

100 Useful Apps Script Code Snippets

Google Apps Script is a powerful way to automate, customize, and supercharge your workflow in Google Docs. Whether you want to clean up references, format your document, or do advanced text processing, a simple script can save you hours. Here are 10 essential Google Apps Script snippets every Google Docs power user should know!

1. Remove All [cite: 1234] References

Description:

This script removes every instance of [cite: 1234] (where the number can be anything) from your document.

```
function removeCiteReferences() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var pattern = /\[cite:\s*\d+\]/g;  
  var text = body.getText();  
  var cleanedText = text.replace(pattern, '');  
  body.setText(cleanedText);  
}
```

2. Convert All “Exercise” Sections to Heading 2

Description:

Automatically converts every paragraph starting with "Exercise <number>:" to the Heading 2 style.

```
function convertExercisesToHeading2() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var totalParas = body.getNumChildren();  
  
  for (var i = 0; i < totalParas; i++) {  
    var element = body.getChild(i);  
    if (element.getType() == DocumentApp.ElementType.PARAGRAPH) {  
      var para = element.asParagraph();  
    }  
  }  
}
```

```
    var text = para.getText();
    if (/^Exercise\s+\d+:/i.test(text)) {
        para.setHeading(DocumentApp.ParagraphHeading.HEADING2);
    }
}
}
}
```

3. Find and Highlight All TODOs

Description:

Highlights every “TODO” in yellow so you can spot unfinished work at a glance.

```
function highlightTodos() {
    var body = DocumentApp.getActiveDocument().getBody();
    var found = body.findText('TODO');
    while (found) {
        found.getElement().setBackgroundColor('yellow');
        found = body.findText('TODO', found);
    }
}
```

4. Replace All Instances of a Word

Description:

Find and replace every instance of a specific word or phrase.

```
function replaceWord(oldWord, newWord) {
    var body = DocumentApp.getActiveDocument().getBody();
    body.replaceText(oldWord, newWord);
}
// Example usage: replaceWord('foo', 'bar');
```

5. Add a Table of Contents at the Top

Description:

Inserts an up-to-date Table of Contents at the beginning of your document.

```
function insertTableOfContents() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  body.insertParagraph(0, "Table of  
Contents").setHeading(DocumentApp.ParagraphHeading.HEADING1);  
  body.insertTableOfContents(1, DocumentApp.TableOfContentsType.FLAT);  
}
```

6. Remove Extra Blank Lines

Description:

Deletes consecutive blank paragraphs for a cleaner document.

```
function removeExtraBlankLines() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var numChildren = body.getNumChildren();  
  for (var i = numChildren - 1; i > 0; i--) {  
    var curr = body.getChild(i);  
    var prev = body.getChild(i - 1);  
    if (curr.getType() == DocumentApp.ElementType.PARAGRAPH &&  
        prev.getType() == DocumentApp.ElementType.PARAGRAPH &&  
        !curr.asParagraph().getText() &&  
        !prev.asParagraph().getText()) {  
      body.removeChild(curr);  
    }  
  }  
}
```

7. List All Images with Their Index

Description:

Prints a log of all images in your document with their position number.

```
function listAllImages() {
  var body = DocumentApp.getActiveDocument().getBody();
  var imgs = body.getImages();
  for (var i = 0; i < imgs.length; i++) {
    Logger.log("Image " + (i + 1) + " found.");
  }
}
```

8. Count Words in Your Document

Description:

Counts and logs the number of words in your document.

```
function countWords() {
  var text = DocumentApp.getActiveDocument().getBody().getText();
  var wordCount = text.match(/\b\S+\b/g)?.length || 0;
  Logger.log('Total words: ' + wordCount);
}
```

9. Change All Headings to Title Case

Description:

Automatically updates all headings (H1, H2, H3, etc.) to Title Case.

```
function headingsToTitleCase() {
  var body = DocumentApp.getActiveDocument().getBody();
  var total = body.getNumChildren();
  for (var i = 0; i < total; i++) {
    var elem = body.getChild(i);
    if (elem.getType() == DocumentApp.ElementType.PARAGRAPH) {
      var para = elem.asParagraph();
      var heading = para.getHeading();
      if (heading != DocumentApp.ParagraphHeading.NORMAL) {
        var txt = para.getText();
```

```
        var newTxt = txt.replace(/\w\S*/g, w =>
w.charAt(0).toUpperCase() + w.slice(1).toLowerCase());
        para.setText(newTxt);
    }
}
}
}
```

10. Insert the Current Date Anywhere

Description:

Quickly insert today's date wherever your cursor is in the doc.

```
function insertCurrentDate() {
    var cursor = DocumentApp.getActiveDocument().getCursor();
    if (cursor) {
        var dateStr = Utilities.formatDate(new Date(),
Session.getScriptTimeZone(), "yyyy-MM-dd");
        cursor.insertText(dateStr);
    } else {
        DocumentApp.getUi().alert('Place your cursor in the document
first.');
    }
}
```

How to Use These Snippets

1. Open your Google Doc.
2. Go to **Extensions → Apps Script**.
3. Paste your chosen script(s) into the code editor.
4. Save and run the function you want (authorize if prompted).

11. Duplicate the Active Document

```
function duplicateActiveDoc() {  
  var doc = DocumentApp.getActiveDocument();  
  DriveApp.getFileById(doc.getId()).makeCopy('Copy of ' +  
doc.getName());  
}
```

12. Get a List of All Headings

```
function listAllHeadings() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var numChildren = body.getNumChildren();  
  for (var i = 0; i < numChildren; i++) {  
    var elem = body.getChild(i);  
    if (elem.getType() == DocumentApp.ElementType.PARAGRAPH) {  
      var para = elem.asParagraph();  
      if (para.getHeading() != DocumentApp.ParagraphHeading.NORMAL) {  
        Logger.log(para.getText());  
      }  
    }  
  }  
}
```

13. Export Document as PDF and Email

```
function emailAsPDF() {  
  var doc = DocumentApp.getActiveDocument();  
  var file = DriveApp.getFileById(doc.getId());  
  var pdf = file.getAs('application/pdf');  
  MailApp.sendEmail('your@email.com', 'Document PDF', 'See attached.',  
{attachments:[pdf]});  
}
```

14. Remove All Hyperlinks

```
function removeAllHyperlinks() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var search = body.findText('https?://[^\\s]+', null, {matchCase:  
false, wholeWord: false, useRegularExpression: true});  
  while (search) {  
    search.getElement().setLinkUrl(null);  
    search = body.findText('https?://[^\\s]+', search,  
{useRegularExpression: true});  
  }  
}
```

15. Capitalize All Paragraphs

```
function capitalizeAllParagraphs() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var n = body.getNumChildren();  
  for (var i = 0; i < n; i++) {  
    var el = body.getChild(i);  
    if (el.getType() == DocumentApp.ElementType.PARAGRAPH) {  
      var txt = el.asParagraph().getText();  
      el.asParagraph().setText(txt.toUpperCase());  
    }  
  }  
}
```

16. Count All Images

```
function countImages() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  Logger.log('Image count: ' + body.getImages().length);  
}
```

17. Convert All Text to Arial

```
function convertAllTextToArial() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  body.editAsText().setFontFamily('Arial');  
}
```

18. Remove All Comments

```
function removeAllComments() {  
  var doc = DocumentApp.getActiveDocument();  
  var comments = Drive.Comments.list(doc.getId()).items;  
  comments.forEach(function(comment) {  
    Drive.Comments.remove(doc.getId(), comment.commentId);  
  });  
}
```

19. Insert a Footer

```
function insertFooter() {  
  var doc = DocumentApp.getActiveDocument();  
  doc.addFooter().appendParagraph("Confidential");  
}
```

20. Insert a Header

```
function insertHeader() {  
  var doc = DocumentApp.getActiveDocument();  
  doc.addHeader().appendParagraph("Draft Version");  
}
```

21. Highlight All Email Addresses

```
function highlightEmails() {
  var body = DocumentApp.getActiveDocument().getBody();
  var found =
body.findText('[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.\w{2,}', null,
{useRegularExpression:true});
  while (found) {
    found.getElement().setBackgroundColor('lightblue');
    found =
body.findText('[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.\w{2,}', found,
{useRegularExpression:true});
  }
}
```

22. Make All Paragraphs Double-Spaced

```
function doubleSpaceParagraphs() {
  var body = DocumentApp.getActiveDocument().getBody();
  var n = body.getNumChildren();
  for (var i = 0; i < n; i++) {
    var p = body.getChild(i);
    if (p.getType() == DocumentApp.ElementType.PARAGRAPH) {
      p.asParagraph().setLineSpacing(2);
    }
  }
}
```

23. Add Page Numbers to Footer

```
function addPageNumbersToFooter() {
  var doc = DocumentApp.getActiveDocument();
  var footer = doc.addFooter();
  footer.appendPageNumber();
}
```

24. Insert Image by URL

```
function insertImageByUrl() {  
  var url =  
    'https://www.google.com/images/branding/googlelogo/1x/googlelogo_color  
_272x92dp.png';  
  var imgBlob = UrlFetchApp.fetch(url).getBlob();  
  DocumentApp.getActiveDocument().getBody().appendImage(imgBlob);  
}
```

25. Find and Bold All Numbers

```
function boldAllNumbers() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var found = body.findText('\d+', null, {useRegularExpression:  
true});  
  while (found) {  
    found.getElement().setBold(true);  
    found = body.findText('\d+', found, {useRegularExpression:  
true});  
  }  
}
```

26. List All Bookmarks

```
function listBookmarks() {  
  var doc = DocumentApp.getActiveDocument();  
  var bookmarks = doc.getBookmarks();  
  bookmarks.forEach(function(bm, i) {  
    Logger.log('Bookmark ' + (i+1) + ': ' + bm.getId());  
  });  
}
```

27. Remove All Bookmarks

```
function removeBookmarks() {  
  var doc = DocumentApp.getActiveDocument();  
  doc.getBookmarks().forEach(function(bm) { bm.remove(); });  
}
```

28. Merge All Tables into One

```
function mergeAllTables() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var tables = [];  
  for (var i = 0; i < body.getNumChildren(); i++) {  
    if (body.getChild(i).getType() == DocumentApp.ElementType.TABLE) {  
      tables.push(body.getChild(i).asTable());  
    }  
  }  
  if (tables.length < 2) return;  
  var mainTable = tables[0];  
  for (var i = 1; i < tables.length; i++) {  
    var table = tables[i];  
    for (var r = 0; r < table.getNumRows(); r++) {  
      mainTable.appendTableRow(table.getRow(r).copy());  
    }  
    body.removeChild(table);  
  }  
}
```

29. Count All Tables

```
function countTables() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var count = 0;
```

```
for (var i = 0; i < body.getNumChildren(); i++) {
    if (body.getChild(i).getType() == DocumentApp.ElementType.TABLE)
        count++;
}
Logger.log("Table count: " + count);
}
```

30. Add “Draft” Watermark to Each Page (Header)

```
function draftHeader() {
    var doc = DocumentApp.getActiveDocument();
    doc.addHeader().appendParagraph("DRAFT");
}
```

31. Center Align All Headings

```
function centerAlignHeadings() {
    var body = DocumentApp.getActiveDocument().getBody();
    var n = body.getNumChildren();
    for (var i = 0; i < n; i++) {
        var elem = body.getChild(i);
        if (elem.getType() == DocumentApp.ElementType.PARAGRAPH) {
            var para = elem.asParagraph();
            if (para.getHeading() != DocumentApp.ParagraphHeading.NORMAL) {
                para.setAlignment(DocumentApp.HorizontalAlignment.CENTER);
            }
        }
    }
}
```

32. Remove All Bold Formatting

```
function removeAllBold() {
```

```
var text = DocumentApp.getActiveDocument().getBody().editAsText();
text.setBold(false);
}
```

33. List All Links

```
function listAllLinks() {
  var body = DocumentApp.getActiveDocument().getBody();
  var text = body.getText();
  var matches = text.match(/https?:\/\/[^\s]+/g);
  if (matches) Logger.log(matches.join('\n'));
}
```

34. Insert a Horizontal Line

```
function insertHorizontalLine() {
  DocumentApp.getActiveDocument().getBody().appendHorizontalRule();
}
```

35. Add “Reviewed” to Document Title

```
function markReviewedInTitle() {
  var doc = DocumentApp.getActiveDocument();
  doc.setName(doc.getName() + ' (Reviewed)');
}
```

36. Replace All Numbers with “#”

```
function replaceAllNumbers() {
  var body = DocumentApp.getActiveDocument().getBody();
  body.setText(body.getText().replace(/\d+/g, '#'));
}
```

37. Add Indentation to All Paragraphs

```
function indentAllParagraphs() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var n = body.getNumChildren();  
  for (var i = 0; i < n; i++) {  
    var para = body.getChild(i);  
    if (para.getType() == DocumentApp.ElementType.PARAGRAPH) {  
      para.asParagraph().setIndentStart(36); // Half-inch  
    }  
  }  
}
```

38. Remove All Indentation

```
function removeAllIndentation() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var n = body.getNumChildren();  
  for (var i = 0; i < n; i++) {  
    var para = body.getChild(i);  
    if (para.getType() == DocumentApp.ElementType.PARAGRAPH) {  
      para.asParagraph().setIndentStart(0);  
    }  
  }  
}
```

39. Append Text at the End

```
function appendTextEnd() {  
  DocumentApp.getActiveDocument().getBody().appendParagraph('--- End  
of Document ---');  
}
```

40. Insert Current User's Email

```
function insertCurrentUserEmail() {  
  var email = Session.getActiveUser().getEmail();  
  DocumentApp.getActiveDocument().getBody().appendParagraph(email);  
}
```

41. List All Unique Words

```
function listUniqueWords() {  
  var text = DocumentApp.getActiveDocument().getBody().getText();  
  var words = text.match(/\b\w+\b/g);  
  var unique = Array.from(new Set(words));  
  Logger.log(unique.join(', '));  
}
```

42. Delete All Tables

```
function deleteAllTables() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  for (var i = body.getNumChildren() - 1; i >= 0; i--) {  
    if (body.getChild(i).getType() == DocumentApp.ElementType.TABLE) {  
      body.removeChild(body.getChild(i));  
    }  
  }  
}
```

43. Add Highlight to All Headings

```
function highlightAllHeadings() {  
  var body = DocumentApp.getActiveDocument().getBody();
```

```
var n = body.getNumChildren();
for (var i = 0; i < n; i++) {
    var el = body.getChild(i);
    if (el.getType() == DocumentApp.ElementType.PARAGRAPH) {
        var para = el.asParagraph();
        if (para.getHeading() !== DocumentApp.ParagraphHeading.NORMAL) {
            para.setBackgroundColor('#ffe599');
        }
    }
}
}
```

44. Find and Italicize All Quoted Text

```
function italicizeQuotes() {
    var body = DocumentApp.getActiveDocument().getBody();
    var found = body.findText('"[^"]+"', null,
{useRegularExpression:true});
    while (found) {
        found.getElement().setItalic(true);
        found = body.findText('"[^"]+"', found,
{useRegularExpression:true});
    }
}
```

45. Add a Hyperlink to a Selected Word

```
function addHyperlinkToSelection(url) {
    var selection = DocumentApp.getActiveDocument().getSelection();
    if (selection) {
        var elements = selection.getRangeElements();
        elements.forEach(function(el) {
            var elem = el.getElement().asText();
            elem.setLinkUrl(el.getStartOffset(), el.getEndOffsetInclusive(),
url);
    }
}
```

```
    });
}
// Example usage: addHyperlinkToSelection('https://google.com');
```

46. Replace All Tab Characters with Spaces

```
function replaceTabsWithSpaces() {
  var body = DocumentApp.getActiveDocument().getBody();
  var txt = body.getText().replace(/\t/g, ' ');
  body.setText(txt);
}
```

47. Remove All Underlines

```
function removeAllUnderlines() {
  var text = DocumentApp.getActiveDocument().getBody().editAsText();
  text.setUnderline(false);
}
```

48. List Paragraphs Longer Than 100 Characters

```
function longParagraphs() {
  var body = DocumentApp.getActiveDocument().getBody();
  var n = body.getNumChildren();
  for (var i = 0; i < n; i++) {
    var el = body.getChild(i);
    if (el.getType() == DocumentApp.ElementType.PARAGRAPH) {
      var txt = el.asParagraph().getText();
      if (txt.length > 100) Logger.log(txt);
    }
  }
}
```

49. Set All Text Color to Black

```
function setAllTextBlack() {
  var body = DocumentApp.getActiveDocument().getBody().editAsText();
  body.setForegroundColor('black');
}
```

50. Make All Headings Bold

```
function makeHeadingsBold() {
  var body = DocumentApp.getActiveDocument().getBody();
  var n = body.getNumChildren();
  for (var i = 0; i < n; i++) {
    var el = body.getChild(i);
    if (el.getType() == DocumentApp.ElementType.PARAGRAPH) {
      var para = el.asParagraph();
      if (para.getHeading() != DocumentApp.ParagraphHeading.NORMAL) {
        para.setBold(true);
      }
    }
  }
}
```

51. Log All Paragraph Styles

```
function logParagraphStyles() {
  var body = DocumentApp.getActiveDocument().getBody();
  for (var i = 0; i < body.getNumChildren(); i++) {
    var p = body.getChild(i);
    if (p.getType() == DocumentApp.ElementType.PARAGRAPH) {
      Logger.log(p.asParagraph().getHeading());
    }
  }
}
```

```
    }
}
```

52. Set Document Font Size to 12

```
function setFontSize12() {
  var text = DocumentApp.getActiveDocument().getBody().editAsText();
  text.setFontSize(12);
}
```

53. Remove All Strikethrough

```
function removeAllStrikethrough() {
  var text = DocumentApp.getActiveDocument().getBody().editAsText();
  text.setStrikethrough(false);
}
```

54. Highlight All Paragraphs Containing “Important”

```
function highlightImportant() {
  var body = DocumentApp.getActiveDocument().getBody();
  var n = body.getNumChildren();
  for (var i = 0; i < n; i++) {
    var el = body.getChild(i);
    if (el.getType() == DocumentApp.ElementType.PARAGRAPH) {
      if (el.asParagraph().getText().match(/important/i)) {
        el.asParagraph().setBackgroundColor('pink');
      }
    }
  }
}
```

55. Add Table with 3 Rows and 3 Columns

```
function add3x3Table() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  
  body.appendTable([['A1','B1','C1'],['A2','B2','C2'],['A3','B3','C3']])  
};  
}
```

56. Remove All Tables Except the First

```
function keepFirstTableOnly() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var foundFirst = false;  
  for (var i = body.getNumChildren() - 1; i >= 0; i--) {  
    var child = body.getChild(i);  
    if (child.getType() == DocumentApp.ElementType.TABLE) {  
      if (!foundFirst) {  
        foundFirst = true;  
      } else {  
        body.removeChild(child);  
      }  
    }  
  }  
}
```

57. Add Timestamp to End of Doc

```
function addTimestampEnd() {  
  var now = new Date();  
  var str = Utilities.formatDate(now, Session.getScriptTimeZone(),  
'yyyy-MM-dd HH:mm:ss');  
  DocumentApp.getActiveDocument().getBody().appendParagraph(str);  
}
```

58. Highlight All Numbers Over 100

```
function highlightLargeNumbers() {
  var body = DocumentApp.getActiveDocument().getBody();
  var found = body.findText('\b\d{3,}\b', null,
{useRegularExpression:true});
  while (found) {
    found.getElement().setBackgroundColor('#f4cccc');
    found = body.findText('\b\d{3,}\b', found,
{useRegularExpression:true});
  }
}
```

59. Insert Line Breaks After Every Sentence

```
function breakAfterSentences() {
  var body = DocumentApp.getActiveDocument().getBody();
  var text = body.getText().replace(/([.?!])\s/g, '$1\n');
  body.setText(text);
}
```

60. Add “Confidential” Watermark (Footer)

```
function addConfidentialFooter() {

  DocumentApp.getActiveDocument().addFooter().appendParagraph("CONFIDENTIAL");
}
```

61. Set All Paragraphs to Single Spacing

```
function singleSpacing() {
  var body = DocumentApp.getActiveDocument().getBody();
  for (var i = 0; i < body.getNumChildren(); i++) {
    var p = body.getChild(i);
    if (p.getType() == DocumentApp.ElementType.PARAGRAPH) {
      p.asParagraph().setLineSpacing(1);
    }
  }
}
```

62. Find All Paragraphs Starting with “Note:”

```
function findNotes() {
  var body = DocumentApp.getActiveDocument().getBody();
  for (var i = 0; i < body.getNumChildren(); i++) {
    var para = body.getChild(i);
    if (para.getType() == DocumentApp.ElementType.PARAGRAPH) {
      var text = para.asParagraph().getText();
      if (/^Note:/i.test(text)) Logger.log(text);
    }
  }
}
```

63. Convert All Text to Sentence Case

```
function toSentenceCase() {
  var body = DocumentApp.getActiveDocument().getBody();
  var text =
body.getText().toLowerCase().replace(/(^s*\w|[^.!?]\s*\w)/g,
function(c){return c.toUpperCase()});
  body.setText(text);
}
```

64. Insert a Blank Page at End

```
function insertBlankPage() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  body.appendPageBreak();  
}
```

65. Insert a Table of Figures Placeholder

```
function insertTableOfFigures() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  body.insertParagraph(0, "Table of  
Figures").setHeading(DocumentApp.ParagraphHeading.HEADING1);  
}
```

66. Add a Horizontal Line After Every Heading

```
function lineAfterHeading() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  for (var i = body.getNumChildren() - 1; i >= 0; i--) {  
    var para = body.getChild(i);  
    if (para.getType() == DocumentApp.ElementType.PARAGRAPH &&  
        para.asParagraph().getHeading() !=  
DocumentApp.ParagraphHeading.NORMAL) {  
      body.insertHorizontalRule(i+1);  
    }  
  }  
}
```

67. Highlight All Unique Words Longer Than 10 Letters

```
function highlightLongUniqueWords() {  
  var body = DocumentApp.getActiveDocument().getBody();
```

```
var text = body.getText();
var longWords = Array.from(new Set(text.match(/\b\w{11,}\b/g) || []));
longWords.forEach(function(word) {
  var found = body.findText('\b'+word+'\b');
  while (found) {
    found.getElement().setBackgroundColor('#d9ead3');
    found = body.findText('\b'+word+'\b', found);
  }
});
}
```

68. Remove All Images

```
function removeAllImages() {
  var body = DocumentApp.getActiveDocument().getBody();
  var imgs = body.getImages();
  for (var i = imgs.length - 1; i >= 0; i--) {
    var parent = imgs[i].getParent();
    if (parent) parent.removeChild(imgs[i]);
  }
}
```

69. Add a Cover Page

```
function addCoverPage() {
  var body = DocumentApp.getActiveDocument().getBody();
  body.insertParagraph(0, "My Document
Title").setHeading(DocumentApp.ParagraphHeading.TITLE);
  body.insertParagraph(1, "Author
Name").setHeading(DocumentApp.ParagraphHeading.HEADING2);
  body.insertParagraph(2,
 "").setHeading(DocumentApp.ParagraphHeading.NORMAL);
  body.insertPageBreak(3);
}
```

70. Randomize Paragraph Order

```
function randomizeParagraphs() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var paras = [];  
  for (var i = 0; i < body.getNumChildren(); i++) {  
    var el = body.getChild(i);  
    if (el.getType() == DocumentApp.ElementType.PARAGRAPH) {  
      paras.push(el.asParagraph().copy());  
    }  
  }  
  for (var i = body.getNumChildren() - 1; i >= 0; i--) {  
    if (body.getChild(i).getType() ==  
        DocumentApp.ElementType.PARAGRAPH)  
      body.removeChild(body.getChild(i));  
  }  
  paras.sort(function(){return 0.5-Math.random()});  
  paras.forEach(function(para) { body.appendParagraph(para.getText());  
});  
}
```

71. Replace All “lorem” With “ipsum”

```
function replaceLoremWithIpsum() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  body.replaceText('lorem', 'ipsum');  
}
```

72. List All Paragraphs Containing a Number

```
function paragraphsWithNumber() {  
  var body = DocumentApp.getActiveDocument().getBody();
```

```
for (var i = 0; i < body.getNumChildren(); i++) {
  var p = body.getChild(i);
  if (p.getType() == DocumentApp.ElementType.PARAGRAPH) {
    var txt = p.asParagraph().getText();
    if (/^\d/.test(txt)) Logger.log(txt);
  }
}
}
```

73. Delete All Paragraphs Containing “delete me”

```
function deleteParagraphsWithDeleteMe() {
  var body = DocumentApp.getActiveDocument().getBody();
  for (var i = body.getNumChildren() - 1; i >= 0; i--) {
    var para = body.getChild(i);
    if (para.getType() == DocumentApp.ElementType.PARAGRAPH) {
      if (/delete me/i.test(para.asParagraph().getText()))
        body.removeChild(para);
    }
  }
}
```

74. Make All Links Blue

```
function setAllLinksBlue() {
  var body = DocumentApp.getActiveDocument().getBody();
  var found = body.findText('https?://[^\\s]+', null,
{useRegularExpression:true});
  while (found) {
    found.getElement().setForegroundColor('blue');
    found = body.findText('https?://[^\\s]+', found,
{useRegularExpression:true});
  }
}
```

75. Number All Headings Sequentially

```
function numberHeadings() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var n = body.getNumChildren();  
  var count = 1;  
  for (var i = 0; i < n; i++) {  
    var el = body.getChild(i);  
    if (el.getType() == DocumentApp.ElementType.PARAGRAPH) {  
      var para = el.asParagraph();  
      if (para.getHeading() != DocumentApp.ParagraphHeading.NORMAL) {  
        para.setText(count + ". " + para.getText());  
        count++;  
      }  
    }  
  }  
}
```

76. Highlight Palindromes

```
function highlightPalindromes() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var text = body.getText();  
  var words = text.match(/\b\w+\b/g) || [];  
  words.forEach(function(word) {  
    if (word.length > 2 && word == word.split('').reverse().join(''))  
    {  
      var found = body.findText('\b' + word + '\b');  
      while (found) {  
        found.getElement().setBackgroundColor('#d9ead3');  
        found = body.findText('\b' + word + '\b', found);  
      }  
    }  
  });  
}
```

77. Log Document Creation Date

```
function logDocCreationDate() {  
  var doc = DocumentApp.getActiveDocument();  
  var file = DriveApp.getFileById(doc.getId());  
  Logger.log(file.getDateCreated());  
}
```

78. Remove All Formatting

```
function removeAllFormatting() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var text = body.getText();  
  body.setText(text); // Removes all formatting  
}
```

79. Duplicate the First Paragraph

```
function duplicateFirstParagraph() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var first = body.getChild(0);  
  if (first.getType() == DocumentApp.ElementType.PARAGRAPH) {  
    body.insertParagraph(1, first.asParagraph().getText());  
  }  
}
```

80. Move All Headings to Top

```
function moveHeadingsToTop() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  var headings = [];
```

```
for (var i = body.getNumChildren() - 1; i >= 0; i--) {
  var para = body.getChild(i);
  if (para.getType() == DocumentApp.ElementType.PARAGRAPH &&
      para.asParagraph().getHeading() !==
DocumentApp.ParagraphHeading.NORMAL) {
    headings.unshift(para.asParagraph().getText());
    body.removeChild(para);
  }
}
headings.forEach(function(h, idx) {
  body.insertParagraph(idx,
h).setHeading(DocumentApp.ParagraphHeading.HEADING1);
});
}
```

81. Insert Blank Paragraph After Each Heading

```
function blankAfterHeadings() {
  var body = DocumentApp.getActiveDocument().getBody();
  for (var i = body.getNumChildren() - 1; i >= 0; i--) {
    var para = body.getChild(i);
    if (para.getType() == DocumentApp.ElementType.PARAGRAPH &&
        para.asParagraph().getHeading() !==
DocumentApp.ParagraphHeading.NORMAL) {
      body.insertParagraph(i + 1, '');
    }
  }
}
```

82. Replace All Exclamation Marks With Periods

```
function exclamationsToPeriods() {
  var body = DocumentApp.getActiveDocument().getBody();
  var text = body.getText().replace(/!/g, '.');
  body.setText(text);
```

```
}
```

83. Add “Summary” Section at End

```
function addSummarySection() {
  var body = DocumentApp.getActiveDocument().getBody();

  body.appendParagraph('Summary').setHeading(DocumentApp.ParagraphHeading.HEADING1);
}
```

84. Find and Bold All Proper Nouns

```
function boldProperNouns() {
  var body = DocumentApp.getActiveDocument().getBody();
  var found = body.findText('\b[A-Z][a-z]+\b', null,
{useRegularExpression:true});
  while (found) {
    found.getElement().setBold(true);
    found = body.findText('\b[A-Z][a-z]+\b', found,
{useRegularExpression:true});
  }
}
```

85. Insert Document Word Count at the Top

```
function insertWordCountAtTop() {
  var body = DocumentApp.getActiveDocument().getBody();
  var count = body.getText().match(/\b\S+\b/g)? .length || 0;
  body.insertParagraph(0, 'Word Count: ' + count);
}
```

86. Insert a Divider Line After Each Table

```
function lineAfterEachTable() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  for (var i = body.getNumChildren() - 1; i >= 0; i--) {  
    var el = body.getChild(i);  
    if (el.getType() == DocumentApp.ElementType.TABLE) {  
      body.insertHorizontalRule(i + 1);  
    }  
  }  
}
```

87. Log All Paragraph Indentations

```
function logIndentations() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  for (var i = 0; i < body.getNumChildren(); i++) {  
    var p = body.getChild(i);  
    if (p.getType() == DocumentApp.ElementType.PARAGRAPH) {  
      Logger.log(p.asParagraph().getIndentStart());  
    }  
  }  
}
```

88. Remove All Empty Headings

```
function removeEmptyHeadings() {  
  var body = DocumentApp.getActiveDocument().getBody();  
  for (var i = body.getNumChildren() - 1; i >= 0; i--) {  
    var para = body.getChild(i);  
    if (para.getType() == DocumentApp.ElementType.PARAGRAPH &&  
        para.asParagraph().getHeading() !=  
        DocumentApp.ParagraphHeading.NORMAL &&  
        !para.asParagraph().getText().trim()) {  
      body.removeChild(para);  
    }  
  }  
}
```

```
        }
    }
}
```

89. Highlight All Dates (YYYY-MM-DD)

```
function highlightDates() {
  var body = DocumentApp.getActiveDocument().getBody();
  var found = body.findText('\b\d{4}-\d{2}-\d{2}\b', null,
{useRegularExpression:true});
  while (found) {
    found.getElement().setBackgroundColor('#fce5cd');
    found = body.findText('\b\d{4}-\d{2}-\d{2}\b', found,
{useRegularExpression:true});
  }
}
```

90. List All Tables and Their Size

```
function logTableSizes() {
  var body = DocumentApp.getActiveDocument().getBody();
  for (var i = 0; i < body.getNumChildren(); i++) {
    if (body.getChild(i).getType() == DocumentApp.ElementType.TABLE) {
      var table = body.getChild(i).asTable();
      Logger.log('Table at index ' + i + ': ' + table.getNumRows() + ' rows x ' + table.getRow(0).getNumCells() + ' cols');
    }
  }
}
```

91. Make All Text 1.5 Line Spaced

```
function onePointFiveSpacing() {
```

```
var body = DocumentApp.getActiveDocument().getBody();
for (var i = 0; i < body.getNumChildren(); i++) {
  var p = body.getChild(i);
  if (p.getType() == DocumentApp.ElementType.PARAGRAPH) {
    p.asParagraph().setLineSpacing(1.5);
  }
}
}
```

92. Highlight All Questions

```
function highlightQuestions() {
  var body = DocumentApp.getActiveDocument().getBody();
  var found = body.findText('[^?]+\?', null,
{useRegularExpression:true});
  while (found) {
    found.getElement().setBackgroundColor('#d9ead3');
    found = body.findText('[^?]+\?', found,
{useRegularExpression:true});
  }
}
```

93. Insert an Index of All Headings at Top

```
function insertHeadingsIndex() {
  var body = DocumentApp.getActiveDocument().getBody();
  var headings = [];
  for (var i = 0; i < body.getNumChildren(); i++) {
    var para = body.getChild(i);
    if (para.getType() == DocumentApp.ElementType.PARAGRAPH &&
        para.asParagraph().getHeading() !==
DocumentApp.ParagraphHeading.NORMAL) {
      headings.push(para.asParagraph().getText());
    }
  }
}
```

```
    body.insertParagraph(0, "Index: " + headings.join(" | "));
}
```

94. Log All Paragraph Alignments

```
function logAlignments() {
  var body = DocumentApp.getActiveDocument().getBody();
  for (var i = 0; i < body.getNumChildren(); i++) {
    var p = body.getChild(i);
    if (p.getType() == DocumentApp.ElementType.PARAGRAPH) {
      Logger.log(p.asParagraph().getAlignment());
    }
  }
}
```

95. Convert All Uppercase Words to Lowercase

```
function uppercaseToLowercase() {
  var body = DocumentApp.getActiveDocument().getBody();
  var text = body.getText().replace(/\b[A-Z]{2,}\b/g,
function(w){return w.toLowerCase();});
  body.setText(text);
}
```

96. Highlight All Headings with Custom Color

```
function highlightHeadingsColor() {
  var body = DocumentApp.getActiveDocument().getBody();
  for (var i = 0; i < body.getNumChildren(); i++) {
    var para = body.getChild(i);
    if (para.getType() == DocumentApp.ElementType.PARAGRAPH &&
        para.asParagraph().getHeading() !=
DocumentApp.ParagraphHeading.NORMAL) {
```

```
        para.asParagraph().setBackgroundColor('#b6d7a8');
    }
}
}
```

97. List All Section Breaks

```
function logSectionBreaks() {
  var body = DocumentApp.getActiveDocument().getBody();
  for (var i = 0; i < body.getNumChildren(); i++) {
    var el = body.getChild(i);
    if (el.getType() == DocumentApp.ElementType.SECTION_BREAK) {
      Logger.log('Section break at index ' + i);
    }
  }
}
```

98. Highlight All Capitalized Words

```
function highlightCapWords() {
  var body = DocumentApp.getActiveDocument().getBody();
  var found = body.findText('\b[A-Z][A-Z]+\b', null,
{useRegularExpression:true});
  while (found) {
    found.getElement().setBackgroundColor('#f4cccc');
    found = body.findText('\b[A-Z][A-Z]+\b', found,
{useRegularExpression:true});
  }
}
```

99. Remove All Paragraph Background Colors

```
function clearAllBackgroundColors() {
```

```
var body = DocumentApp.getActiveDocument().getBody();
for (var i = 0; i < body.getNumChildren(); i++) {
  var para = body.getChild(i);
  if (para.getType() == DocumentApp.ElementType.PARAGRAPH) {
    para.asParagraph().setBackgroundColor(null);
  }
}
}
```

100. Add "Appendix" Section at End

```
function addAppendixSection() {
  var body = DocumentApp.getActiveDocument().getBody();

  body.appendParagraph('Appendix').setHeading(DocumentApp.ParagraphHeading.HEADING1);
}
```