/\*\*

\* ============================================================================

\* @module GAssistant.Tools.CodeReview – v17.0

\* @description

\* واجهة وسيطة بين Sidebar الذكي و وحدة Gemini Script Doctor.

\* تهيئ بيئة المراجعة وتستدعي التحليل البرمجي المخصص.

\* تعتمد على: GAssistant.Tools.GeminiScriptDoctor, GAssistant.Helpers, GAssistant.AI.Memory, GAssistant.Config

\* ============================================================================

\*/

var GAssistant = GAssistant || {};

GAssistant.Tools = GAssistant.Tools || {};

// حقن التبعيات: يجب تمرير الوحدات المعتمدة هنا بأسماء مساحات الأسماء الكاملة

GAssistant.Tools.CodeReview = (function(GA\_GeminiScriptDoctor, GA\_Helpers, GA\_AI\_Memory, GA\_Config) {

'use strict';

/\*\*

\* @function reviewCodeForSidebar

\* @description

\* 🎯 دالة تستدعى من CodeReviewSidebar.html.

\* تستقبل كود نصي وتحلله باستخدام Gemini Script Doctor.

\* @param {string} scriptText – الكود المطلوب مراجعته.

\* @returns {object} – كائن يتضمن: overallScore, summary, feedbackItems[] أو كائن خطأ موحد.

\*/

function reviewCodeForSidebar(scriptText) {

try {

const sessionId = new Date().toISOString();

GA\_AI\_Memory.logOperation('GAssistant.Tools.CodeReview', 'reviewCodeForSidebar:start', { sessionId, length: scriptText.length });

// استدعاء دالة المراجعة من Gemini Script Doctor مباشرة.

// في Apps Script، UrlFetchApp.fetch هو متزامن، لذا لا حاجة لـ .then() / .catch() هنا.

const result = GA\_GeminiScriptDoctor.reviewCode(scriptText); // افتراض أن هذه هي الدالة الصحيحة في GeminiScriptDoctor

if (result?.overallScore === undefined) {

GA\_Helpers.logOperation('GAssistant.Tools.CodeReview', 'reviewCodeForSidebar:incomplete\_response', { result });

throw new Error("رد غير مكتمل من وحدة التحليل.");

}

GA\_AI\_Memory.logOperation('GAssistant.Tools.CodeReview', 'reviewCodeForSidebar:success', { score: result.overallScore });

return result;

} catch (e) {

GA\_Helpers.logOperation('GAssistant.Tools.CodeReview', 'reviewCodeForSidebar:failure', {

error: e.message,

stack: e.stack,

scriptTextPreview: scriptText.substring(0, 100) // تسجيل جزء من الكود للمساعدة في التصحيح

});

// إرجاع كائن خطأ متوافق مع ما يتوقعه الواجهة الأمامية (CodeReviewSidebar.html)

return {

overallScore: 0,

summary: "حدث خطأ أثناء المراجعة: " + e.message,

feedbackItems: [],

error: e.message // إضافة حقل error لتسهيل التعامل في الواجهة الأمامية

};

}

}

// 📦 واجهات التصدير

return {

reviewCodeForSidebar

};

})(

GAssistant.Tools.GeminiScriptDoctor, // افتراض وجود هذه الوحدة

GAssistant.Helpers,

GAssistant.AI.Memory,

GAssistant.Config // تم حقنها هنا، ولكن قد لا تكون ضرورية لهذه الوحدة تحديدًا

);

// لتمكين google.script.run من استدعاء الدالة مباشرة من الواجهة الأمامية (HTML)

// هذه الدالة هي الواجهة العامة لـ reviewCodeForSidebar.

function reviewCodeForSidebar(scriptText) {

return GAssistant.Tools.CodeReview.reviewCodeForSidebar(scriptText);

}