function extractLatestEmailsAndStoreInSheet24() {

var inputSheetUrl = "https://docs.google.com/spreadsheets/d/10SdEY28gEgJNSnVnGXV3bxe5\_Z2cV3ARGSNXjs2Ie4Y/edit#gid=0"; // URL for the sheet with domains to read

var outputSheetUrl = "https://docs.google.com/spreadsheets/d/1ISjybHAt\_pQfHP2AostOI60APgCnqeY6isThualCRMA/edit#gid=0"; // URL for the sheet to append the output

var inputSpreadsheet = SpreadsheetApp.openByUrl(inputSheetUrl);

var outputSpreadsheet = SpreadsheetApp.openByUrl(outputSheetUrl);

var inputSheet = inputSpreadsheet.getSheetByName("Replying\_Latest\_Email\_Based\_On\_Domain");

var outputSheet = outputSpreadsheet.getSheetByName("Replying\_And\_Extracting");

outputSheet.clear();

outputSheet.appendRow([

"Timestamp", "From name", "From email", "To name", "To email", "From domain", "To domain",

"Subject", "Body", "CC", "Labels", "internalDomainDate", "externalDomainDate",

"internalDomainMessageCount", "externalDomainMessageCount"

]);

var domainData = inputSheet.getRange("A2:A" + inputSheet.getLastRow()).getValues();

var bodyData = inputSheet.getRange("B2:B" + inputSheet.getLastRow()).getValues();

domainData.forEach(function (row, index) {

var domain = row[0];

var body = bodyData[index][0];

var threads = GmailApp.search("from:@" + domain);

threads.forEach(function (thread) {

var threadMessages = thread.getMessages();

var latestEmail = threadMessages[threadMessages.length - 1];

var internalDomain = "morulaa.com";

var internalDomainLatestMessage = null;

var externalDomainLatestMessage = null;

var internalDomainMessageCount = 0;

var externalDomainMessageCount = 0;

var internalDomainDate = "";

var externalDomainDate = "";

if (latestEmail.getDate() >= new Date(new Date() - 1 \* 24 \* 60 \* 60 \* 1000)) {

var subject = latestEmail.getSubject();

var timestamp = latestEmail.getDate();

var fromEmail = latestEmail.getFrom();

var toemail = latestEmail.getTo();

var fromName = extractNameFromEmail(fromEmail);

var emailId = extractEmailIdFromEmail(fromEmail);

var ToName = extractNameFromEmail(toemail);

var tomailid = extractEmailIdFromEmail(toemail);

var fromdomain = extractDomainFromEmail(fromEmail);

var todomain = extractDomainFromEmail2(tomailid);

var cc = extractCCFromEmail(latestEmail);

var tbody = body; // Using the email body from the Google Sheets

var threadLabels = thread.getLabels();

var labelNames = [];

if (threadLabels.length > 0) {

labelNames = threadLabels.map(function (label) {

return label.getName();

});

}

var llabels = labelNames.join(", ");

for (var i = 0; i < threadMessages.length; i++) {

var fromEmailloop = threadMessages[i].getFrom();

var tm = threadMessages[i];

var fromName1 = extractDomainFromEmail(fromEmailloop);

if (internalDomain === fromName1) {

internalDomainMessageCount++;

internalDomainLatestMessage = latestEmail;

internalDomainDate = tm.getDate();

} else {

externalDomainMessageCount++;

externalDomainLatestMessage = latestEmail;

externalDomainDate = tm.getDate();

}

}

var internalDomainSubject = internalDomainLatestMessage ? internalDomainLatestMessage.getSubject() : "";

var internalDomainBody = internalDomainLatestMessage ? internalDomainLatestMessage.getPlainBody() : "";

var externalDomainSubject = externalDomainLatestMessage ? externalDomainLatestMessage.getSubject() : "";

var externalDomainBody = externalDomainLatestMessage ? externalDomainLatestMessage.getPlainBody() : "";

// Reply to the latest email and include the body from the Google Sheets

var replyBody = "Your custom reply body here for domain: " + domain;

latestEmail.reply(subject, replyBody);

outputSheet.appendRow([

timestamp, fromName, emailId, ToName, tomailid, fromdomain, todomain, subject, tbody,

cc, llabels, internalDomainDate, externalDomainDate,

internalDomainMessageCount, externalDomainMessageCount

]);

}

});

});

}

// Helper functions (unchanged)

function extractNameFromEmail(email) {

var name = email.match(/^(.\*)</);

return name ? name[1].trim() : "";

}

function extractEmailIdFromEmail(email) {

var emailId = email.match(/<([^>]+)>/);

return emailId ? emailId[1] : email;

}

function extractDomainFromEmail(email) {

var domain = email.match(/@(.+?)(?:>|$)/);

return domain && domain.length >= 2 ? domain[1] : "Invalid domain";

}

function extractDomainFromEmail2(email) {

var atIndex = email.indexOf("@");

return atIndex !== -1 ? email.substring(atIndex + 1) : "Invalid email format";

}

function extractCCFromEmail(email) {

var ccAddresses = email.getCc();

return ccAddresses && Array.isArray(ccAddresses) ? ccAddresses.join(", ") : (ccAddresses || "");

}