



SÉRIE SPÉCIALE · CLAUDE CODE



Le Mémento des Commandes

Toutes les slash commands de Claude Code, réunies, expliquées et complétées.

Le mémo complet pour piloter Claude Code : session, contexte, configuration, mémoire, git et extensions. Plus de 45 commandes.

Gratuit pour la communauté SYSTINFO





BON À SAVOIR

Qu'est-ce qu'une slash command ?

Une slash command est un raccourci tapé dans la session Claude Code. Au lieu d'écrire une longue consigne, vous tapez deux mots qui commencent par « / », et l'outil agit aussitôt.

- **Tapez « / » pour tout voir**

Une simple barre oblique affiche la liste complète des commandes disponibles dans votre session.

- **Tab pour compléter**

La touche Tab auto-complète les commandes, les chemins de fichiers et les mentions « @ ».

- **Plugins et skills en ajoutent**

Vos extensions installées enrichissent la liste : tapez /plugins et /mcp pour voir ce qu'elles exposent.

Source : *cheatsheet Staying Ahead*, complété par la doc Claude Code (docs.claude.com)



Six familles, plus l'essentiel.

Les commandes sont classées par usage. En fin de fiche, un rappel des raccourcis clavier, des mentions « @ », des modes de permission et du lancement en ligne de commande.

01 **Coups de maître de session**

· goal, plan, branch, rewind...

02 **Gestion de session**

· clear, compact, resume, export...

03 **Contexte et revue**

· context, diff, usage, copy...

04 **Configuration et réglages**

· config, model, effort, hooks...

05 **Mémoire et projet**

· init, memory, add-dir, #...

06 **Travail, qualité, extensions**

· review, commit, mcp, agents...



Coups de maître de session

/goal

Définit un objectif ; Claude travaille tour après tour jusqu'à l'atteindre, avec suivi de progression. Vous définissez le « fini », il trouve le chemin.

/btw

Posez une question annexe avec le contexte courant pendant qu'une tâche longue continue, sans l'interrompre.

/branch *alias /fork*

Duplique la conversation depuis l'état actuel pour explorer deux pistes en parallèle.

/rewind *alias /undo*

Revient à un point de contrôle antérieur, conversation ET fichiers, pour effacer une impasse.

/rc *alias /remote-control*

Pilotez la session depuis votre téléphone par messages de suivi pendant que vous vous éloignez.



Coups de maître de session (suite)

/plan

Entre en mode plan : Claude détaille les étapes avant d'exécuter, pour valider l'approche.

/workflows

Lance des workflows intégrés (recherche approfondie, etc.) sans ouvrir l'application web.

/voice

Dictez vos prompts à la voix plutôt qu'au clavier.



Gestion de session

/clear

Vide le contexte de la conversation. À faire après chaque tâche terminée.

/compact

Comprime le contexte sans perdre l'essentiel ; un mot-clé de focus priorise ce qui est gardé.

/rename

Nomme la session pour la retrouver facilement (utile avec /resume et /rc).

/resume

Reprend une session passée par identifiant ou par nom ; contexte et fichiers restaurés.

/recap

Donne un résumé rapide du contexte de la session en cours.

/export

Exporte toute la conversation dans un fichier (documentation, partage, archivage).



Gestion de session (suite)

/exit *alias /quit*

Termine la session.

/login

Se reconnecter à votre compte Claude.

/logout

Se déconnecter du compte.

/status

Affiche le compte, le modèle et l'environnement actifs.



Contexte et revue

/context

Visualise le contexte sous forme de grille : ce qui est chargé et l'espace occupé.

/diff

Ouvre une visionneuse de différences interactive pour relire chaque changement avant de l'accepter.

/usage

Détail des tokens, du coût et du cache de la session (remplace /cost et /stats).

/copy

Copie la dernière réponse (ou la Nième) dans le presse-papiers.

/focus

Bascule en plein écran sans distraction.

/todos

Affiche la liste des tâches que l'agent suit actuellement.



Configuration et réglages

/config

Affiche ou règle des paramètres persistants (enregistrés dans `~/.claude/settings.json`).

/model

Change de modèle en cours de session (Opus, Sonnet, Haiku).

/effort

Règle le niveau d'effort (low, medium, high, xhigh, max) ; sans argument, curseur interactif.

/fast

Active ou coupe le mode rapide, qui privilégie la vitesse de réponse.

/permissions

Affiche et règle ce que Claude peut faire : écriture de fichiers, shell, réseau.

/theme

Crée et change de thème de couleurs ; une option « Auto » suit votre terminal.



Configuration et réglages (suite)

/color

Change la couleur de la barre de saisie (pratique avec plusieurs onglets ouverts).

/keybindings

Personnalise les raccourcis clavier selon vos habitudes.

/hooks

Inspecte et gère les hooks : actions automatiques déclenchées sur des événements.



Mémoire et projet

/init

Génère un fichier CLAUDE.md pour ce dépôt : règles, stack, commandes du projet.

/memory

Édite vos fichiers de mémoire, globale et propre au projet.

#<texte>

Ajoute un fait à la mémoire à la volée, sans quitter la saisie.

/add-dir

Donne à Claude l'accès à un dossier supplémentaire.



Travail, qualité et extensions

/review

Lance une revue de code sur la branche courante.

/security-review

Réalise un audit de sécurité des changements en attente.

/pr

Génère une pull request à partir du travail en cours.

/commit

Indexe et committe avec un message généré automatiquement.

/bug

Signale un bug à Anthropic avec le contexte de la session.

/mcp

Liste, ajoute ou retire des serveurs MCP (bases de données, API, services externes).

/plugins

Affiche les plugins installés et les commandes qu'ils ajoutent.



FAMILLE 06 · COMMANDES

Travail, qualité et extensions (suite)

/agents

Gère les sous-agents spécialisés.

/help

Affiche toutes les commandes disponibles.



Au clavier, dans le terminal

Enter	Envoyer le prompt
Shift+Enter	Insérer un saut de ligne
Ctrl+C	Interrompre (deux fois : quitter)
Ctrl+D	Quitter quand la saisie est vide
Ctrl+L	Effacer l'écran, garder l'historique
Ctrl+R	Rechercher dans les prompts passés
Flèches ↑ / ↓	Parcourir l'historique des prompts
Tab	Auto-compléter commandes, fichiers, @
Esc	Annuler la saisie / fermer un volet
Esc Esc	Ouvrir le sélecteur d'historique
@	Mentionner un fichier (chemins auto)
!	Lancer une commande bash ponctuelle
?	Afficher l'aide en surimpression



Nourrir et encadrer Claude

Les mentions « @ » (charger du contexte)

<code>@fichier.md</code>	Inclure un fichier dans le contexte
<code>@dossier/</code>	Inclure tout un dossier
<code>@https://url</code>	Récupérer une page web (lecture seule)

Les modes de permission

<code>default</code>	Demande avant chaque action risquée
<code>acceptEdits</code>	Édits de fichiers auto, shell sur demande
<code>plan</code>	Lecture seule : Claude regarde, ne modifie pas
<code>bypassPermissions</code>	Aucune question (CI en bac à sable seulement)

On bascule en cours de route avec `/permissions`, ou au lancement avec `--permission-`



Lancer Claude Code

`claude`

Démarrer une session interactive ici

`claude "corrige le bug"`

One-shot : exécute puis rend la main

`claude -c`

Reprendre la dernière session

`claude -r <id>`

Reprendre une session précise

`claude --model opus`

Choisir le modèle (opus, sonnet, haiku)

`claude --print "..."`

Mode non interactif, sortie standard

`claude --permission-mode plan`

Démarrer en lecture seule

`claude --add-dir ../autre`

Autoriser un dossier voisin

`claude mcp add <nom> <cmd>`

Installer un serveur MCP

`cat err.log | claude "?"`

Envoyer un fichier par un tube (pipe)



MERCI · À VOUS DE JOUER



Deux mots, un grand merci.

Ces fiches vous sont offertes, sans publicité. Notre seule récompense est votre reconnaissance.

Offrez-nous 30 secondes

Prenez un instant pour nous dire merci sur l'un de nos réseaux.
Un mot public, même court, nous encourage à continuer de vous accompagner, conseil après conseil.

Scannez pour nous rejoindre et nous remercier



WhatsApp



Mattermost



LinkedIn



Facebook