

Intelligent Book Workshop Cheat Sheet

v 0.01 12/1/2025

<https://bit.ly/dans-book-workshop>

Setup

```
$ mkdir ~/projects
$ cd projects
$ git clone https://github.com/dmccreary/claude-skills
$ export BK_HOME=$HOME/projects/claude-skills
$ cd claude-skills/scripts
$ ./bk-install-scripts
$ ./bk-install-skills
$ cd ~/projects
$ git clone https://github.com/GITHUB_ID/BOOK_REPO
$ cd BOOK_REPO
$ claude
```

Book Levels

L5: Autonomous

L4: Chatbot

L3: Adaptive Textbook

L2: Interactive Textbook

Today

L1: Static Textbook

2001 Bloom Taxonomy

Create

Analyze

Evaluate

Apply

Understand

Remember



Sample Book Generation Skills

```
> /skill install-mkdocs-template
> /skill course-description-analyzer
> /skill learning-graph-generator
> /skill install-learning-graph-viewer
> /skill book-chapter-generator
> /skill chapter-content-generator chapter 1
> /skill glossary-generator
> /skill faq-generator
> /skill quiz-generator
> /skill reference-generator
> /skill linkedin-announcement-generator
```

Sample BK Scripts

1. bk	Build/Book utilities menu
2. bk-analyze-skill-usage	Skill usage analysis report (minutes, tokens)
3. bk-capture-screenshot	Capture a screenshot of a MicroSim using Chrome
4. bk-check-loops	Check for loops/cycles in a learning graph
5. bk-diagram-reports	Diagram and MicroSim quality report
6. bk-generate-book-metrics	Comprehensive metrics reports for book
7. bk-install-scripts	Creates symbolic links for all bk* scripts in \$...
8. bk-install-skills	Creates symbolic links in ~/.claude/skills/ for skills
9. bk-install-social-override-plugin	Installs social_override plugin for mkdocs
10. bk-list-skills	List all Claude skills with their descriptions
11. bk-microsim-quality-report-generator	Generate quality reports for MicroSims
12. bk-resize-images	Compresses large images to approximately 300KB ...
13. bk-status	Check the status of an intelligent textbook pro...

Sample MicroSim Skills

```
> /skill microsim-p5
> /skill mermaid-generator
> /skill microsim-vis-network
> /skill chartjs-generator
> /skill timeline-generator
> /skill map-generator
> /skill math-function-plotter-plotly
> /skill causal-loop-microsim-generator
> /skill microsim-matcher
> /skill microsim-screen-capture
> /skill readme-generator
> /skill microsim-standardization
```

Tip and Tricks

1. Use the @docs/{TAB} syntax to specify a path
2. Ask claude what skills it knows about
3. Ask claude what new skills should be created to save you time
4. Use the skill-creator skill to create specialized media types (circuit diagram)
5. Always do `git pull` on the claude skills directory before you start a session
6. If you don't see a new skill, rerun the **bk-install-skills** script
7. Run the `>/ide` command to install claude on your IDE (VSCode)
8. Use the `>/skill microsim-matcher` to match a learning objective with a microsim type
9. If your book is always checked into GitHub you can set up a UNIX command alias `claude-danger` to add the `--dangerously-skip-permissions`
10. Use the `mkdocs gh-deploy` to publish your book to GitHub Pages
11. Use the `microsim-add-icons` skill to add the license and full-screen icons to the lower right corner of your MicroSim (adds 44 lines of code)