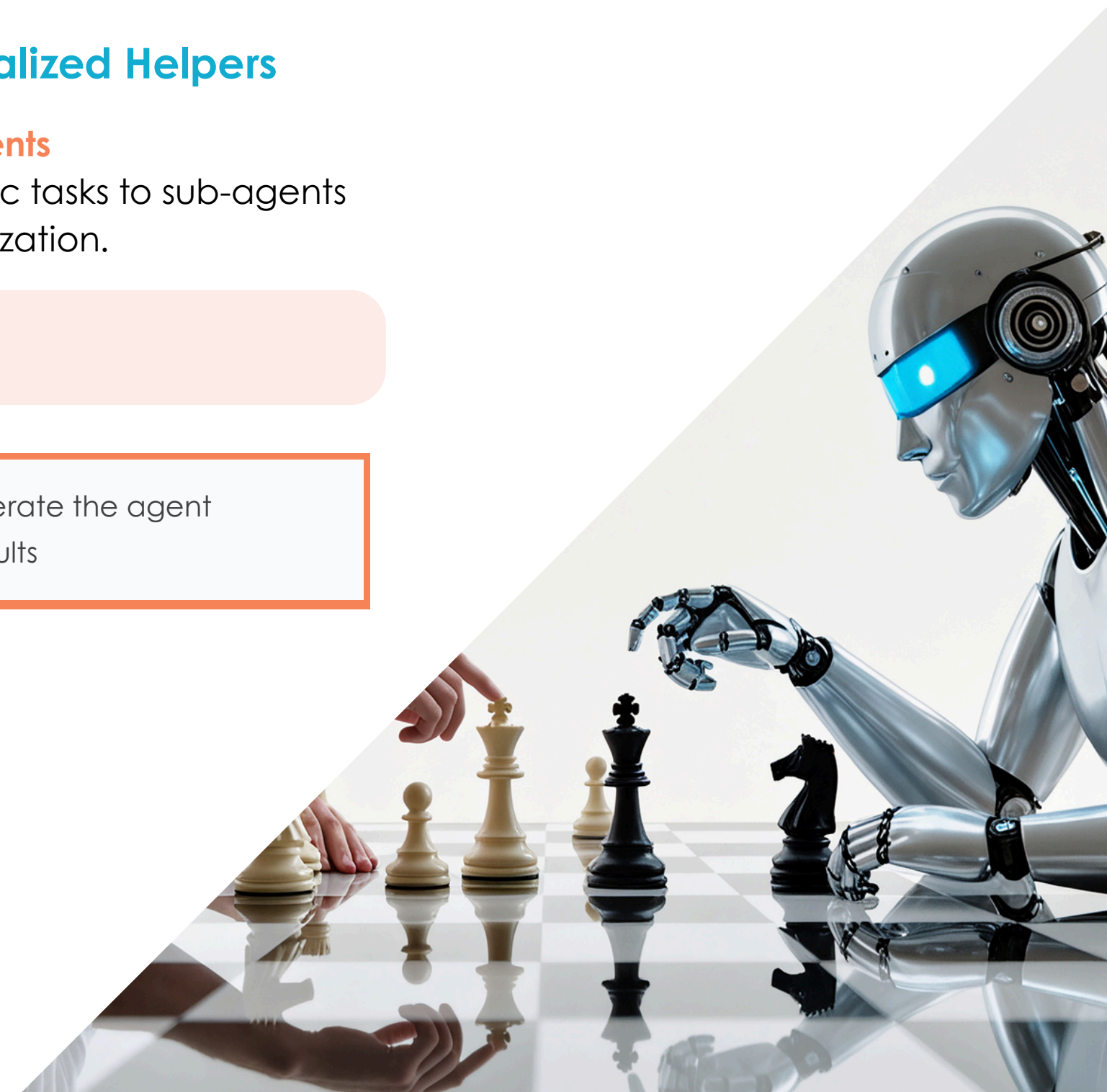
➔ Create Specialized Helpers

Command: `/agents`

Delegate specific tasks to sub-agents for better organization.

```
> /agents
```

✓ **Tip:** Let Claude generate the agent description for best results



PART 7

Advanced (But Essential)

Customize Claude Code to your needs

➔ Batch Operations for Speed

Technique: Multiple file operations

Ask Claude to do multiple file reads or searches in one request for speed.

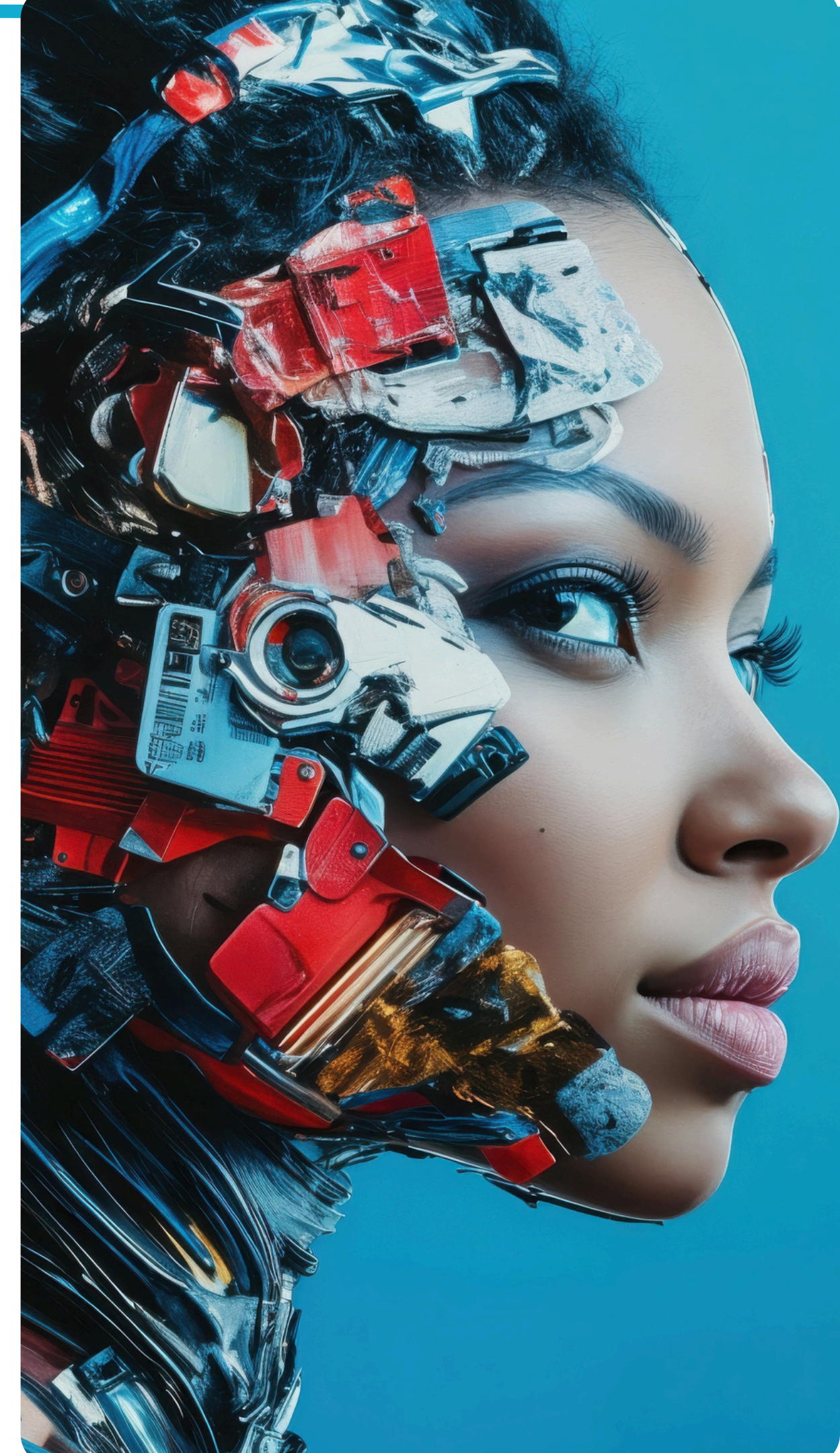
```
> Read the README.md, package.json, and all  
files in src/components/
```

➔ Persistent Project Instructions

Feature: `claude.md` files

Create a `claude.md` file in your project root with:

- Project conventions and coding standards
- Common tasks and team guidelines
- This loads automatically every time Claude Code starts in that directory!



GETTING STARTED

Your First Claude Code Session

Follow these steps to get started immediately.

➔ STEP 1: Start Claude Code

```
$ claude
```

➔ STEP 2: Initialize Your Project

```
> /init
```

➔ STEP 3: Ask For Help With Your Task

```
> Help me understand this codebase
```

➔ STEP 4: Use Plan Mode For Changes

```
> [Press Shift+Tab] Refactor the authentication system
```

➔ STEP 5: Track Your Progress

```
> /todo
```

➔ STEP 6: When Context Fills Up

```
> /compact
```

➔ STEP 7: Exit When Done

```
> /exit
```


QUICK REFERENCE

When to Use What

Feeling lost?	Type /help
Made a mistake?	Press ESC to stop, then redirect
Context getting full?	Use /compact to continue infinitely
Complex task?	Press Shift+Tab for Plan Mode
Want to track progress?	Use /todo for task management
Need more power?	Use /model to switch to Opus
Project-specific rules?	Create a claude.md file



FINAL NOTE

Pro Tips for Beginners

- ➔ **Start simple**—Master the first four commands before moving on.
- ➔ **Use Plan Mode**—For any complex changes, let Claude plan first.
- ➔ **Don't fear `/compact`**—It's your friend for long sessions.
- ➔ **Create `claude.md`**—Document your project conventions once.
- ➔ **Interrupt freely**—`ESC` is there to help you redirect anytime.

Next Steps

Once you're comfortable with these basics:

- Explore `/hooks` for automation
- Try MCP servers with `/mcp` for external integrations
- Set up GitHub integration with `@claude` mentions



REMEMBER: Claude Code is here to help! Don't hesitate to ask questions or use `/help` whenever you're unsure. Every expert was once a beginner who kept practicing.

Happy coding! 🚀



We transform cultures through trust and care.

At Round Table Companies (RTC), we provide high-impact people development for leaders, teams, and individuals of companies preparing for (or in the midst of) significant growth and change.

We care for the heart and spirit of your company culture using practical tools and immersive training that combines **trust**, **psychological safety**, and **AI technology** to unlock potential and awaken high performance.

[Learn More](#)