



# DocumentApp Class

The document service creates and opens Documents that can be edited.

<b>DocumentApp Class</b>	<b>1</b>
<b>The document service creates and opens Documents that can be edited.</b>	<b>1</b>
<b>DocumentApp:</b>	<b>3</b>
Script Editor:	3
Creating Documents:	3
Manipulating Text and Formatting:	4
Working with Tables:	4
Managing Document Properties:	4
Hyperlinks and URLs:	4
Event Triggers:	5
Integration with Other Services:	5
<b>Coding Exercises</b>	<b>5</b>
Exercise 1: Create a New Document	5
Exercise 2: Insert Text into a Document	6
Exercise 3: Format Text in Document	7
Exercise 4: Add a Table to Document	7
Exercise 5: Replace Text in Document	8
Exercise 6: Set Document Properties	9
Exercise 7: Delete a Paragraph	9

Learn more about JavaScript with Examples and Source Code Laurence Svekis  
Courses <https://basescripts.com/>

Exercise 8: Create a Hyperlink	10
Exercise 9: Count Words in Document	11
Exercise 10: Protect Document	11
<b>Quiz Questions:</b>	<b>12</b>
Question: How do you create a new Google Document using Google Apps Script?	13
Question: Which method is used to insert text into a Google Document?	13
Question: How can you format text in a Google Document using Google Apps Script?	13
Question: What function is used to add a table to a Google Document?	14
Question: How do you replace specific text in a Google Document?	14
Question: What method is used to set the title of a Google Document?	15
Question: How can you delete a specific paragraph from a Google Document?	15
Question: What is the correct way to insert a hyperlink into a Google Document?	16
Question: How do you count the number of words in a Google Document using Google Apps Script?	16
Question: What function is used to protect a Google Document to restrict editing?	17
<b>Test Your knowledge Quiz</b>	<b>17</b>

Google Apps Script is a scripting language that allows developers to automate tasks and extend the functionality of various Google Workspace applications, including Google Docs, Sheets, Slides, and more. It provides a powerful way to customize and enhance Google Workspace products, making them more tailored to specific needs.

DocumentApp is a specific service within Google Apps Script that focuses on manipulating and automating actions within Google Documents. With DocumentApp, developers can programmatically create, modify, and interact with Google Docs, allowing for the automation of document-related tasks.

DocumentApp:

### **Script Editor:**

Google Apps Script is typically written and executed in the Script Editor, an integrated development environment (IDE) provided by Google. The Script Editor allows developers to write, test, and debug their scripts.

### **Creating Documents:**

Developers can use Google Apps Script to dynamically create new Google Documents. This is particularly useful for generating documents based on templates, form responses, or any other dynamic data source.

## **Manipulating Text and Formatting:**

DocumentApp provides methods for inserting, replacing, and formatting text within a document. This includes setting styles such as bold, italic, underline, font size, and more.

## **Working with Tables:**

Developers can add tables to documents and manipulate their content using DocumentApp. This allows for the creation of structured data, such as reports or grids, within a document.

## **Managing Document Properties:**

Document properties, such as title and description, can be set or modified using DocumentApp. This enables customization of document metadata.

## **Hyperlinks and URLs:**

DocumentApp allows for the insertion of hyperlinks into documents. This is useful for creating interactive documents that link to external resources.

## **Event Triggers:**

Google Apps Script supports triggers that allow scripts to run automatically based on events such as document edits or form submissions.

## **Integration with Other Services:**

Google Apps Script can be integrated with other Google services and APIs, providing a seamless way to extend functionality across the Google Workspace.

In summary, Google Apps Script - DocumentApp offers a versatile set of tools for automating tasks and creating customized solutions within Google Documents. Whether it's generating reports, formatting text, or managing document properties, developers can leverage the power of scripting to enhance the capabilities of Google Docs.

## **Coding Exercises**

### **Exercise 1: Create a New Document**

Objective: Write a script to create a new Google Document.

Steps:

Learn more about JavaScript with Examples and Source Code Laurence Svekis  
Courses <https://basescripts.com/>

1. Open Google Drive.
2. Create a new Google Document.
3. Open the Script Editor.
4. Write a script to create a new document.
5. Run the script.

Code:

```
function createNewDocument() {  
  var newDoc = DocumentApp.create('My New Document');  
  Logger.log('Document created: ' + newDoc.getUrl());  
}
```

## **Exercise 2: Insert Text into a Document**

Objective: Insert a specific text into an existing document.

Steps:

1. Open the Script Editor.
2. Write a script to open an existing document and insert text.
3. Run the script.

Code:

```
function insertTextIntoDocument() {  
  var doc = DocumentApp.openById('DOCUMENT_ID'); // Replace  
  with your document ID  
  doc.getBody().appendParagraph('This is a new paragraph.');
```

### **Exercise 3: Format Text in Document**

Objective: Apply formatting (bold, italic, underline) to specific text in a document.

Steps:

1. Open the Script Editor.
2. Write a script to format text.
3. Run the script.

Code:

```
function formatTextInDocument() {  
  var doc = DocumentApp.openById('DOCUMENT_ID'); // Replace  
  with your document ID  
  var text = doc.getBody().getText();  
  var newText = text.replace('specific text', '**specific text**');  
  doc.getBody().setText(newText);  
}
```

### **Exercise 4: Add a Table to Document**

Objective: Insert a table into a document.

Steps:

1. Open the Script Editor.

Learn more about JavaScript with Examples and Source Code Laurence Svekis  
Courses <https://basescripts.com/>

2. Write a script to add a table.
3. Run the script.

Code:

```
function addTableToDocument() {  
  var doc = DocumentApp.openById('DOCUMENT_ID'); // Replace  
with your document ID  
  var table = doc.getBody().appendTable([[ 'Cell 1', 'Cell 2'], [ 'Cell  
3', 'Cell 4']]);  
}
```

## Exercise 5: Replace Text in Document

Objective: Replace specific text with new text in a document.

Steps:

1. Open the Script Editor.
2. Write a script to find and replace text.
3. Run the script.

Code:

```
function replaceTextInDocument() {  
  var doc = DocumentApp.openById('DOCUMENT_ID'); // Replace  
with your document ID  
  var body = doc.getBody();  
  var newText = body.getText().replace('old text', 'new text');  
  body.clear(); // Clear existing content
```

Learn more about JavaScript with Examples and Source Code Laurence Svekis  
Courses <https://basescripts.com/>



```
body.appendParagraph(newText);  
}
```

## Exercise 6: Set Document Properties

Objective: Set properties (title, description) for a document.

Steps:

1. Open the Script Editor.
2. Write a script to set document properties.
3. Run the script.

Code:

```
function setDocumentProperties() {  
  var doc = DocumentApp.openById('DOCUMENT_ID'); // Replace  
  with your document ID  
  doc.setTitle('New Title');  
  doc.setDescription('This is a description for the document.');
```

```
}
```

## Exercise 7: Delete a Paragraph

Objective: Delete a specific paragraph from a document.

Steps:

1. Open the Script Editor.

2. Write a script to delete a paragraph.
3. Run the script.

Code:

```
function deleteParagraph() {  
  var doc = DocumentApp.openById('DOCUMENT_ID'); // Replace  
with your document ID  
  var body = doc.getBody();  
  var paragraphs = body.getParagraphs();  
  paragraphs[1].remove(); // Delete the second paragraph  
}
```

## **Exercise 8: Create a Hyperlink**

Objective: Insert a hyperlink into a document.

Steps:

1. Open the Script Editor.
2. Write a script to add a hyperlink.
3. Run the script.

Code:

```
function insertHyperlink() {  
  var doc = DocumentApp.openById('DOCUMENT_ID'); // Replace  
with your document ID  
  var text = 'Click here';  
  var url = 'https://www.example.com';
```

Learn more about JavaScript with Examples and Source Code Laurence Svekis  
Courses <https://basescripts.com/>

```
doc.getBody().appendParagraph(text).setLinkUrl(url);
}
```

## Exercise 9: Count Words in Document

Objective: Count the number of words in a document.

Steps:

1. Open the Script Editor.
2. Write a script to count words.
3. Run the script.

Code:

```
function countWordsInDocument() {
  var doc = DocumentApp.openById('DOCUMENT_ID'); // Replace
  with your document ID
  var wordCount = doc.getBody().getText().split(/\s+/).length;
  Logger.log('Word count: ' + wordCount);
}
```

## Exercise 10: Protect Document

Objective: Protect a document to restrict editing.

Steps:

1. Open the Script Editor.

2. Write a script to protect the document.
3. Run the script.

Code:

```
function protectDocument() {  
  var doc = DocumentApp.openById('DOCUMENT_ID'); // Replace  
  with your document ID  
  var protection = doc.addEditor('user@example.com'); // Replace  
  with the email of the user to be granted editing access  
  protection.remove();  
}
```

Make sure to replace 'DOCUMENT\_ID' and other placeholder values with the actual values relevant to your use case. These exercises cover various DocumentApp functionalities and provide a good starting point for working with Google Apps Script and DocumentApp.

## Quiz Questions:

Creating a New Document (Exercise 1-5)

**Question: How do you create a new Google Document using Google Apps Script?**

- A) new DocumentApp.createDocument('Document Name');
- B) DocumentApp.create('Document Name');
- C) createDocument('Document Name');
- D) create.newDocument('Document Name');

Answer: B) DocumentApp.create('Document Name');

**Question: Which method is used to insert text into a Google Document?**

- A) insertText('Text');
- B) addText('Text');
- C) doc.insert('Text');
- D) getBody().appendParagraph('Text');

Answer: D) getBody().appendParagraph('Text');

**Question: How can you format text in a Google Document using Google Apps Script?**

- A) setFormat('Text', {bold: true});

- B) `formatText('Text', {bold: true});`
- C) `getBody().setBold(true);`
- D) `getBody().setTextAttributes(startOffset, endOffset, {bold: true});`

Answer: D) `getBody().setTextAttributes(startOffset, endOffset, {bold: true});`

**Question: What function is used to add a table to a Google Document?**

- A) `appendTable(rows, columns);`
- B) `createTable(rows, columns);`
- C) `insertTable(rows, columns);`
- D) `getBody().appendTable(data);`

Answer: D) `getBody().appendTable(data);`

**Question: How do you replace specific text in a Google Document?**

- A) `replaceText('Old', 'New');`
- B) `searchAndReplace('Old', 'New');`
- C) `getBody().replace('Old', 'New');`

D) `replace('Old', 'New');`

Answer: C) `getBody().replace('Old', 'New');`

**Question: What method is used to set the title of a Google Document?**

A) `setTitle('New Title');`

B) `setDocumentTitle('New Title');`

C) `doc.title('New Title');`

D) `set('Title', 'New Title');`

Answer: A) `setTitle('New Title');`

**Question: How can you delete a specific paragraph from a Google Document?**

A) `deleteParagraph(2);`

B) `removeParagraph(2);`

C) `getBody().deleteParagraph(2);`

D) `getBody().getParagraph(2).delete();`

Answer: D) `getBody().getParagraph(2).delete();`

**Question: What is the correct way to insert a hyperlink into a Google Document?**

- A) `appendLink('Click here', 'https://www.example.com');`
- B) `insertHyperlink('Click here', 'https://www.example.com');`
- C) `getBody().insertLink('Click here', 'https://www.example.com');`
- D) `getBody().appendParagraph('Click here').setLinkUrl(https://www.example.com);`

Answer: D) `getBody().appendParagraph('Click here').setLinkUrl(https://www.example.com);`

**Question: How do you count the number of words in a Google Document using Google Apps Script?**

- A) `countWords();`
- B) `getWordCount();`
- C) `wordCount();`
- D) `getBody().getText().split(/\s+/).length;`

Answer: D) `getBody().getText().split(/\s+/).length;`



## Question: What function is used to protect a Google Document to restrict editing?

- A) restrictEditing();
- B) protectDocument();
- C) setEditingPermissions();
- D) secureDocument();

Answer: B) protectDocument();

## Test Your knowledge Quiz

Creating and Manipulating Documents (Questions 1-15):

Question: How do you create a new Google Document using Google Apps Script?

- A) DocumentApp.createDocument('Document Name');
- B) createNewDocument('Document Name');
- C) DocumentApp.create('Document Name');
- D) new DocumentApp.Document('Document Name');

Answer: C) DocumentApp.create('Document Name');

Question: Which method is used to insert text into a Google Document?

- A) insertText('Text');
- B) doc.getBody().appendText('Text');
- C) getBody().insertParagraph('Text');
- D) getBody().appendParagraph('Text');

Answer: D) `getBody().appendParagraph('Text');`

Question: How can you format a specific text in a Google Document to be bold?

- A) `setBoldText('Text');`
- B) `formatText('Text', {bold: true});`
- C) `getBody().setBold(true);`
- D) `doc.format('Text', 'bold');`

Answer: B) `formatText('Text', {bold: true});`

Question: What function is used to add a table to a Google Document?

- A) `createTable(rows, columns);`
- B) `addTable(rows, columns);`
- C) `doc.createTable(rows, columns);`
- D) `getBody().appendTable(data);`

Answer: D) `getBody().appendTable(data);`

Question: How do you replace specific text in a Google Document?

- A) `replaceText('Old', 'New');`
- B) `searchAndReplace('Old', 'New');`
- C) `getBody().replace('Old', 'New');`
- D) `replace('Old', 'New');`

Answer: C) `getBody().replace('Old', 'New');`

Question: What method is used to set the title of a Google Document?

- A) `setTitle('New Title');`
- B) `setDocumentTitle('New Title');`
- C) `doc.title('New Title');`
- D) `set('Title', 'New Title');`

Answer: A) setTitle('New Title');

Question: How can you delete a specific paragraph from a Google Document?

- A) deleteParagraph(2);
- B) removeParagraph(2);
- C) getBody().deleteParagraph(2);
- D) getBody().getParagraph(2).remove();

Answer: D) getBody().getParagraph(2).remove();

Question: What is the correct way to insert a hyperlink into a Google Document?

- A) appendLink('Click here', 'https://www.example.com');
- B) insertHyperlink('Click here', 'https://www.example.com');
- C) getBody().insertLink('Click here', 'https://www.example.com');
- D) getBody().appendParagraph('Click

here').setLinkUrl('https://www.example.com');

Answer: D) getBody().appendParagraph('Click here').setLinkUrl('https://www.example.com');

Question: How do you count the number of words in a Google Document using Google Apps Script?

- A) countWords();
- B) getWordCount();
- C) wordCount();
- D) getBody().getText().split(/\s+/).length;

Answer: D) getBody().getText().split(/\s+/).length;

Question: What function is used to protect a Google Document to restrict editing?

- A) restrictEditing();

- B) `protectDocument();`
- C) `setEditingPermissions();`
- D) `secureDocument();`

Answer: B) `protectDocument();`

Question: Which method is used to clear all content from the body of a Google Document?

- A) `clearContent();`
- B) `deleteContent();`
- C) `removeAll();`
- D) `getBody().clear();`

Answer: D) `getBody().clear();`

Question: What is the purpose of the `getActiveDocument()` method?

- A) It retrieves the most recently opened document.
- B) It creates a new active document.
- C) It gets the document currently being edited.
- D) It opens the last saved document.

Answer: C) It gets the document currently being edited.

Question: How do you append a new paragraph with a heading style to a Google Document?

- A) `getBody().appendHeading('Heading Text');`
- B) `getBody().appendParagraph('Heading Text').setHeadingStyle(DocumentApp.ParagraphHeading.HEADING1);`
- C) `appendParagraph('Heading Text', {heading: true});`
- D) `appendHeading('Heading Text', 1);`

Answer: B) `getBody().appendParagraph('Heading Text').setHeadingStyle(DocumentApp.ParagraphHeading.HEADING1);`

Question: What method is used to insert an image into a Google Document?

- A) `insertImage('URL');`
- B) `getBody().addImage('URL');`
- C) `getImage('URL');`
- D) `appendImage('URL');`

Answer: B) `getBody().addImage('URL');`

Question: How can you set line spacing in a paragraph within a Google Document?

- A) `getBody().setLineSpacing(1.5);`
- B) `setParagraphLineSpacing('Paragraph Text', 1.5);`
- C) `setLineSpacing('Paragraph Text', 1.5);`
- D) `appendParagraph('Paragraph Text').setLineSpacing(1.5);`

Answer: A) `getBody().setLineSpacing(1.5);`

Managing Document Properties and Advanced Concepts (Questions 16-25):

Question: What is the purpose of the `getDescription()` method in Google Apps Script?

- A) It sets the description of a document.
- B) It retrieves the description of a document.
- C) It appends a description paragraph to a document.
- D) It removes the description of a document.

Answer: B) It retrieves the description of a document.

Question: How can you retrieve the URL of a Google Document using Google Apps Script?

Learn more about JavaScript with Examples and Source Code Laurence Svekis  
Courses <https://basescripts.com/>

- A) `doc.getLink();`
- B) `getDocumentURL();`
- C) `doc.getUrl();`
- D) `getUrl(doc);`

Answer: C) `doc.getUrl();`

Question: What is the purpose of the `removeEditor(email)` method in `DocumentApp`?

- A) It deletes the editor from the document.
- B) It removes editing permissions for the specified email.
- C) It clears all editors from the document.
- D) It prevents editors from accessing the document.

Answer: B) It removes editing permissions for the specified email.

Question: Which method is used to create a named range in a Google Document?

- A) `createNamedRange('Range Name');`
- B) `doc.getNamedRange('Range Name');`
- C) `getBody().addNamedRange('Range Name');`
- D) `addNamedRange('Range Name');`

Answer: C) `getBody().addNamedRange('Range Name');`

Question: How can you iterate through all paragraphs in the body of a Google Document?

- A) `forEachParagraph(callbackFunction);`
- B) `for (var i = 0; i < getBody().getParagraphs().length; i++) {}`
- C) `doc.getBody().forEachParagraph(callbackFunction);`
- D) `for (var paragraph in getBody().getParagraphs()) {}`

Answer: C) `doc.getBody().forEachParagraph(callbackFunction);`

Question: Which event trigger is used to run a script when a Google Document is opened?

- A) `onOpen();`
- B) `onLoad();`
- C) `onEdit();`
- D) `onDocumentOpen();`

Answer: A) `onOpen();`

Question: How do you set the background color of a specific paragraph in a Google Document?

- A) `setBackgroundColor('Color');`
- B) `getBody().setParagraphBackgroundColor('Color');`
- C) `doc.getBody().getParagraph(1).setBackgroundColor('Color');`
- D) `getBody().getParagraph(1).setBackgroundColor('Color');`

Answer: D) `getBody().getParagraph(1).setBackgroundColor('Color');`

Question: What is the purpose of the `getChild(childIndex)` method in `DocumentApp`?

- A) It adds a child element to the document.
- B) It retrieves the child at the specified index.
- C) It removes the child at the specified index.
- D) It counts the number of children in the document.

Answer: B) It retrieves the child at the specified index.

Question: How do you retrieve the text of a specific paragraph in a Google Document?

- A) `getTextOfParagraph(1);`
- B) `doc.getBody().getParagraph(1).getText();`
- C) `getParagraphText(1);`
- D) `getBody().getTextOfParagraph(1);`

Answer: B) `doc.getBody().getParagraph(1).getText();`

Question: What method is used to insert a page break in a Google Document?

- A) `insertPageBreak();`
- B) `getBody().appendPageBreak();`
- C) `addPageBreak();`
- D) `appendPageBreak();`

Answer: B) `getBody().appendPageBreak();`

Scripting and Advanced Topics (Questions 26-35):

Question: How can you create a custom menu in Google Docs using Google Apps Script?

- A) `createMenu('Custom Menu');`
- B) `addMenu('Custom Menu');`
- C) `DocumentApp.createMenu('Custom Menu');`
- D) `createCustomMenu('Custom Menu');`

Answer: C) `DocumentApp.createMenu('Custom Menu');`

Question: Which method is used to execute a function when a menu item is selected?

- A) `onMenuSelect(callbackFunction);`
- B) `executeFunction(callbackFunction);`
- C) `getMenu().addItem('Item', callbackFunction);`



D) getMenu().select('Item', callbackFunction);

Answer: C) getMenu().addItem('Item', callbackFunction);

Question: How can you set a named range to cover a specific range of elements in a Google Document?

A) getBody().setNamedRange('Range Name', start, end);

B) getNamedRange('Range Name').setRange(start, end);

C) getBody().getRange(start, end).setNamedRange('Range Name');

D) addNamedRange('Range Name', start, end);

Answer: B) getNamedRange('Range Name').setRange(start, end);

Question: What is the purpose of the getCursor() method in Google Apps Script?

A) It retrieves the cursor position in the document.

B) It sets the cursor to a specific location.

C) It adds a cursor element to the document.

D) It clears the cursor from the document.

Answer: A) It retrieves the cursor position in the document.

Question: How can you insert a horizontal rule (line) in a Google Document?

A) appendHorizontalRule();

B) getBody().insertHorizontalRule();

C) addHorizontalRule();

D) insertHorizontalRule();

Answer: B) getBody().insertHorizontalRule();

Question: What is the purpose of the getLinkUrl() method in Google Apps Script?

A) It retrieves the URL of a hyperlink in the document.

B) It sets the URL for a hyperlink.

C) It adds a link to a specific element in the document.

D) It removes a hyperlink from the document.

Answer: A) It retrieves the URL of a hyperlink in the document.

Question: How do you add a comment to a specific range in a Google Document?

A) `getBody().addComment('Comment', start, end);`

B) `addComment('Comment', start, end);`

C) `getCommentRange(start, end).addComment('Comment');`

D) `getRange(start, end).addComment('Comment');`

Answer: C) `getCommentRange(start, end).addComment('Comment');`

Question: What is the purpose of the `getBookmark(id)` method in `DocumentApp`?

A) It retrieves the bookmark with the specified ID.

B) It creates a new bookmark with the specified ID.

C) It adds a bookmark to the document.

D) It removes the bookmark with the specified ID.

Answer: A) It retrieves the bookmark with the specified ID.