/\*\*

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

\* @file analytics\_dashboard.gs – v1.0

\* @module GAssistant.Analytics

\* @description

\* وحدة لوحة التحكم التحليلية. مسؤولة عن جمع وعرض الإحصاءات والبيانات

\* من مختلف وحدات المشروع، بما في ذلك أدوات المحاسبة.

\* تهدف إلى توفير رؤى مرئية للمستخدم.

\* تعتمد على: GAssistant.Helpers, GAssistant.AI.Dialogue, GAssistant.Tools.Accounting

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

\*/

var GAssistant = GAssistant || {};

GAssistant.Analytics = GAssistant.Analytics || {};

GAssistant.Analytics = (function(GA\_Helpers, GA\_AI\_Dialogue, GA\_Tools\_Accounting) {

'use strict';

/\*\*

\* @function generateDashboardSummary

\* @description

\* تُنشئ ملخصًا شاملاً للوحة التحكم، بما في ذلك إحصاءات المحاسبة.

\* @returns {object} - كائن يحتوي على ملخص البيانات للوحة التحكم.

\*/

function generateDashboardSummary() {

GA\_Helpers.logOperation('GAssistant.Analytics', 'generateDashboardSummary:start', {});

try {

// جمع إحصاءات المحاسبة

const today = Utilities.formatDate(new Date(), Session.getScriptTimeZone(), "yyyy-MM-dd");

const accountingSummary = GA\_Tools\_Accounting.calculateGrossProfit(today, today); // مثال: جلب ربح اليوم

const summaryData = [

{ Metric: "إجمالي الإيرادات (اليوم)", Value: accountingSummary.totalRevenue ? `${accountingSummary.totalRevenue.toFixed(2)} ${accountingSummary.currency}` : "N/A" },

{ Metric: "إجمالي المصروفات (اليوم)", Value: accountingSummary.totalExpenses ? `${accountingSummary.totalExpenses.toFixed(2)} ${accountingSummary.currency}` : "N/A" },

{ Metric: "صافي الربح (اليوم)", Value: accountingSummary.grossProfit ? `${accountingSummary.grossProfit.toFixed(2)} ${accountingSummary.currency}` : "N/A" },

// يمكن إضافة المزيد من المقاييس هنا

{ Metric: "عدد العمليات المسجلة (اليوم)", Value: GA\_Helpers.getOperationLogCountByDate(today) || 0 } // افتراض وجود دالة في Helpers

];

GA\_Helpers.logOperation('GAssistant.Analytics', 'generateDashboardSummary:success', { data: summaryData });

return GA\_AI\_Dialogue.showSummaryTable(summaryData, "📊 ملخص لوحة التحكم التحليلية");

} catch (e) {

GA\_Helpers.logOperation('GAssistant.Analytics', 'generateDashboardSummary:error', { error: e.message });

return GA\_AI\_Dialogue.showError(`❌ فشل في توليد ملخص لوحة التحكم: ${e.message}`);

}

}

/\*\*

\* @function showAnalyticsDashboardSidebar

\* @description

\* تعرض الشريط الجانبي للوحة التحكم التحليلية.

\*/

function showAnalyticsDashboardSidebar() {

GA\_Helpers.logOperation('GAssistant.Analytics', 'showAnalyticsDashboardSidebar:start', {});

try {

// هنا يمكن بناء واجهة HTML بسيطة لعرض الملخص

const summary = generateDashboardSummary(); // جلب البيانات

let htmlContent = `<div style="font-family: 'Inter', sans-serif; padding: 15px; background-color: #f0f2f5; color: #333; border-radius: 8px;">`;

htmlContent += `<h2 style="text-align: center; color: #004085; margin-bottom: 20px;">📊 لوحة التحكم التحليلية</h2>`;

if (summary.type === 'summary\_table' && summary.data) {

htmlContent += `<table style="width: 100%; border-collapse: collapse; margin-bottom: 20px;">`;

htmlContent += `<tr style="background-color: #e0e7ff;"><th style="padding: 8px; border: 1px solid #ddd; text-align: right;">المقياس</th><th style="padding: 8px; border: 1px solid #ddd; text-align: right;">القيمة</th></tr>`;

summary.data.forEach(item => {

htmlContent += `<tr><td style="padding: 8px; border: 1px solid #ddd; text-align: right;">${item.Metric}</td><td style="padding: 8px; border: 1px solid #ddd; text-align: right;">${item.Value}</td></tr>`;

});

htmlContent += `</table>`;

} else {

htmlContent += `<p style="color:red;">${summary.text || "فشل في جلب بيانات لوحة التحكم."}</p>`;

}

htmlContent += `<button onclick="google.script.run.withSuccessHandler(closeSidebar).showAnalyticsDashboardSidebar()" style="background-color: #007bff; color: white; padding: 10px 15px; border: none; border-radius: 5px; cursor: pointer; display: block; margin: 10px auto;">تحديث</button>`;

htmlContent += `</div>`;

const html = HtmlService.createHtmlOutput(htmlContent)

.setTitle('📊 لوحة التحكم')

.setWidth(400);

SpreadsheetApp.getUi().showSidebar(html);

GA\_Helpers.logOperation('GAssistant.Analytics', 'showAnalyticsDashboardSidebar:end', {});

} catch (e) {

GA\_Helpers.logOperation('GAssistant.Analytics', 'showAnalyticsDashboardSidebar:error', { error: e.message });

GA\_AI\_Dialogue.showError(`❌ فشل في فتح لوحة التحكم التحليلية: ${e.message}`);

}

}

// Helper to close sidebar, used in HTML button

function closeSidebar() {

SpreadsheetApp.getUi().closeSidebar();

}

// --- تصدير الواجهة العامة للوحدة ---

return {

generateDashboardSummary,

showAnalyticsDashboardSidebar

};

})(

GAssistant.Helpers,

GAssistant.AI.Dialogue,

GAssistant.Tools.Accounting // افتراض وجود هذه الوحدة

);

// ---------------------------------------------------------------------------

// الدوال العامة (Global Functions)

// ---------------------------------------------------------------------------

// هذه الدالة ضرورية لتمكين google.script.run من استدعاء الدوال من HTML.

/\*\*

\* @function showAnalyticsDashboardSidebar

\* @description

\* دالة عامة لفتح الشريط الجانبي للوحة التحكم التحليلية.

\*/

function showAnalyticsDashboardSidebar() {

GAssistant.Analytics.showAnalyticsDashboardSidebar();

}