# ⚡ Task Master AI - Quick Reference

AI-Powered Task Management for Development Projects • github.com/eyaltoledano/claude-task-master

## 🚀 SETUP & INIT

`task-master init`
*Initialize new project*

`task-master parse-prd <file>`
*Parse PRD and generate tasks*

`task-master generate`
*Generate task files from tasks.json*

## 🗒️ VIEW TASKS

`task-master list`
*List all tasks*

`task-master list --status=<status>`
*Filter by status*

`task-master list --with-subtasks`
*Include subtasks in list*

`task-master next`
*Show next task to work on*

`task-master show <id>`
*Show specific task details*

`task-master show 1.2`
*View specific subtask*

## ✏️ UPDATE TASKS

`task-master update --from=<id> --prompt="..."`
*Update from specific task*

`task-master update-task --id=<id> --prompt="..."`
*Update single task*

`task-master update-subtask --id=<id> --prompt="..."`
*Append to subtask (preserves original)*

## 🎯 TASK STATUS

`task-master set-status --id=<id> --status=<status>`
*Set single task status*

`task-master set-status --id=1,2,3 --status=<status>`
*Bulk status update*

`task-master set-status --id=1.1,1.2 --status=<status>`
*Update subtask status*

## 🔍 EXPAND & RESEARCH

`task-master expand --id=<id> --num=<n>`
*Expand task with subtasks*

`task-master expand --all`
*Expand all pending tasks*

`task-master expand --id=<id> --research`
*Research-backed expansion*

`task-master expand --all --force`
*Force regenerate subtasks*

## 📊 ANALYSIS

`task-master analyze-complexity`
*Analyze all task complexity*

`task-master complexity-report`
*Display complexity report*

`task-master analyze-complexity --research`
*Research-backed analysis*

`task-master analyze-complexity --threshold=6`
*Custom complexity threshold (1-10)*

## 🔗 DEPENDENCIES

`task-master add-dependency --id=<id> --depends-on=<id>`
*Add task dependency*

`task-master remove-dependency --id=<id> --depends-on=<id>`
*Remove dependency*

`task-master validate-dependencies`
*Validate without fixing*

`task-master fix-dependencies`
*Auto-fix invalid dependencies*

## ✔️ MAINTENANCE

`task-master clear-subtasks --id=<id>`
*Clear subtasks from task*

`task-master clear-subtasks --all`
*Clear all subtasks*

`task-master add-task --prompt="..."`
*Add new task with AI*

`task-master add-task --prompt="..." --dependencies=1,2`
*Add task with dependencies*

## 💬 MCP CHAT COMMANDS

`Initialize taskmaster-ai in my project`
*Set up Task Master*

`Parse my PRD at .taskmaster/docs/prd.txt`
*Parse requirements*

`What's the next task I should work on?`
*Get next task*

`Help me implement task 3`
*Get AI assistance*

`Show me tasks 1, 3, 5`
*View multiple tasks*

`Expand task 4`
*Break into subtasks*

`Research best practices for [topic]`
*Research-backed info*

`Change main model to claude-code/sonnet`
*Switch AI model*

---

### ▦ Research Flag

- Add `--research` to update-task, update-subtask, expand, or analyze-complexity
- Uses Perplexity AI for research-backed enhancements
- Requires PERPLEXITY_API_KEY in config
- Example: `task-master expand --id=5 --research`

---

### ⚙ Quick Tips

- **Task IDs:** Use comma-separated IDs: 1,2,3
- **Subtasks:** Reference as `parentId.subtaskId` (e.g., 5.2)
- **Statuses:** pending, in-progress, completed, blocked
- **MCP Mode:** Use natural language in editor chat
- **Claude Code:** No API key needed with local Claude