

# How come my Google Sheets that export to PDF script is no longer working?

First identify if you are using this approach to print to PDF.

Using manual URL manipulation to generate the export link for the sheet.

```
function PrintToPdf(sheetNumber, pdfName) {  
  var spreadsheet = SpreadsheetApp.getActiveSpreadsheet();  
  var spreadsheetId = spreadsheet.getId();  
  var sheetId = sheetNumber ? spreadsheet.getSheets()[sheetNumber].getSheetId() : null;  
  //var url_base = spreadsheet.getUrl().replace(/edit$/, '');  
  var url_base = "https://docs.google.com/spreadsheets/d/[REDACTED]";  
  
  var url_ext = '/export?exportFormat=pdf&format=pdf' //exporter en pdf  
  
  + (sheetId ? ('&gid=' + sheetId) : ('&id=' + spreadsheetId))  
  // les paramètres suivants sont optionnels ...  
  + '&size=A4' // taille de papier  
  + '&portrait=true' // orientation, false pour le paysage  
  + '&fitw=true' // ajustement à la largeur, faux pour la taille réelle  
  + '&sheetnames=true&printtitle=false&pagenumbers=true' //masquer les en-têtes et les pieds de page facultatifs  
  + '&gridlines=true' // masquer le quadrillage  
  + '&fzr=false'; // ne répétez pas les en-têtes de lignes (lignes figées) sur chaque page  
  
  var options = {  
    headers: {  
      'Authorization': 'Bearer ' + ScriptApp.getOAuthToken(),  
    }  
  }  
}
```

This approach relies on Google to maintain the URL to be the same. However Google will always changes and make the product better.

Therefore they will change the URL very often which is not a feasible way to implement the export to PDF.

The more sustainable approach is to use Google's library. For the example below, it uses SpreadsheetApp and DriveApp library provided default within the Google App Script.

When Google implement changes, these libraries are less likely to be changed that often. They only change it only when they really have to.

Here is an example code to download the download the sheet as PDF.

```
function downloadAsPDF( folderId ) {  
  var pdfFile = SpreadsheetApp.getActiveSpreadsheet().getAs("application/pdf");  
  var newfile = DriveApp.createFile(pdfFile);  
  
  var destinationFolder = DriveApp.getFolderById(folderId);  
  destinationFolder.addFile(newfile);  
  DriveApp.getRootFolder().removeFile(newfile);  
}
```