function myFunction(){

var ss = SpreadsheetApp.getActiveSpreadsheet();

var result = [['Channel URL', 'Name','Comment','Time','Likes','Reply Count','Reply Author','Reply','Published','Updated']];

var vid = ss.getSheetByName('Input').getRange(1,2).getValue();

var maxComments = 50; // Limit main comments to 50

var maxReplies = 10; // Limit replies per comment to 10

var commentCount = 0;

var nextPageToken = undefined;

// Loop through comment pages until we reach the limit or run out of comments

while(commentCount < maxComments){

var commentThreads = YouTube.CommentThreads.list('snippet', {videoId: vid, maxResults: Math.min(100, maxComments - commentCount), pageToken: nextPageToken});

nextPageToken = commentThreads.nextPageToken;

// Process each comment

for (var row = 0; row < commentThreads.items.length && commentCount < maxComments; row++) {

commentCount++;

result.push([

commentThreads.items[row].snippet.topLevelComment.snippet.authorChannelUrl,

commentThreads.items[row].snippet.topLevelComment.snippet.authorDisplayName,

commentThreads.items[row].snippet.topLevelComment.snippet.textDisplay,

commentThreads.items[row].snippet.topLevelComment.snippet.publishedAt,

commentThreads.items[row].snippet.topLevelComment.snippet.likeCount,

commentThreads.items[row].snippet.totalReplyCount,'','','','']);

// If comment has replies and we haven't exceeded the limit

if(commentThreads.items[row].snippet.totalReplyCount > 0){

var parent = commentThreads.items[row].snippet.topLevelComment.id;

var replyCount = 0;

var nextPageTokenRep = undefined;

// Loop through reply pages until we reach the limit or run out of replies

while(replyCount < maxReplies){

var commentThreads2 = YouTube.Comments.list('snippet', {

videoId: vid,

maxResults: Math.min(100, maxReplies - replyCount),

pageToken: nextPageTokenRep,

parentId: parent

});

// Process each reply

for (var i = Math.min(commentThreads2.items.length, maxReplies - replyCount) - 1; i >= 0; i--){

replyCount++;

result.push(['', '','','','','',

commentThreads2.items[i].snippet.authorDisplayName,

commentThreads2.items[i].snippet.textDisplay,

commentThreads2.items[i].snippet.publishedAt,

commentThreads2.items[i].snippet.updatedAt]);

}

nextPageTokenRep = commentThreads2.nextPageToken;

if(nextPageTokenRep == "" || typeof nextPageTokenRep === "undefined" || replyCount >= maxReplies){

break;

}

}

}

}

if(nextPageToken == "" || typeof nextPageToken === "undefined" || commentCount >= maxComments){

break;

}

}

// Create new sheet and write results

var newSheet = ss.insertSheet(ss.getNumSheets());

newSheet.getRange(1, 1, result.length, 10).setValues(result);

// Optional: Add info about limits

newSheet.getRange(result.length + 2, 1).setValue("Note: Results limited to " + maxComments + " comments and " + maxReplies + " replies per comment.");

}