# Creating a Gmail Add-on

**GSP249**

Google Cloud Self-Paced Labs

# Overview

Gmail add-ons automate tasks within Gmail, saving time and effort for users. Gmail add-ons can examine incoming messages, then act on these messages in various ways, such as:

- Displaying additional information for the user in the Gmail UI.

- Connecting to non-Google services, to retrieve information or take other actions.

- Providing an interactive interface to allow the user to control the add-on or send information to another service.

## What you'll do

In this hands-on lab you create a Gmail Add-on that allows you to quickly change the labels of an email thread.

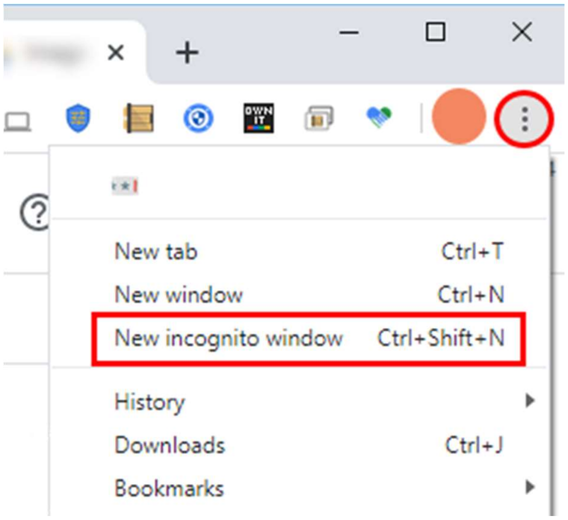
# Setup and requirements

## What you need

- An internet browser. Chrome browser is recommended.
- Time. 10 minutes to read through steps in the lab. 30 minutes to step through the hands-on practice lab. Once you start this lab, you cannot pause and return later.
All work for this lab is done in your Qwiklab G Suite account. If you have a different G Suite or Google account open, **sign out** before you start this lab. If you are signed into more than one account, you get the message "Server error is not authorized to access Admin Console of .... Note: Multi-Login is not supported for resellers in Admin Console`".

**Note:** If you are using a Pixelbook, run this lab in an Incognito window. To do this:

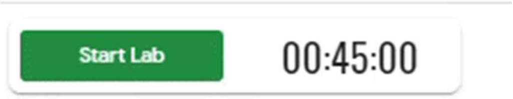
In the Chrome browser, click on the three dots next to your user picture, then select **New incognito window**.



## Start your lab

When you are ready, click **Start Lab**. If you need to pay for the lab, a pop-up opens for you to select your payment method.

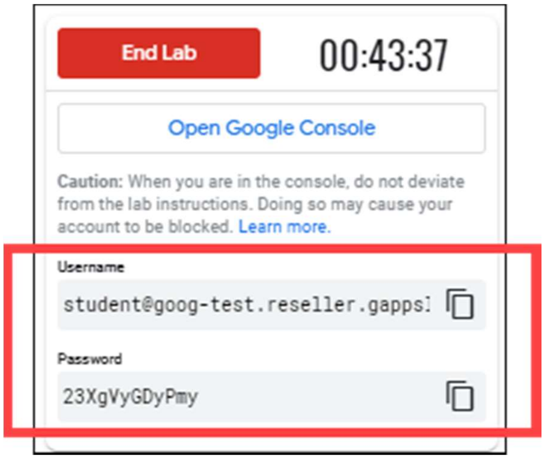
You can track your lab's setup progress with the status message in the blue box.



When the lab is set up, you will see the button to **Open Google Console** and the temporary credentials you must use for this lab.

## Username and Password

Find the Username and Password you need to access your Qwiklab's Google account under the button to **Open Google Console**.



Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. Learn more.

Username

student@goog-test.reseller.gappsl

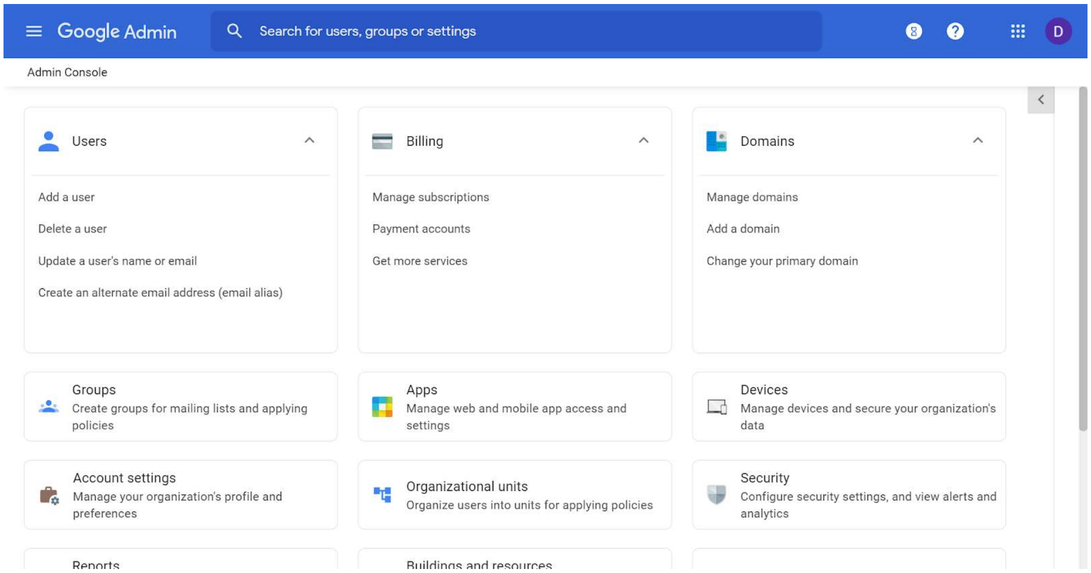
Password

23XgVyGDyPmy

## Sign in to Google Workspace Admin Console

Visit admin.google.com or click **Open Google Console** in the upper-left panel and enter the username and password located in the connection details panel
**Accept** the Google Workspace via Reseller Agreement and **Accept Terms of Service** to accept the Supplemental Terms and Conditions For the Google Workspace Enterprise Plus free trial.

The **Admin console** opens.



Create groups for mailing lists and applying policies

Manage web and mobile app access and settings

Manage devices and secure your organization's data

Manage your organization's profile and preferences

Organize users into units for applying policies

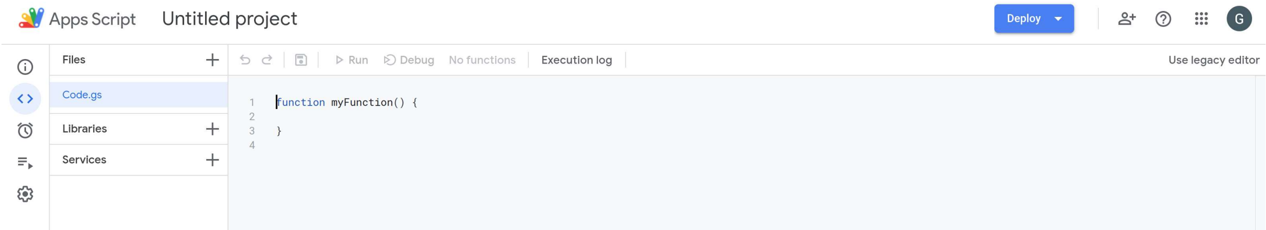
Configure security settings, and view alerts and analytics

If you get the server error "We are unable to process your request at this time, please try again later", the Admin console is still spinning up. Close the browser window and click **Open Google Console** again.

# Create the script project

1. Click this [link to Apps Script]() to open in a new tab or browser window. A new project opens. Apps Script places the project file in your Drive root folder so you can find it later.

**Note**: If the Apps Script project is not appearing then wait for a while and try again.



```
function myFunction() {

}
```

2. Choose Blank Project if presented with a welcome screen. Click on **Use Legacy editor** and check **Missing menus**. Click **Submit**.

3. Close the **Editor Opt-out Survey** and then click **Dismiss**. Replace the `Code.gs` file contents with the following:

### Code.gs

```
/**
 * Copyright Google LLC
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

/**
 * Returns the array of cards that should be rendered for the current
 * e-mail thread. The name of this function is specified in the
 * manifest 'onTriggerFunction' field, indicating that this function
 * runs every time the add-on is started.
 *
 * @param {Object} e The data provided by the Gmail UI.
 * @return {Card[]}
 */
function buildAddOn(e) {
  // Activate temporary Gmail add-on scopes.
  var accessToken = e.messageMetadata.accessToken;
  GmailApp.setCurrentMessageAccessToken(accessToken);

  var messageId = e.messageMetadata.messageId;
  var message = GmailApp.getMessageById(messageId);

  // Get user and thread labels as arrays to enable quick sorting and indexing.
  var threadLabels = message.getThread().getLabels();
  var labels = getLabelArray(GmailApp.getUserLabels());
  var labelsInUse = getLabelArray(threadLabels);
  // Create a section for that contains all user Labels.
  var section = CardService.newCardSection()
    .setHeader("<font color=\"#1257e0\"><b>Available User Labels</b></font>");

  // Create a checkbox group for user labels that are added to prior section.
  var checkboxGroup = CardService.newSelectionInput()
    .setType(CardService.SelectionInputType.CHECK_BOX)
    .setFieldName('labels')
    .setOnChangeAction(CardService.newAction().setFunctionName('toggleLabel'));

  // Add checkbox with name and selected value for each User Label.
  for(var i = 0; i < labels.length; i++) {
    checkboxGroup.addItem(labels[i], labels[i], labelsInUse.indexOf(labels[i])!= -1);
  }

  // Add the checkbox group to the section.
  section.addWidget(checkboxGroup);

  // Build the main card after adding the section.
  var card = CardService.newCardBuilder()
    .setHeader(CardService.newCardHeader()
    .setTitle('Quick Label')

.setImageUrl('https://www.gstatic.com/images/icons/material/system/1x/label_googblue_48
dp.png'))
    .addSection(section)
    .build();

  return [card];
}

/**
 * Updates the labels on the current thread based on
 * user selections. Runs via the OnChangeAction for
 * each CHECK_BOX created.
 *
 * @param {Object} e The data provided by the Gmail UI.
*/
function toggleLabel(e){
  var selected = e.formInputs.labels;

  // Activate temporary Gmail add-on scopes.
  var accessToken = e.messageMetadata.accessToken;
  GmailApp.setCurrentMessageAccessToken(accessToken);

  var messageId = e.messageMetadata.messageId;
  var message = GmailApp.getMessageById(messageId);
  var thread = message.getThread();

  if (selected != null){
     for each (var label in GmailApp.getUserLabels()) {
       if(selected.indexOf(label.getName()) != -1){
          thread.addLabel(label);
       }
       else {
         thread.removeLabel(label);
       }
     }
  }
  else {
    for each (var label in GmailApp.getUserLabels()) {
      thread.removeLabel(label);
    }
  }
}
```

```
/**
 * Converts an GmailLabel object to a array of strings.
 * Used for easy sorting and to determine if a value exists.
 *
 * @param {labelsObjects} A GmailLabel object array.
 * @return {lables[]} An array of labels names as strings.
*/
function getLabelArray(labelsObjects){
  var labels = [];
  for(var i = 0; i < labelsObjects.length; i++) {
    labels[i] = labelsObjects[i].getName();
  }
  labels.sort();
  return labels;
}
```

4. Before saving, we'll need to update the script manifest to avoid errors.

# Update the script manifest

Update the manifest (`appsscript.json`) to provide the add-on with required deployment information.

1. In the script editor, select the **View** > **Show manifest file** menu item. Name your project "`Gmail Add-on Quickstart`", and click **OK**. This opens the manifest file, `appsscript.json` in the editor.

2. Replace the content of the manifest (`appsscript.json`)with the following:

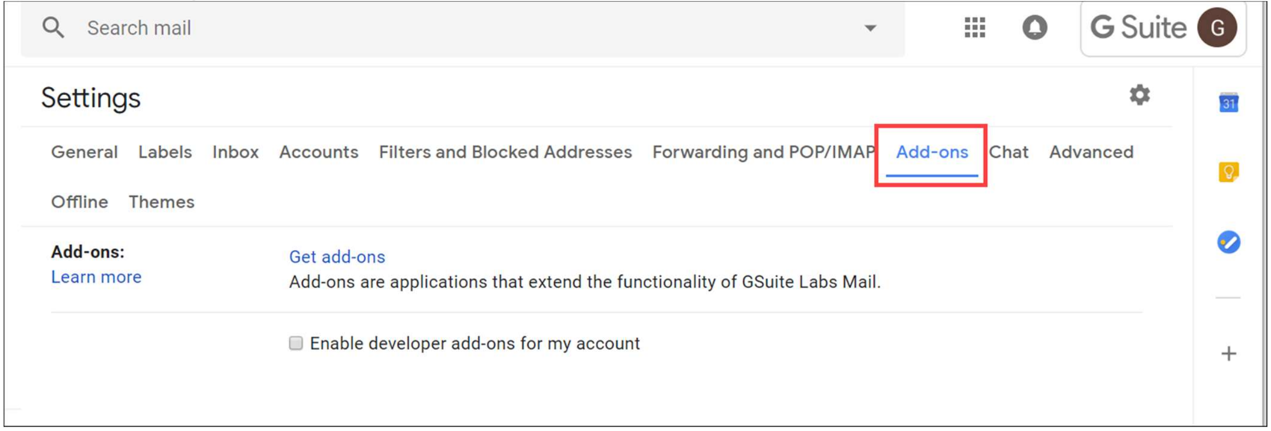
**appsscript.json**

```
{
  "oauthScopes": [
    "https://www.googleapis.com/auth/gmail.addons.execute",
    "https://www.googleapis.com/auth/gmail.addons.current.message.metadata",
    "https://www.googleapis.com/auth/gmail.modify"
  ],
  "gmail": {
    "name": "Gmail Add-on Quickstart - QuickLabels",
    "logoUrl":
"https://www.gstatic.com/images/icons/material/system/1x/label_googblue_24dp.png",
    "contextualTriggers": [{
      "unconditional": {
      },
      "onTriggerFunction": "buildAddOn"
    }],
    "openLinkUrlPrefixes": [
      "https://mail.google.com/"
    ],
    "primaryColor": "#4285F4",
    "secondaryColor": "#4285F4"
  }
}
```

3. Select **File** > **Save all** to save these changes to the manifest and Code.gs. This step provides the add-on with required deployment information.

# Deploy the add-on

1. In the script editor, select **Publish** > **Deploy from manifest...**.
2. In the **Deployments** dialog, click the **Install add-on** link.
3. After receiving the **Deployment installed** message, still in the **Deployments** dialog, click **Close**. Creating a new deployment is not necessary, as you are using the head deployment only.
4. Open Gmail in another tab dismissing any welcome messages that might appear.
5. Open the Gmail add-on settings tab. Click the Gear Wheel ⚙ > **See All Settings** and then select the **Add-ons** tab.

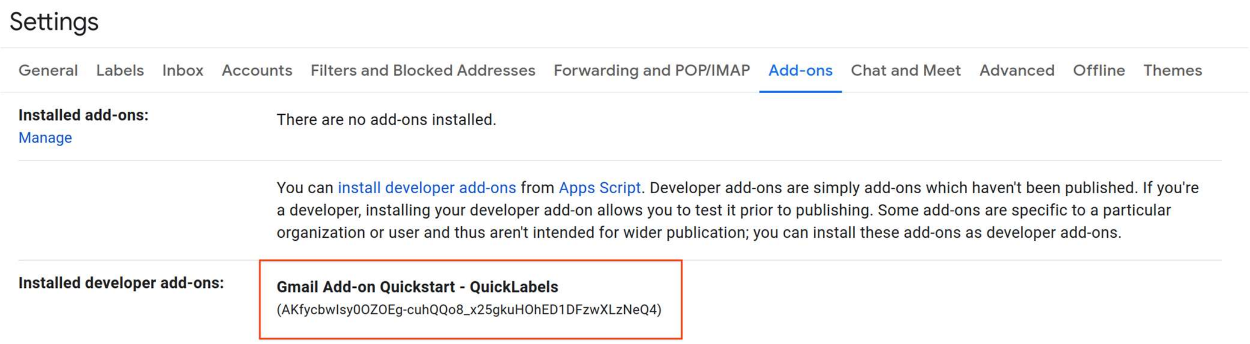


Add-ons:
Learn more

Get add-ons
Add-ons are applications that extend the functionality of GSuite Labs Mail.

□ Enable developer add-ons for my account

When you have completed these steps, the add-on appears in the **Installed developer add-ons** list and becomes available for use in Gmail.



Installed add-ons:
Manage

There are no add-ons installed.

You can install developer add-ons from Apps Script. Developer add-ons are simply add-ons which haven't been published. If you're a developer, installing your developer add-on allows you to test it prior to publishing. Some add-ons are specific to a particular organization or user and thus aren't intended for wider publication; you can install these add-ons as developer add-ons.

Installed developer add-ons:

Gmail Add-on Quickstart - QuickLabels
(AKfycbwIsy0OZOEg-cuhQQo8_x25gkuHOhED1DFzwXLzNeQ4)

Try refreshing the page if you don't see your add-on in installed developer add-ons after following the steps. If it's still not there after refreshing, return to the **Gmail Add-on Quickstart** tab and uninstall and reinstall the add-on from the **Publish** > **Deploy from manifest...** window.

# Run the add-on

1. On a desktop device, open Gmail. You may need to refresh the tab if Gmail is already open.
2. Choose a message in Gmail and open it. If for some reason you don't have any emails in your account, send this account at least 2 emails from another account (or even to yourself through this account).

3. Create some labels

4. Click the **Labels** button ( 🏷 ) from the Gmail menu bar above the email.

5. Click **Create new**.

6. Enter "Test 1" as your label name and click the **Create** button.

7. Repeat steps a-c, but replacing the next label name with "Test 2".

8. Your installed add-ons appear as a column of icons to the right of the open email thread. Click on the 🏷 icon that corresponds to the "Gmail Add-on Quickstart" you just installed.

9. The add-on should place a contextual card on the right-side of the window, with a message asking for authorization. Click the **Authorize access** link to open a dialog where you can authorize the add-on.

10. Select the account that should authorize the add-on.

11. (optional) The next dialog may inform you that the app is not verified. In this case you can proceed by doing the following:

12. Click **Advanced**.

13. At the bottom of the dialog, click **Go to Gmail Add-on Quickstart (unsafe)**.

14. In the new dialog, type "Continue" into the text field, then click **Next**.

15. Read the notice in the next dialog carefully, then click **Allow**.

16. Once authorized, the add-on should automatically refresh and start operating.

# Use the add-on

1. You should see that **Test 1** and the **Test 2** labels are checked under the **Available User Labels** section of the add-on.

2. Click **Test 2** checkbox to deselect the label. This will remove the label from that thread.

3. Verify that the appropriate labels have been applied by navigating back to the inbox and observing that only the **Test 1** label is still applied and **Test 2** has been removed.

4. Open a different email.

5. Select **Test 1** and **Test 2** checkboxes enabling them both.

6. Return to the inbox and observe that both labels have been applied to the email thread.

# Congratulations!

This concludes the hands-on lab, Creating a Gmail Add-on. You now have an idea of what Gmail add-ons are capable of. You could do many other things like displaying other recent threads that the sender has started or translating the text of an email from one language to another without leaving Gmail.

## Finish Your Quest

This self-paced lab is part of the Qwiklabs Quest, Workspace Integrations. A Quest is a series of related labs that form a learning path.
Completing this Quest earns you the badge above, to recognize your achievement. You can make your badge (or badges) public and link to them in your online resume or social media account. Enroll in this Quest and get immediate completion credit if you've taken this lab. See other available Qwiklabs Quests.

## Take your Next Lab

Learn more about Workspace:

- Google Apps Script: Access Google Sheets, Maps & Gmail in 4 Lines of Code
- Hangouts Chat bot - Apps Script

**Next Steps**

- Learn more about Gmail Add-ons
- Google Workspace Learning Center

## Google Cloud Training & Certification

...helps you make the most of Google Cloud technologies. Our classes include technical skills and best practices to help you get up to speed quickly and continue your learning journey. We offer fundamental to advanced level training, with on-demand, live, and virtual options to suit your busy schedule. Certifications help you validate and prove your skill and expertise in Google Cloud technologies.

Manual Last Updated April 1, 2021

Lab Last Tested April 1, 2021