

■ KCG Xojo API Guard – User Guide

Overview

KCG Xojo API Guard scans Xojo source for deprecated API 1.0 usage and suggests API 2.0 equivalents. It supports text-based Xojo projects (.xojo_window, .xojo_code, .xojo_project) and free-form snippets via the clipboard. Results appear in-app and can be exported as CSV for triage.

Key Features

- File scan: browse to a .xojo_window / .xojo_code / .xojo_project and analyze it.
- Clipboard scan: analyze text from the system clipboard (code, snippets, or pasted file content).
- Precise locations: shows raw file line and code line numbers.
- Context capture: outputs class/window name and owner.method for events/handlers (e.g., winMain.KeyDown or SolveButton.Pressed).
- Common API 1.0 → API 2.0 findings (Window → DesktopWindow, MsgBox → MessageBox, Invalidate() → Refresh(True), etc.).
- Save as CSV: exports structured results for spreadsheets and tracking.
- Copy Results: copy the formatted text report to your clipboard.

Getting Started

- Launch the application.
- Click **Browse...** and choose a Xojo text-project file, or click **Scan Clipboard** to analyze text from the clipboard.
- Click **Process File** (for file scans). For clipboard scans, results appear immediately.
- Review the findings in **Results Output**.
- Use **Save as CSV** to export, or **Copy Results** to copy the report text.

Controls

Control	Function
Browse	Select a file to scan (.xojo_window, .xojo_code, .xojo_project).
Input File	Shows the selected file path, or "(Clipboard)" for clipboard scans.
Process File	Runs the scan for the selected file and shows results.
Scan Clipboard	Reads plain text from the system clipboard and scans it.
Results Output	Displays human-readable findings: line numbers, method context, token, and hint.
Save Output	Saves the contents of the Results Output panel as text.
Copy Results	Copies the Results Output to the clipboard.
Save as CSV	Exports results as CSV with columns: file, class, raw_line, code_line, method, token, hint.

CSV Output

CSV export includes the following columns:

Column	Description
file	The scanned source (path or source label).

class	Detected window/class name (from file name or “Begin DesktopWindow ...”).
raw_line	Line number in the raw text file.
code_line	Line number within the current code block, excluding #tag lines.
method	Owner + event/method (e.g., winMain.KeyDown, SolveButton.Pressed).
token	The matched API 1.0 symbol/text.
hint	Suggested API 2.0 replacement or guidance.

Example rows:

file,raw_line,code_line,method,token,hint

C:\MyProject\winMain2.xoyo_window,777,6,winMain2.KeyDown,Invalidate(),Use Refresh(True)

C:\MyProject\winMain2.xoyo_window,2137,923,winMain2.Opening,Window,Use DesktopWindow

What the Scanner Looks For

- UI Classes: Window → DesktopWindow, PushButton → DesktopButton, ListBox → DesktopListBox, TextArea → DesktopTextArea, HTMLViewer → DesktopHTMLViewer, Canvas → DesktopCanvas
- Dialogs/Messages: MsgBox → MessageBox(...), ShowModalDialog → MessageDialog.ShowModalWithin(window) or MessageBox
- Events: Open → Opening
- ListBox members: Cell(...) → CellTextAt(row, column), RemoveRow(...) → RemoveRowAt(row), ListIndex → SelectedRowIndex
- Filesystem: AbsolutePath → NativePath, GetOpenFolderItem → SelectFolderDialog, GetOpenFile(Name)? → OpenFileDialog
- Repaint: Invalidate(...) or Invalidate → Refresh(True)

About Line Numbers

raw_line matches your external text editor (e.g., Notepad++). code_line counts inside the current code block (between #tag Method|Event... and #tag End), skipping #tag lines and comment-only lines. This dual numbering helps you jump in either the text file or the Xojo IDE.

Clipboard Scanning

Use Scan Clipboard to analyze any text on your clipboard. The results label the source as “(Clipboard)”. If your snippet contains Xojo text-project tags (#tag Method, #tag Event, etc.), the scanner will honor code blocks and report code_line accordingly. Otherwise, it treats the entire snippet as code.

Troubleshooting

CSV won’t save: Select a writable folder (e.g., Documents). If you pick a folder instead of a filename, the app will use a default name like Results.csv inside it.

No method shown: Some window-level events omit Name=. The app infers the method/event name from the first Sub/Function signature it encounters.