**Cheat Sheet – Google Apps Script & CLASP (Windows 7)**

*Version imprimable • Généré le 24/08/2025 à 00:38*

**Prérequis conseillés pour Windows 7 (compatibilité)**

• Visual Studio Code : utilisez la version 1.70.3 (dernière version officiellement compatible Windows 7).

• Node.js : privilégiez une version 12.x ou 13.x (dernières versions compatibles/encore installables sous Windows 7).

• NPM : inclus avec Node.js. Installez ensuite CLASP globalement.

|  |  |
| --- | --- |
| Commande | Description |
| node -v / npm -v | Vérifie les versions installées. |
| npm i -g @google/clasp | Installe CLASP (Apps Script CLI) globalement. |
| clasp -v | Vérifie la version de CLASP. |

**I. CLASP – Commandes (triées du plus courant → avancé)**

**Très courantes**

|  |  |  |
| --- | --- | --- |
| Commande | Ce que ça fait | Exemple / remarques |
| clasp login | Se connecter à Google (ouvre le navigateur). |  |
| clasp create --type sheets --title "Mon Projet" --parentId <ID\_Dossier> | Créer un nouveau projet Apps Script (lié à Sheets/Docs/Slides/Forms/standalone). | Types valides : standalone, docs, sheets, slides, forms. |
| clasp clone <scriptId> | Cloner un projet Apps Script existant (via son Script ID). |  |
| clasp pull | Télécharger les fichiers du projet depuis Apps Script vers le dossier local. |  |
| clasp status | Lister les différences (local ↔ distant) avant un push. |  |
| clasp push | Envoyer les fichiers locaux vers Apps Script. | Option utile : --force pour forcer l'écriture. |
| clasp open | Ouvrir le projet dans l’éditeur Apps Script (navigateur). |  |

**Fréquentes**

|  |  |  |
| --- | --- | --- |
| Commande | Ce que ça fait | Exemple / remarques |
| clasp logs --watch | Afficher les journaux (logs) en continu. |  |
| clasp logs --open | Ouvrir la console des logs dans le navigateur. |  |
| clasp version "Description" | Créer une nouvelle version du projet (numérotée). | Retourne le numéro de version créé. |
| clasp deploy -d "Description" | Créer un déploiement (ou en créer un nouveau si aucun n’existe). | On peut aussi préciser la version : -V <version>. |
| clasp deployments | Lister les déploiements existants du projet. |  |
| clasp redeploy -i <deploymentId> -V <version> -d "Desc" | Ré-déployer sur un déploiement existant avec une version précise. |  |
| clasp undeploy <deploymentId> | Supprimer un déploiement existant. |  |
| clasp run 'maFonction' --params '["arg1","arg2"]' | Exécuter une fonction Apps Script à distance (Execution API). | Sur Windows, utiliser des quotes simples autour des paramètres JSON. |

**Occasionnelles**

|  |  |  |
| --- | --- | --- |
| Commande | Ce que ça fait | Exemple / remarques |
| clasp list | Lister vos projets Apps Script accessibles. |  |
| clasp apis list | Lister les APIs Google activables pour le projet GCP lié. |  |
| clasp apis enable "drive.googleapis.com" | Activer une API pour le projet. |  |
| clasp apis disable "drive.googleapis.com" | Désactiver une API. |  |
| clasp logout | Se déconnecter de Google dans CLASP. |  |
| clasp help [commande] | Aide intégrée sur une commande. |  |

**Avancées (selon besoins)**

|  |  |  |
| --- | --- | --- |
| Commande | Ce que ça fait | Exemple / remarques |
| clasp login --creds chemin\oauth\_client.json | Se connecter en utilisant vos propres identifiants OAuth (projet Google Cloud). | Pratique pour 'clasp run' et l’Execution API. |
| clasp push --watch | Surveille les fichiers locaux et pousse automatiquement les changements. |  |
| clasp status --json | Sortie JSON des différences (utile en CI). |  |

**II. Fichiers de configuration utiles**

• .clasp.json : paramètres du projet local (ex. scriptId, rootDir, fileExtension, parentId).

• .clasp.ignore : motifs de fichiers/dossiers à ignorer lors du push (similaire à .gitignore).

|  |  |
| --- | --- |
| Fichier | Exemple minimal |
| .clasp.json | {  "scriptId": "1AbcDEF…XYZ",  "rootDir": "src",  "fileExtension": "gs" } |
| .clasp.ignore | # Ignorer node\_modules et fichiers temporaires node\_modules/ \*.map .DS\_Store |

**III. Mini‑workflows prêts à l’emploi**

**A. Lier un projet à un Google Sheet et le déployer**

|  |  |  |
| --- | --- | --- |
| Étape | Commande | But de |
| 1 | clasp login | Connexion à Google. |
| 2 | clasp create --type sheets --title "Automatisation" --parentId <ID\_Dossier> | Créer le projet lié à Sheets dans un dossier Drive. |
| 3 | (Éditer vos fichiers sous VS Code) | Travail local dans votre dossier. |
| 4 | clasp push | Envoyer le code vers Apps Script. |
| 5 | clasp version "v1 initiale" | Créer une version numérotée. |
| 6 | clasp deploy -V <numVersion> -d "Déploiement v1" | Déployer (web app/add-on selon votre manifest). |
| 7 | clasp logs --watch | Surveiller les logs pendant vos tests. |

**B. Mettre à jour un déploiement existant (redeploy)**

|  |  |  |
| --- | --- | --- |
| Étape | Commande | But |
| 1 | clasp push | Pousser les modifications. |
| 2 | clasp version "Correction bugs" | Créer une nouvelle version. |
| 3 | clasp deployments | Récupérer le deploymentId cible (ex: web app publiée). |
| 4 | clasp redeploy -i <deploymentId> -V <version> -d "Maj" | Mettre à jour ce même déploiement. |

**IV. Astuces spécifiques Windows 7**

• VS Code : préférez la version 1.70.3. Évitez d’activer l’auto‑mise à jour vers des versions plus récentes non compatibles.

• Node.js : si l’installateur récent refuse Windows 7, installez Node 12.x ou 13.x (puis NPM mettra à jour les paquets, y compris CLASP).

• Guichets Google (navigateur) : utilisez un navigateur moderne à jour autant que possible (Chrome/Firefox ESR) pour l’authentification OAuth.

Astuce : vous pouvez créer des 'Tasks' dans VS Code pour déclencher rapidement 'clasp push' ou 'clasp logs --watch' via un raccourci clavier.