Adding Conversions to The KW Performance Report - Script Tutorial

Follow these instructions to get going with AdWords scripts in under a minute.

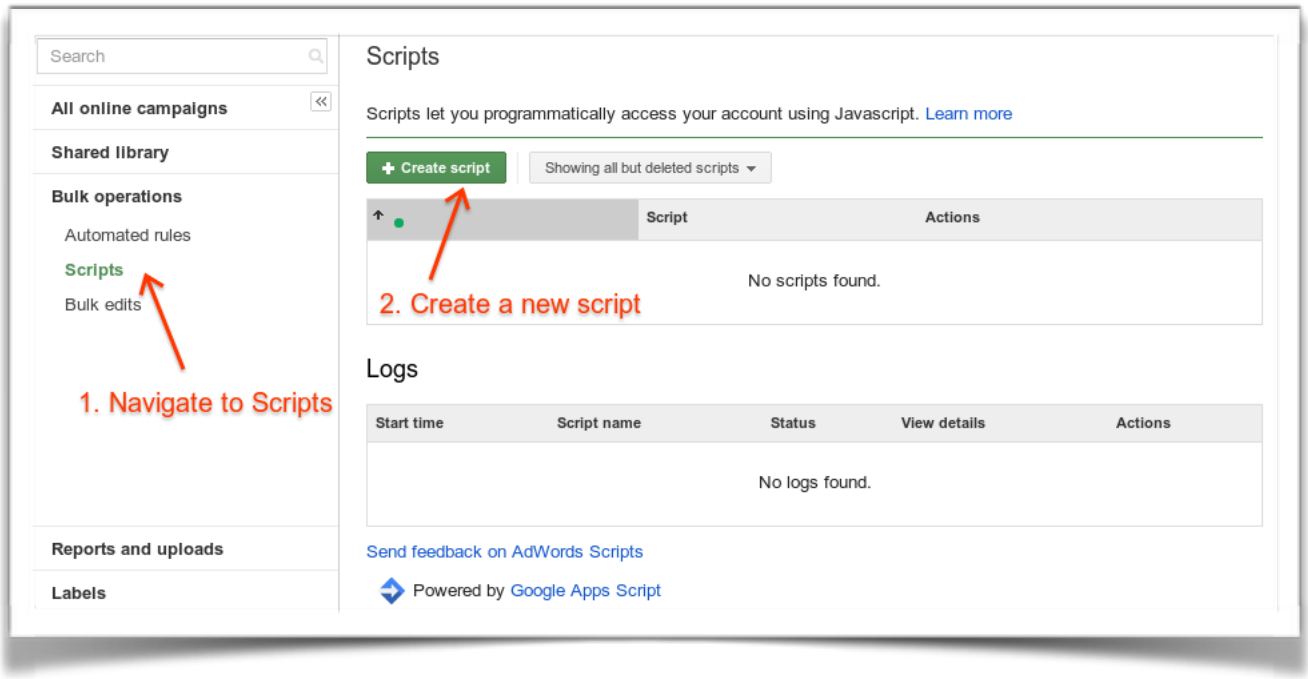
Note: These scripts are proprietary and will not work without an active free trial or Hero Pro plan.

What it Does: This script breaks down all keywords in an account by average position and quality score. The metrics are summed across these segments, letting you see how your performance varies across position and quality score.

What Changed: This script takes conversions and conversion rate and adds them to the report. This allows you to see how your conversions and conversion rate vary by position and quality score. By optimizing for quality score, are you getting a worthwhile return? Similarly, the higher positions get many more clicks, are these clicks actually converting at a reasonable rate?

1. **Navigate to "Bulk operations" → "Scripts" in the AdWords web application.**

2. **Click "Create script.**



3. **Copy and paste the following code into the AdWords Script editor area:**

```
// SpreadSheet Url. Starts with https://docs.google.com...
var SPREADSHEET_URL = 'spreadsheet URL';

/**
* This script computes a keyword performance report
* and outputs it to a Google spreadsheet
*/
function main() {
 var spreadsheet = getSpreadsheet(SPREADSHEET_URL);
 var sheet = spreadsheet.getSheetByName('Sheet1');
 outputQualityScoreData(sheet);
```

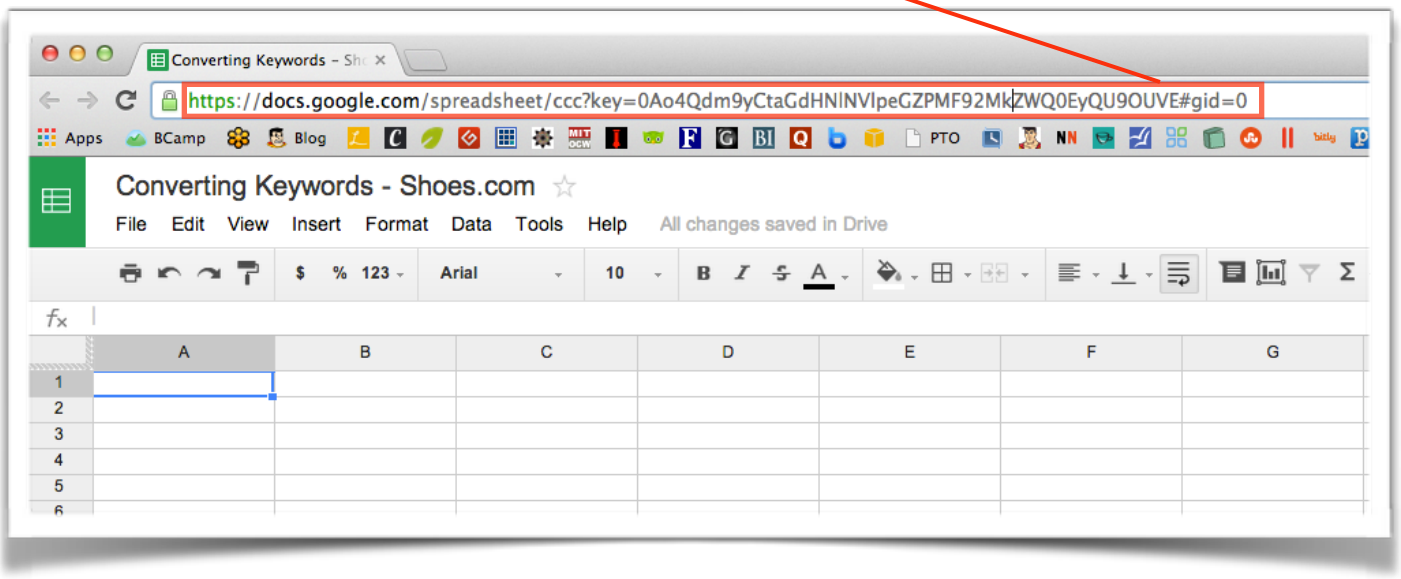
```
 outputPositionData(sheet);
 Logger.log('Keyword performance report - ' + SPREADSHEET_URL);
}

/**
* Retrieves the spreadsheet identified by the URL.
* @param {string} spreadsheetUrl The URL of the spreadsheet.
* @return {SpreadSheet} The spreadsheet.
*/
function getSpreadsheet(spreadsheetUrl) {
 var matches = new RegExp('key=([^&#]*)').exec(spreadsheetUrl);
 if (!matches || !matches[1]) {
   throw 'Invalid spreadsheet URL: ' + spreadsheetUrl;
 }
 var spreadsheetId = matches[1];
 return SpreadsheetApp.openById(spreadsheetId);
}

/**
* Outputs Quality score related data to the spreadsheet
* @param {Sheet} sheet The sheet to output to.
*/
function outputQualityScoreData(sheet) {
 // Output header row
 var header = [
   'Quality Score',
   'Num Keywords',
   'Impressions',
   'Clicks',
   'CTR (%)',
   'Cost',
   'Conversions',
   'Conversion Rate'
 ];
 sheet.getRange(1, 1, 1, 8).setValues([header]);

 // Initialize
 var qualityScoreMap = [];
 for (i = 1; i <= 10; i++) {
   qualityScoreMap[i] = {
     numKeywords: 0,
     totalImpressions: 0,
     totalClicks: 0,
     totalCost: 0.0,
     totalConversions: 0,
     conversionRate: 0,
   };
 }

 // Compute data
 var keywordIterator = AdWordsApp.keywords()
     .forDateRange('LAST_WEEK')
     .withCondition('Impressions > 0')
     .get();
 while (keywordIterator.hasNext()) {
   var keyword = keywordIterator.next();
   var stats = keyword.getStatsFor('LAST_WEEK');
   var data = qualityScoreMap[keyword.getQualityScore()];
   if (data) {
     data.numKeywords++;
     data.totalImpressions += stats.getImpressions();
     data.totalClicks += stats.getClicks();
     data.totalCost += stats.getCost();
     data.totalConversions += stats.getConversions();
     data.conversionRate += stats.getConversionRate();
```

```
  }
 }

 // Output data to spreadsheet
 var rows = [];
 for (var key in qualityScoreMap) {
  var ctr = 0;
  var cost = 0.0;
  var conversionRate= 0;
  if (qualityScoreMap[key].numKeywords > 0) {
   ctr = (qualityScoreMap[key].totalClicks /
     qualityScoreMap[key].totalImpressions) * 100;
   conversionRate= (qualityScoreMap[key].totalConversions / qualityScoreMap[key].totalClicks) * 100;
  }
  var row = [
   key,
   qualityScoreMap[key].numKeywords,
   qualityScoreMap[key].totalImpressions,
   qualityScoreMap[key].totalClicks,
   ctr.toFixed(2),
   qualityScoreMap[key].totalCost,
   qualityScoreMap[key].totalConversions,
   conversionRate.toFixed(2)

  ];
  rows.push(row);
 }
 sheet.getRange(2, 1, rows.length, 8).setValues(rows);
}

/**
* Outputs average position related data to the spreadsheet.
* @param {Sheet} sheet The sheet to output to.
*/
function outputPositionData(sheet) {
 // Output header row
 headerRow = [];
 var header = [
  'Avg Position',
  'Num Keywords',
  'Impressions',
  'Clicks',
  'CTR (%)',
  'Cost',
  'Conversions',
  'Conversion Rate'
 ];
 headerRow.push(header);
 sheet.getRange(14, 1, 1, 8).setValues(headerRow);

 // Initialize
 var positionMap = [];
 for (i = 1; i <= 12; i++) {
  positionMap[i] = {
   numKeywords: 0,
   totalImpressions: 0,
   totalClicks: 0,
   totalCost: 0.0,
   totalConversions: 0,
   conversionRate: 0

  };
 }

 // Compute data
```

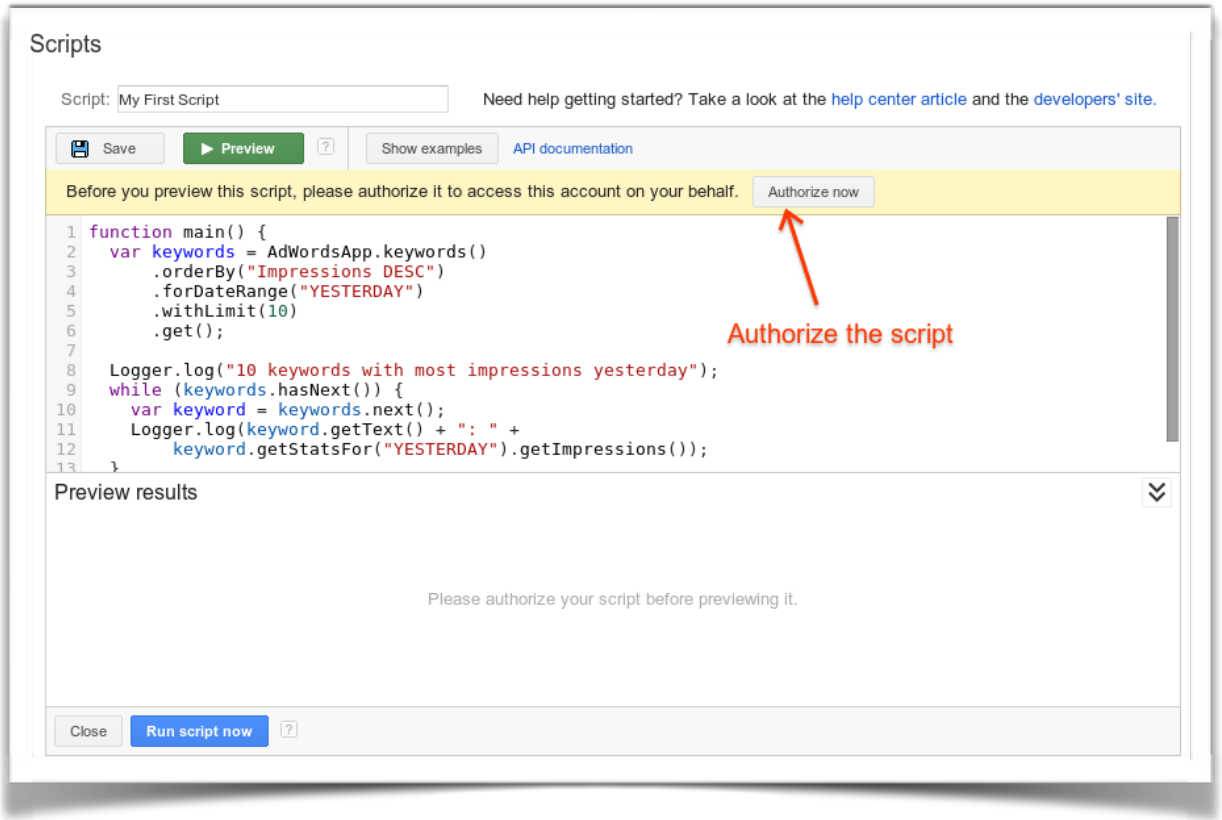
```
 var keywordIterator = AdWordsApp.keywords()
     .forDateRange('LAST_WEEK')
     .withCondition('Impressions > 0')
     .get();
 while (keywordIterator.hasNext()) {
   var keyword = keywordIterator.next();
   var stats = keyword.getStatsFor('LAST_WEEK');

   if (stats.getAveragePosition() <= 11) {
     var data = positionMap[Math.ceil(stats.getAveragePosition())];
   } else {
     // All positions greater than 11
     var data = positionMap[12];
   }
   data.numKeywords++;
   data.totalImpressions += stats.getImpressions();
   data.totalClicks += stats.getClicks();
   data.totalCost += stats.getCost();
   data.totalConversions += stats.getConversions();
   data.conversionRate += stats.getConversionRate();

 }

// Output data to spreadsheet
var rows = [];
for (var key in positionMap) {
  var ctr = 0;
  var cost = 0.0;
  var conversionRate = 0;
  if (positionMap[key].numKeywords > 0) {
    ctr = (positionMap[key].totalClicks /
      positionMap[key].totalImpressions) * 100;
    conversionRate = (positionMap[key].totalConversions / positionMap[key].totalClicks) * 100;

  }
  var row = [
    key <= 11 ? key - 1 + ' to ' + key : '>11',
    positionMap[key].numKeywords,
    positionMap[key].totalImpressions,
    positionMap[key].totalClicks,
    ctr.toFixed(2),
    positionMap[key].totalCost,
    positionMap[key].totalConversions,
    conversionRate.toFixed(2)
  ];
  rows.push(row);
}
sheet.getRange(15, 1, rows.length, 8).setValues(rows);
}
```

4. **Now copy and paste your newly created template URL into the script between the apostrophe.**

```
// SpreadSheet Url. Starts with https://docs.google.com...
  var SPREADSHEET_URL = 'spreadsheet URL';
```



5. **Authorize the script. Don't worry, this only has to be done once per script.**



6. **Click “Run script now” and select “Run without previewing.”**

    - Since the reports live in your Google Drive spreadsheet previewing does nothing.

7. **After the script status says “Completed” you can view results on your Google Drive spreadsheet!**