/\*\*

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

\* @file workshop.gs – v12.9

\* @module WORKSHOP\_TOOLS

\* @description

\* أدوات ذكية لورشة عمل المبرمج. تقوم بتحليل الأكواد المقترحة

\* وتقترح الملف الأنسب لوضعها فيه، مما يسهل عملية تطوير وصيانة المشروع.

\* - تستخدم خوارزمية أوزان لتحديد أولوية اقتراح الملف.

\* - تدعم أنماطًا قابلة للتكوين ديناميكيًا من الإعدادات.

\* - تسجل عمليات الاقتراح للمراجعة.

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

\*/

var WORKSHOP\_TOOLS = (function(ENV, DIALOGUE\_TOOLS, SETTINGS\_PROVIDER, HELPERS) {

'use strict';

/\*\*

\* 🧠 [مساعدة] يحلل محتوى الكود ويقترح اسم الملف الأنسب لوضعه فيه.

\* يستخدم نظام أوزان لتحديد الأولوية.

\* @private

\* @param {string} functionCode - الكود البرمجي للدالة المقترحة.

\* @returns {string} - اسم الملف المقترح (مثال: 'tools\_sheets.gs').

\*/

function suggestTargetFile(functionCode) {

const code = (functionCode || '').toLowerCase();

const cfg = SETTINGS\_PROVIDER.getSettings();

// قائمة الأنماط والكلمات المفتاحية المرتبطة بكل ملف، مع أوزان.

// هذه القائمة يتم جلبها الآن ديناميكيًا من SETTINGS\_PROVIDER.

const patterns = cfg.TOOL\_FILE\_SUGGESTION\_PATTERNS || [

// Fallback default patterns if not defined in settings

{ keywords: ['getactivespreadsheet', 'spreadsheetapp', 'createmenu', 'onopen', 'onsidebar'], file: 'ui.gs', weight: 10 },

{ keywords: ['getrange', 'setvalue', 'appendrow', 'sheet', 'range', 'getvalues', 'setvalues'], file: 'tools\_sheets.gs', weight: 9 },

{ keywords: ['revenue', 'expense', 'profit', 'currency', 'income', 'cost', 'ledger'], file: 'tools\_accounting.gs', weight: 9 },

{ keywords: ['driveapp', 'createbackup', 'export', 'pdf', 'mailapp', 'timestamp', 'system', 'config'], file: 'tools\_general.gs', weight: 8 },

{ keywords: ['documentapp', 'getbody', 'appendparagraph', 'docs', 'google docs'], file: 'builtinDocs.gs', weight: 8 }, // For Apps Script built-in docs

{ keywords: ['tools\_catalog', 'tool\_id', 'toolinfo', 'getfunctionlist', 'gettoolmetadata'], file: 'tools\_catalog.gs', weight: 7 }, // For tool definitions

{ keywords: ['ai\_ask', 'gemini', 'generative', 'api', 'fetch', 'llm', 'model'], file: 'ai\_core.gs', weight: 7 },

{ keywords: ['context', 'session', 'remember', 'recall', 'memory'], file: 'ai\_memory.gs', weight: 6 },

{ keywords: ['message', 'show', 'alert', 'dialogue', 'prompt'], file: 'ai\_tools\_dialogue.gs', weight: 6 },

{ keywords: ['test', 'assert', 'expect', 'runtest'], file: 'tests.gs', weight: 10 },

// Add more as needed

];

let bestMatch = { file: 'tools\_general.gs', score: 0 }; // الافتراضي إذا لم يتم العثور على نمط

for (const pattern of patterns) {

const score = pattern.keywords.reduce((acc, kw) => {

// نمنح نقاطًا أعلى للكلمات المفتاحية الأطول أو الأكثر تحديدًا

if (code.includes(kw)) {

// يمكن تعديل هذه المعادلة لتعطي أولوية أكبر للكلمات الأطول

return acc + pattern.weight \* (1 + kw.length / 10); // وزن إضافي بناءً على طول الكلمة

}

return acc;

}, 0);

if (score > bestMatch.score) {

bestMatch = { file: pattern.file, score: score };

}

}

// تسجيل الملف المقترح ودرجة المطابقة

HELPERS.logOperation('WORKSHOP\_TOOLS', 'suggestTargetFile', { suggestedFile: bestMatch.file, score: bestMatch.score, codePreview: functionCode.substring(0, 50) + '...' });

return bestMatch.file;

}

/\*\*

\* 🛠️ [أداة] تقترح وتُجهز كودًا برمجيًا جديدًا في ورقة "ورشة عمل المبرمج".

\* هذه الدالة تستخدم المنطق الذكي لاقتراح الملف الأنسب.

\* @param {string} functionName - اسم الدالة المقترحة.

\* @param {string} description - وصف للميزة الجديدة وماذا تفعل.

\* @param {string} functionCode - الكود البرمجي الكامل للدالة المقترحة.

\* @returns {object} - كائن رسالة توجيه للمستخدم من DIALOGUE\_TOOLS.

\*/

function proposeAndStageCode(functionName, description, functionCode) {

try {

const ss = SpreadsheetApp.getActiveSpreadsheet();

let sheet = ss.getSheetByName(ENV.STAGING\_SHEET\_NAME);

if (!sheet) {

sheet = ss.insertSheet(ENV.STAGING\_SHEET\_NAME, 0);

sheet.getRange("A1:B1").setValues([["🚀 ورشة عمل المبرمج", "هنا يتم اقتراح أكواد جديدة لتوسيع قدرات المساعد"]])

.merge().setFontWeight('bold').setHorizontalAlignment('center');

sheet.setColumnWidths(1, 2, [250, 750]);

HELPERS.logOperation('WORKSHOP\_TOOLS', 'proposeAndStageCode:sheet\_created', { sheetName: ENV.STAGING\_SHEET\_NAME });

}

const lastRow = sheet.getLastRow();

const startRow = lastRow > 1 ? lastRow + 2 : 2;

const suggestedFile = suggestTargetFile(functionCode); // ✨ استدعاء الدالة الذكية

sheet.getRange(startRow, 1, 1, 2).setValues([[`✨ الميزة: ${functionName}`, `📝 الوصف: ${description}`]])

.setFontWeight('bold').setBackground(ENV.PRIMARY\_HEADER\_COLOR);

sheet.getRange(startRow + 1, 1).setValue("📄 الكود المقترح:").setFontWeight('bold');

sheet.getRange(startRow + 1, 2).setValue(functionCode).setFontFamily(SpreadsheetApp.FontFamily.COURIER\_NEW).setWrap(true);

const scriptEditorUrl = `https://script.google.com/home/projects/${ScriptApp.getScriptId()}/edit`; // رابط محرر Apps Script

const steps = [

`1. انسخ الكود أعلاه.`,

`2. افتح محرر Apps Script: ${scriptEditorUrl}`,

`3. الاقتراح: الصق الكود في ملف "${suggestedFile}".`,

`4. أضف اسم الدالة "${functionName}" إلى قائمة AVAILABLE\_TOOLS في ملف tools\_catalog.gs (أو الملف المناسب).`, // تم التعديل ليكون tools\_catalog.gs

`5. احفظ المشروع 💾 وأعد تشغيل الشريط الجانبي لتفعيل الميزة.`

].join('\n');

sheet.getRange(startRow + 2, 1).setValue("✅ خطوات التفعيل:").setFontWeight('bold');

sheet.getRange(startRow + 2, 2).setValue(steps).setWrap(true).setVerticalAlignment("top");

sheet.getRange(startRow, 1, 3, 2).setBorder(true, true, true, true, '#cccccc', SpreadsheetApp.BorderStyle.SOLID);

sheet.autoResizeRows(startRow, 3);

ss.setActiveSheet(sheet);

HELPERS.logOperation('WORKSHOP\_TOOLS', 'proposeAndStageCode:success', { functionName, suggestedFile });

return DIALOGUE\_TOOLS.showMessage("success", `تم تجهيز ورشة العمل. يرجى مراجعة ورقة "${ENV.STAGING\_SHEET\_NAME}".`);

} catch (e) {

HELPERS.logOperation('WORKSHOP\_TOOLS', 'proposeAndStageCode:error', { error: e.message, functionName });

return DIALOGUE\_TOOLS.showError(`فشل في تجهيز ورشة العمل: ${e.message}`);

}

}

/\*\*

\* 🔌 واجهة مساعدة للشريط الجانبي.

\* تُستخدم لاستدعاء suggestTargetFile من واجهة المستخدم (HTML).

\* @param {string} functionCode - كود الدالة المقترحة من المستخدم.

\* @returns {{suggestedFile: string}} - كائن يحتوي على اسم الملف المقترح.

\*/

function suggestFileFromSidebar(functionCode) {

const file = suggestTargetFile(functionCode);

return { suggestedFile: file };

}

// تصدير الدوال العامة لتكون متاحة

return {

suggestTargetFile: suggestFileFromSidebar, // تصدير الواجهة للشريط الجانبي

proposeAndStageCode

};

})(ENV, DIALOGUE\_TOOLS, SETTINGS\_PROVIDER, HELPERS); // التأكد من تمرير جميع التبعيات