Apps Script GmailApp

🌟 Revolutionizing Email Management with Google Apps Script: Mastering GmailApp! 🌟

Question 1: Sending an Email

Question: How do you send a basic email using Google Apps Script?

Answer:

```
function sendEmail() {
 GmailApp.sendEmail('recipient@example.com', 'Subject', 'Hello, this is the email body.');
```

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses https://basescripts.com/

}

Explanation: This script uses GmailApp.sendEmail() to send an email to 'recipient@example.com' with a subject and body text. It's a straightforward way to send emails programmatically.

Question 2: Searching Emails

Question: How can you search for emails with a specific subject?

Answer:

```
function searchEmails() {
 var threads = GmailApp.search('subject:"Your Subject Here"');
 return threads;
}
```

Explanation: This code snippet uses GmailApp.search() to find all email threads with a specific subject. The search query follows Gmail's search syntax.

Question 3: Applying Labels to Emails

Question: How do you apply a label to an email thread?

Answer:

```
function applyLabelToEmail(threadId, labelName) {
```

```
  var thread = GmailApp.getThreadById(threadId);
  var label = GmailApp.getUserLabelByName(labelName);
  if (!label) {
  label = GmailApp.createLabel(labelName);
  }
  thread.addLabel(label);
}
```

Explanation: This function adds a label to an email thread. It checks if the label exists, creates it if it doesn't, and then applies it to the thread.

Question 4: Marking Emails as Read

Question: How can you mark an email as read?

Answer:

```
function markEmailAsRead(threadId) {
  var thread = GmailApp.getThreadById(threadId);
  thread.markRead();
}
```

Explanation: This script marks a specific email thread as read using markRead() on the thread object.

Question 5: Retrieving Unread Emails

Question: How do you get a list of unread emails?

Answer:

```
function getUnreadEmails() {
 var threads = GmailApp.search('is:unread');
 return threads;
}
```

Explanation: This function fetches all unread email threads using the search query 'is:unread'. It returns an array of thread objects.

Question 6: Sending an Email with Attachment

Question: How can you send an email with an attachment?

Answer:

```
function sendEmailWithAttachment() {
 var file = DriveApp.getFileById('YOUR_FILE_ID'); // Replace with your file ID
 GmailApp.sendEmail('recipient@example.com', 'Subject', 'Email body', {
 attachments: [file.getAs(MimeType.PDF)] // Assuming the file is a PDF
 });
}
```

Explanation: This script sends an email with a file attachment. The file is fetched from Google Drive using its ID, and attached as a PDF.

Question 7: Creating a Draft Email

Question: How do you create a draft email?

Answer:

```
function createDraft() {
 GmailApp.createDraft('recipient@example.com', 'Draft Subject', 'This is the draft body.');
}
```

Explanation: This function creates a new email draft with a specified recipient, subject, and body.

Question 8: Deleting an Email Thread

Question: How can you delete an email thread?

Answer:

```
function deleteEmailThread(threadId) {
 var thread = GmailApp.getThreadById(threadId);
 thread.moveToTrash();
}
```

Explanation: This code snippet moves an email thread to the trash using moveToTrash() on the thread object.

Question 9: Counting Emails from a Sender

Question: How do you count the number of emails received from a specific email address?

Answer:

```
function countEmailsFromSender(senderEmail) {
 var threads = GmailApp.search('from:' + senderEmail);
 return threads.length;
}
```

Explanation: This function counts the number of email threads from a specific sender using a 'from:' search query.

Question 10: Fetching Email Body Content

Question: How can you fetch the body content of the latest email in a thread?

Answer:

```
function fetchLatestEmailBody(threadId) {
 var thread = GmailApp.getThreadById(threadId);
 var messages = thread.getMessages();
```

```
  var latestMessage = messages[messages.length - 1];
  return latestMessage.getBody();
}
```

Explanation: This script retrieves the latest message from an email thread and returns its body content. It accesses the last message in the thread's message array.