

GUIDE D'INSTALLATION, DE CONFIGURATION ET D'UTILISATION

Skill WLangage pour Claude Code

Le WLangage de WINDEV, WEBDEV et WINDEV Mobile 2025 dans
Claude Code : un dictionnaire complet pour ne plus jamais inventer
une fonction, un viewer aux couleurs de l'éditeur, et des outils pour
lire, modifier et recompiler vos projets WINDEV.

Gratuit

WINDEV · WEBDEV · WINDEV Mobile 2025

Dictionnaire complet

Projet WINDEV

Annulation

Sommaire

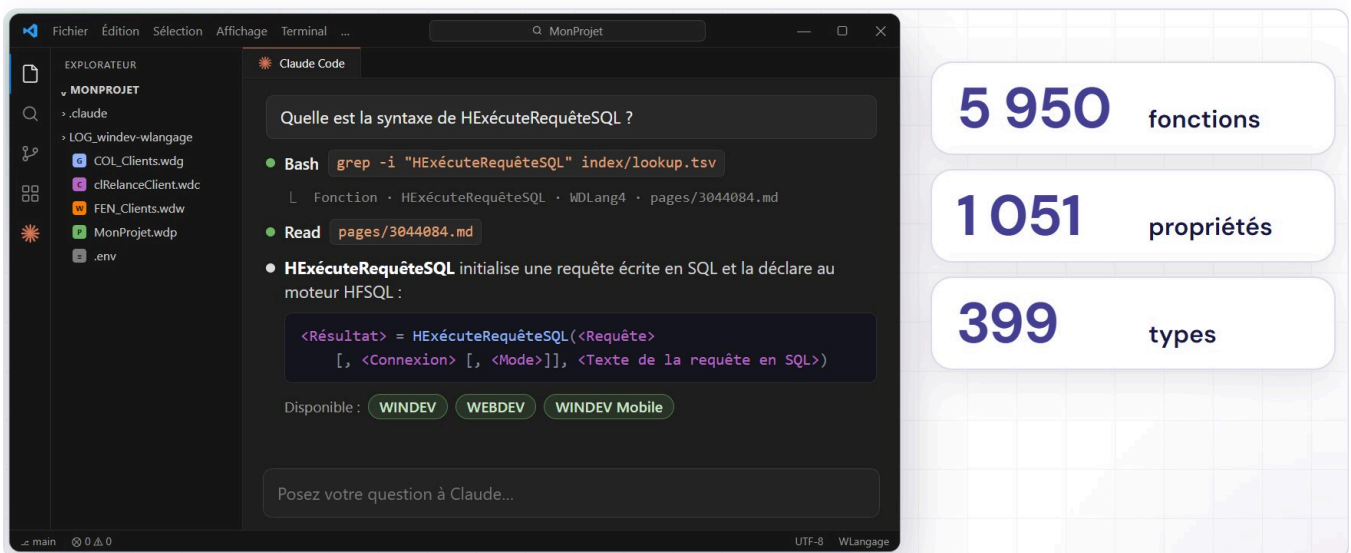
1. Présentation	3
Ce que fait le skill	3
Ce que contient le skill	4
Prérequis	4
2. Installer le skill	5
Télécharger	5
Copier le dossier dans Claude Code	5
Vérifier l'installation	5
Mettre à jour une version précédente	5
3. Configurer	6
Rien d'autre à installer	6
Ouvrir votre projet WINDEV dans Claude Code	6
Les raccourcis du skill dans WINDEV	6
Format binaire ou format texte	6
4. Utiliser au quotidien	8
Interroger le dictionnaire	8
Faire écrire du code contrôlé	8
Lire et copier le code dans le viewer	8
Travailler directement dans le projet	10
Le pilotage de l'IDE	11
Projets sous GDS	11
Annuler une intervention	11
5. Bonnes pratiques et limites	12
6. Dépannage	13
7. Licence et assistance	14

1. Présentation

Ce que fait le skill

Un *skill* est un dossier de connaissances et d'outils que Claude Code charge quand le sujet s'y prête. Le skill `blvtech-wlangage` s'active dès que vous parlez WLangage, WINDEV, WEBDEV ou HFSQL. Il permet à Claude :

- de **répondre juste** sur le WLangage 2025 : syntaxe, paramètres, valeurs de retour, plateformes, à partir d'un dictionnaire complet issu de la documentation de PC SOFT — sans inventer de fonction ;
- d'**écrire du code contrôlé** : chaque identifiant est vérifié dans le dictionnaire, le code est compilé hors de l'éditeur, les fichiers et rubriques HFSQL sont vérifiés dans l'analyse ;
- de vous **montrer le code dans un viewer** aux couleurs de l'éditeur, rangé par élément, avec un bouton pour copier le nom ou le code de chaque procédure ;
- de **travailler directement dans votre projet WINDEV** (nouveau de la version 2) : lire et modifier collections de procédures, classes et fenêtres, créer des éléments, fermer, rouvrir et recompiler le projet, lire la liste des erreurs — avec une copie de sauvegarde avant chaque écriture et une annulation en une commande.



The screenshot shows the VS Code interface with the Claude Code extension. The left sidebar shows the Explorer with the 'MONPROJET' folder. The main editor area displays the Claude Code interface. The search query is 'Quelle est la syntaxe de HExécuteRequêteSQL ?'. The results show a 'Bash' command: `grep -i "HExécuteRequêteSQL" index/lookup.tsv`. Below this, it shows the 'Fonction' 'HExécuteRequêteSQL' from 'WDLang4' in 'pages/3044084.md'. The 'Read' section shows the file 'pages/3044084.md'. The 'HExécuteRequêteSQL' function is described as initializing a request written in SQL and declaring it to the HFSQL engine. The syntax is shown as: `<Résultat> = HExécuteRequêteSQL(<Requête> [, <Connexion> [, <Mode>]], <Texte de la requête en SQL>)`. The 'Disponible' section lists 'WINDEV', 'WEBDEV', and 'WINDEV Mobile'. The bottom of the interface shows a text input field 'Posez votre question à Claude...'. To the right of the screenshot, a summary box lists: 5 950 fonctions, 1 051 propriétés, and 399 types.

Une question dans VS Code : Claude cherche dans le dictionnaire et répond avec la syntaxe exacte.

Ce que contient le skill

	Contenu
Dictionnaire	11 666 rubriques : 5 950 fonctions, 1 051 propriétés, 399 types, 414 exemples, 148 syntaxes préfixées, 76 fonctions de langage externe, 31 mots réservés ; renvois croisés navigables
Recherche	un index d'une ligne par rubrique (nom français ou anglais), index des signatures, familles, propriétés, types, constantes, aide-mémoire
Écriture de code	<code>GUIDE.md</code> : charte, formatage, modèles vérifiés (HFSQL, REST / JSON, e-mail, socket...) et pièges du langage relevés en compilation et en exécution
Contrôles	identifiants inventés, globales mal préfixées, compilation hors IDE, fichiers et rubriques HFSQL de l'analyse
Viewer	coloration de l'éditeur (sombre / clair), explorateur par élément, copie du nom ou du code de chaque procédure, export <code>.wl</code>
Projet WINDEV	lecture et écriture des <code>.wdg</code> , <code>.wdc</code> , <code>.wdw</code> , <code>.wde</code> et des projets texte ; création de classes, collections, procédures globales, fenêtres ; pilotage de l'IDE ; GDS ; sauvegarde et annulation

Prérequis

Pour...	Il faut
le dictionnaire et le guide d'écriture	Claude Code (extension VS Code, application de bureau ou terminal) — rien d'autre
le viewer et les contrôles du code	rien à préparer : Claude installe lui-même, avec votre accord, le seul composant nécessaire (Python, gratuit) s'il manque
le travail dans vos projets WINDEV	Windows et WINDEV 2025 installé

WINDEV 2025

Le dictionnaire et les outils visent la version 2025 de WINDEV, WEBDEV et WINDEV Mobile. Les fichiers de projet d'une version antérieure ne sont pas pris en charge par les outils d'écriture.

2. Installer le skill

Télécharger

- 1 Sur www.blv-tech.com/skill-claude/windev-wlangage, saisissez votre adresse e-mail, puis le **code à 6 chiffres** reçu (valable 30 minutes).
- 2 Téléchargez `blvtech-wlangage.zip` (environ 20 Mo), puis : clic droit › Propriétés › cochez **Débloquer** › OK (fichiers venus d'Internet).
- 3 Décompressez l'archive : vous obtenez un dossier `blvtech-wlangage` qui contient `SKILL.md`.

Copier le dossier dans Claude Code

Copiez le dossier `blvtech-wlangage` (pas son contenu) dans l'un de ces emplacements :

Portée	Emplacement
Tous vos projets (recommandé)	Windows : <code>%USERPROFILE%\claude\skills\blvtech-wlangage\</code> macOS / Linux : <code>~/claude/skills/blvtech-wlangage/</code>
Un seul projet (partagé avec l'équipe)	<code><projet>\claude\skills\blvtech-wlangage\</code>

Le plus simple sous Windows

Touche **Windows** + **R**, tapez `%USERPROFILE%\claude`, Entrée ; créez le dossier `skills` s'il n'existe pas, ouvrez-le et glissez-y le dossier `blvtech-wlangage`.

Contrôle : le fichier `%USERPROFILE%\claude\skills\blvtech-wlangage\SKILL.md` doit exister — et non `...\blvtech-wlangage\blvtech-wlangage\SKILL.md` (dossier copié une fois de trop).

Vérifier l'installation

- 1 Redémarrez Claude Code, ou ouvrez une nouvelle conversation.
- 2 Tapez `/` : `blvtech-wlangage` apparaît dans la liste (ou demandez « quels skills as-tu ? »).
- 3 Posez une question : « *quelle est la syntaxe de HExécuteRequêteSQL ?* ». Claude ouvre la rubrique du dictionnaire et vous répond. Le skill s'active tout seul dès que vous parlez WLangage, WINDEV ou HFSQL, ou à la demande avec `/blvtech-wlangage`.

Mettre à jour une version précédente

Supprimez l'ancien dossier `windev-wlangage` (nom du skill jusqu'à la version 2.0), puis copiez le nouveau dossier `blvtech-wlangage` — ne gardez pas les deux, ne les fusionnez pas : le skill se déclencherait en double et les deux : des fichiers ont été renommés. Les sauvegardes `LOG_windev-wlangage` de vos projets ne sont pas concernées.

Désinstaller

Supprimez le dossier `blvtech-wlangage` des skills. Les dossiers `LOG_windev-wlangage` créés dans vos projets peuvent être supprimés quand vous n'en avez plus besoin.

3. Configurer

Le dictionnaire fonctionne sans aucun réglage. Les quelques points ci-dessous concernent le viewer, les contrôles et le travail dans les projets WINDEV.

Rien d'autre à installer

Vous n'avez rien à préparer. Les outils du skill s'appuient sur **Python**, un logiciel gratuit : s'il n'est pas déjà sur votre poste, Claude vous le signale et, si vous acceptez, l'installe lui-même (sans droits d'administration). Aucun module ni extension supplémentaire n'est nécessaire, et le reste est fourni avec Windows.

Le **viewer** s'ouvre tout seul quand Claude affiche du code, à l'adresse `http://localhost:8753`.

Juste après l'installation de Python

Si Claude vient d'installer Python, redémarrez VS Code (ou Claude Code) à la fin de la séance pour que tout le reste le trouve directement.

Ouvrir votre projet WINDEV dans Claude Code

Claude travaille sur les fichiers du dossier ouvert. Ouvrez dans VS Code (ou lancez Claude Code dans) le **dossier de votre projet WINDEV**, celui qui contient le fichier `.wdp`. Claude y trouve les collections (`.wdg`), classes (`.wdc`), fenêtres (`.wdw`), états (`.wde`) et l'analyse.

Travaillez sur un projet sauvegardé

Les outils écrivent dans les fichiers de votre projet. Même s'ils gardent une copie de chaque fichier avant de le modifier et refusent d'écrire au moindre doute, gardez votre projet sous GDS, sous Git, ou faites-en une copie avant une grosse intervention.

Les raccourcis du skill dans WINDEV

Pour piloter l'IDE (fermer le projet, le recompiler, importer des éléments), le skill utilise trois raccourcis clavier. Au premier pilotage, il les installe lui-même dans WINDEV — WINDEV fermé, après une copie de sauvegarde de vos réglages :

Raccourci	Commande de WINDEV
<code>Ctrl + Alt + Maj + E</code>	Recompilation du projet
<code>Ctrl + Alt + Maj + F</code>	Fermer le projet
<code>Ctrl + Alt + Maj + W</code>	Importer des éléments

Vous pouvez aussi les définir vous-même dans WINDEV (« Personnaliser les raccourcis »). Pendant le pilotage, la session Windows doit rester **ouverte et déverrouillée**.

Format binaire ou format texte

WINDEV enregistre les éléments d'un projet soit en **binaire** (par défaut), soit au **format texte** (option du projet « format de sauvegarde des éléments : Texte »).

- Au **format texte**, tous les éléments sont lus et réécrits par le skill : c'est le format le plus sûr pour travailler avec Claude (et le plus lisible dans Git).
- En **binaire**, la plupart des fenêtres et des classes sont prises en charge, une partie seulement des collections de procédures (mesure sur les exemples de WINDEV 2025 : fenêtres 85 / 88, classes 12 / 12, collections 7 / 12). Quand un fichier n'est pas reconnu, l'outil s'arrête **sans rien écrire** et Claude passe par la création dans l'IDE ou par le viewer.

Le passage au format texte est un choix d'équipe : Claude ne le fait jamais sans votre accord.

4. Utiliser au quotidien

Interroger le dictionnaire

Posez vos questions comme à un collègue qui connaît la documentation par cœur :

- « Quelle est la syntaxe de HExécuteRequêteSQL ? Est-ce disponible sous Android ? »
- « Quelles propriétés permettent de griser un champ ? »
- « Comment parcourir un tableau associatif ? Montre-moi un exemple. »
- « Quelle est la différence entre JSONVersVariant et Déséréalise ? »

Claude cherche la rubrique (nom français ou anglais), la lit et vous répond avec la syntaxe exacte, les paramètres, les plateformes et, au besoin, un exemple. **Il ne cite que des fonctions qui existent** : en cas de doute, il cherche au lieu de deviner.

Faire écrire du code contrôlé

« Écris une procédure globale qui charge les clients actifs d'une ville. » Avant de vous montrer le code, Claude le fait passer par les contrôles du skill :

vous

Écris une procédure qui charge les clients actifs d'une ville.

VÉRIFICATIONS AUTOMATIQUES

4 / 4 ✓

✓

verif_identifiants.py

— 0 identifiant inconnu

✓

verif_globales.py

— globales préfixées

✓

verif_compilation.py

— compilé hors IDE, 0 erreur

✓

analyse.py

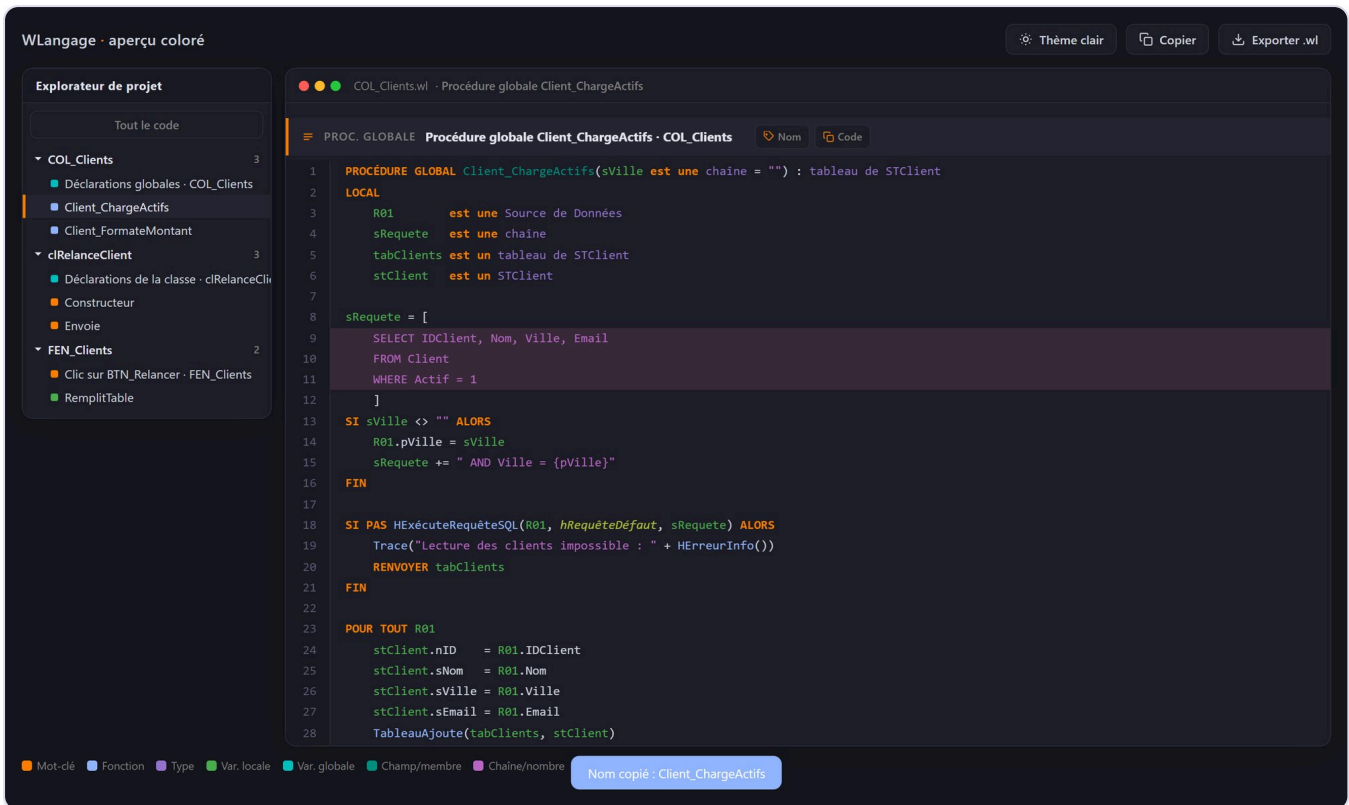
— rubriques HFSQL vérifiées

Les contrôles automatiques avant livraison.

Contrôle	Ce qu'il vérifie
verif_identifiants.py	toute fonction, constante ou propriété absente du dictionnaire est signalée
verif_compilation.py	compilation hors de l'éditeur par WDScriptConsole : syntaxe, signatures, types
verif_globales.py	les variables globales d'une autre collection sont correctement préfixées
analyse.py	les fichiers et rubriques HFSQL utilisés existent dans l'analyse du projet
verif_code.py	continuations interdites, variables jamais utilisées, lignes orphelines

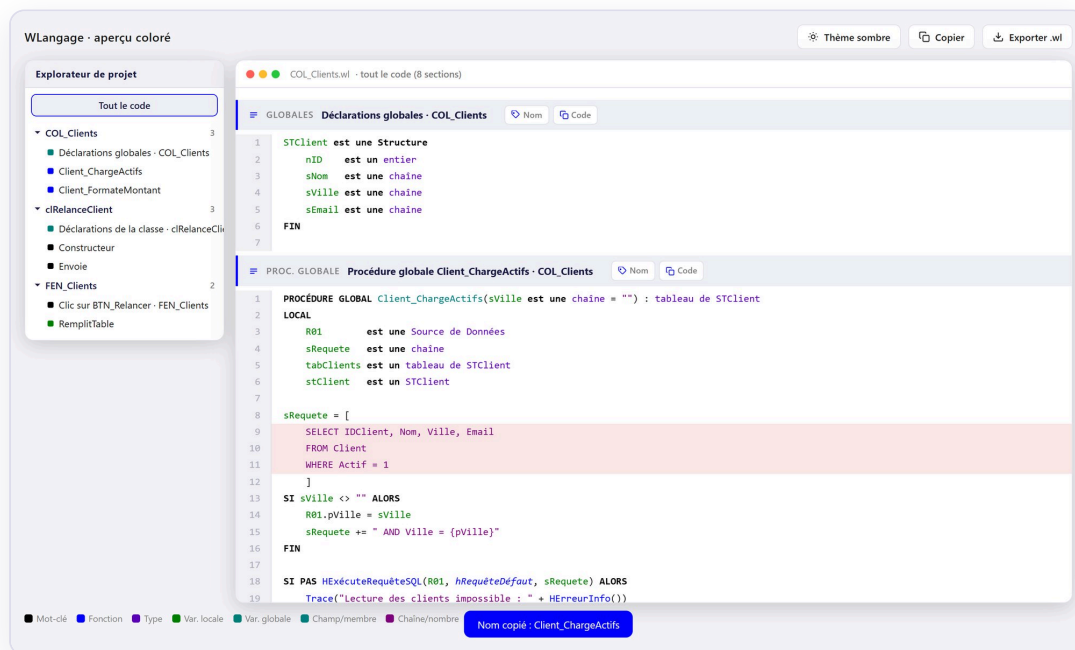
Lire et copier le code dans le viewer

Tout code WLangage que Claude vous présente s'affiche dans le **viewer**, sur `http://localhost:8753`, aux couleurs de l'éditeur de WINDEV.



Le viewer : explorateur par élément, section affichée seule, bouton « Nom » qui vient de copier le nom de la procédure.

- **Explorateur** (à gauche) : le code est rangé par élément (collection, classe, fenêtre), puis par procédure, avec une pastille de couleur par genre. « Tout le code » ou un clic sur une seule procédure.
- **Nom** (sur chaque section) : copie le nom de la procédure, prêt à coller dans WINDEV.
- **Code** : copie tout le code de la section, ligne `PROCÉDURE` comprise.
- **Copier, Exporter .wl**, clic droit › Copier : tout le code ou la sélection.
- **Thème clair / sombre**, numéros de ligne par section, alignement automatique des déclarations.



Le même code en thème clair.

Travailler directement dans le projet

Demandez une modification comme vous la décririez à un développeur de l'équipe :

- « Lis la fenêtre FEN_Article et explique ce que fait le bouton Enregistrer. »
- « Ajoute une procédure globale Outils_FormateMontant à COL_Outils. »
- « Crée une classe clRelanceClient avec un constructeur et une méthode Envoie. »
- « Branche le bouton BTN_Relancer de FEN_Clients sur la nouvelle classe. »
- « Corrige l'erreur de compilation de FEN_Devis et recompile le projet. »



Une demande, trois éléments touchés : procédure ajoutée, classe créée, événement modifié.

À chaque intervention, Claude suit toujours la même séquence :

- 1 **Contrôle le code** à écrire (aucune écriture tant qu'il reste une erreur).
- 2 **Libère le projet** : arrête un test en cours, enregistre tout et ferme le projet dans WINDEV (WINDEV reste ouvert). S'il rencontre une boîte de dialogue ou un menu ouvert, il s'arrête et vous prévient — il ne force jamais la fermeture.
- 3 **Sauvegarde chaque fichier** dans `LOG_windev-wlangage\JJ-MM-AAAA\`, copie vérifiée octet à octet et notée dans un journal.
- 4 **Écrit** dans un fichier temporaire, vérifie qu'il est conforme à l'octet près, puis remplace l'original.
- 5 **Rouvre le projet** sur les seuls éléments modifiés, le **recompile** et lit la liste des erreurs.

Les éléments neufs (classe, collection) sont fabriqués par WINDEV lui-même à partir d'un fichier texte, puis déclarés dans le projet : aucun fichier « à importer à la main ».

Le pilotage de l'IDE



Fermeture, sauvegarde, réouverture, recompilation, lecture des erreurs.

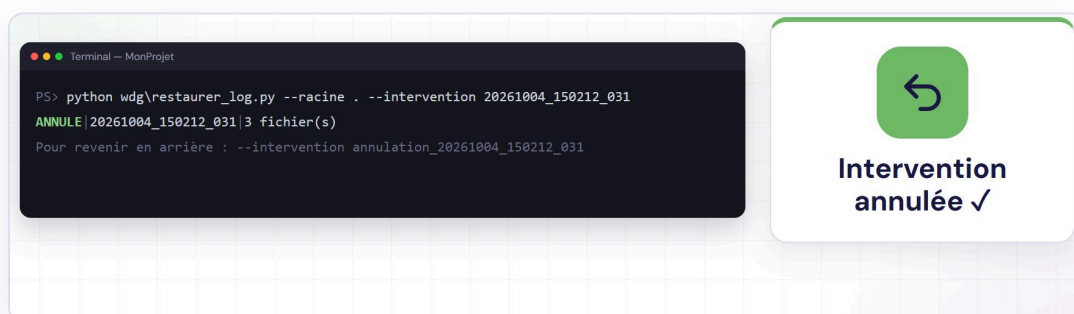
- Claude peut aussi **lancer un test** (F9 ou Ctrl+F9) et récupérer les erreurs d'exécution (message, pile, code) — jamais sur un élément qui écrit dans une base de production ou appelle une API réelle sans votre accord.
- Pendant le pilotage, **ne touchez ni au clavier ni à la souris** dans WINDEV et gardez la session déverrouillée : le skill envoie de vraies frappes à l'éditeur.

Projets sous GDS

Un fichier en lecture seule est un élément **non extrait**. Claude extrait les éléments à modifier (projet laissé ouvert), écrit, recompile et vérifie l'état local et serveur. Il ne lève jamais l'attribut lecture seule d'un fichier pour contourner le GDS, et **la réintégration vous revient**.

Annuler une intervention

« *Annule ta dernière intervention.* » Projet fermé, Claude remet la première copie de chaque fichier modifié, retire les éléments qu'il avait créés (déplacés dans le dossier de sauvegarde, jamais effacés) et sauvegarde l'état courant avant : l'annulation est elle-même annulable.



Une intervention annulée en une commande.

Vous pouvez aussi demander « *Montre-moi la liste de tes interventions* », puis « *Annule celle de ce matin* » : Claude simule d'abord l'annulation et vous dit ce qui sera remis en place.

Un élément créé puis retiré reste déclaré dans le projet : retirez-le dans l'éditeur (Claude vous le rappelle).

5. Bonnes pratiques et limites

Bonnes pratiques

- 1 Projet **sauvegardé** (GDS, Git ou copie) avant toute intervention importante.
- 2 Une demande à la fois, puis relecture dans le viewer et dans l'éditeur.
- 3 **Testez** le code avant de le livrer : les contrôles évitent les erreurs de syntaxe et les fonctions inventées, pas les erreurs de logique métier.
- 4 Préférez le **format texte** pour les projets sur lesquels Claude travaille souvent.
- 5 Fermez les boîtes de dialogue de WINDEV avant de demander une modification du projet.

Limites connues

Sujet	Limite
États (<code>.wde</code>)	lecture du code validée ; écriture non prise en charge
Requêtes (<code>.WDR</code>)	non prises en charge
Classes	seules les classes WINDEV 2025 « pures » sont modifiables ; les autres variantes sont lues mais refusées en écriture
Collections binaires	une partie seulement est modifiable ; une collection neuve se crée par l'IDE (le skill s'en charge)
Bloc de déclarations	jamais édité dans l'IDE : le créer d'abord dans l'éditeur
Anciennes versions	fichiers d'une version antérieure à WINDEV 2025 non pris en charge

Dans tous ces cas, rien n'est écrit : Claude affiche le code dans le viewer et vous indique précisément ce qui reste à faire à la main.

6. Dépannage

Symptôme	Cause probable et solution
Le skill n'apparaît pas avec <code>/</code>	Dossier mal placé (vérifier <code>...\claude\skills\blvtech-wlangage\SKILL.md</code>) ou Claude Code non redémarré.
Claude répond sans le dictionnaire	Mentionner « WLangage » ou « WINDEV » dans la demande, ou l'appeler par <code>/blvtech-wlangage</code> .
« Python introuvable »	Accepter l'installation proposée par Claude, puis redémarrer VS Code. Sur un poste verrouillé : installer Python depuis python.org en cochant « Add python.exe to PATH ».
Le viewer ne s'affiche pas	Ouvrir <code>http://localhost:8753</code> et recharger la page si elle était déjà ouverte. Si un autre logiciel occupe le port 8753, demander à Claude de lancer le viewer sur un autre port.
Le pilotage de WINDEV échoue	Session verrouillée, boîte de dialogue ouverte, menu déroulé ou test en cours : fermer, puis relancer la demande. Claude ne force jamais la fermeture.
« zone inattendue », « Decoupe »	Structure de fichier binaire non reconnue : rien n'a été écrit. Passer par l'IDE ou le format texte, et signaler le cas (avec la version de WINDEV) via la page du skill.
« Recompile le projet » plante après un import	Dans WINDEV : « Réparer le projet », puis recompiler.
Une modification ne convient pas	Demander l'annulation de la dernière intervention (voir Annuler).

7. Licence et assistance

Le skill est gratuit, pour un usage personnel ou professionnel ; vous pouvez le modifier pour votre propre usage. Il est fourni « en l'état », sans garantie : les outils lisent et écrivent les fichiers de vos projets et pilotent l'IDE, travaillez sur un projet sauvegardé et testez le code avant de le livrer. Conditions complètes : [LICENCE.md](#).

Merci de ne pas diffuser l'archive vous-même : renvoyez vers la page officielle, où se trouvent les mises à jour.

WINDEV, WEBDEV, WINDEV Mobile, WLangage et HFSQL sont des marques de PC SOFT. Ce skill est un projet indépendant, ni affilié ni approuvé par PC SOFT ; le dictionnaire est une mise en forme de la documentation du WLangage destinée à la consultation par l'assistant, la documentation officielle de PC SOFT fait foi. Claude et Claude Code sont des marques d'Anthropic.



Ce guide vous a été utile ? Le skill est gratuit : un don sur sa page aide à le faire vivre.

www.blv-tech.com/skill-claude/windev-wlangage · développement WINDEV sur mesure, intégration de l'IA dans vos applications : www.blv-tech.com