

# Apps Script Questions with Solutions and Coding Examples 2



How can I create a menu in Google Sheets using Google Apps Script?	2
How can I protect a range in a Google Sheet using Google Apps Script?	3
How can I get the current date and time using Google Apps Script?	3
How can I copy a sheet within a Google Sheet using Google Apps Script?	4
How can I get the last row of data in a Google Sheet using Google Apps Script?	4
How can I send an email from a Google Sheet using Google Apps Script?	5
How can I add a new row of data to a Google Sheet using Google Apps Script?	6
How can I get the value of a cell in a Google Sheet using Google Apps Script?	6
How can I delete a row of data from a Google Sheet using Google Apps Script?	7
How can I add a custom function to a Google Sheet using Google Apps Script?	7

## How can I create a menu in Google Sheets using Google Apps Script?

```
function onOpen() {  
  var ui = SpreadsheetApp.getUi();  
  ui.createMenu('Custom Menu')  
    .addItem('Menu Item 1', 'menuItem1')  
    .addItem('Menu Item 2', 'menuItem2')  
    .addToUi();  
}
```

```
function menuItem1() {  
  // Code for menu item 1  
}
```

```
function menuItem2() {  
  // Code for menu item 2  
}
```

Explanation: This code creates a custom menu in the Google Sheet using the `onOpen()` function. The `getUi()` function returns the user interface for the spreadsheet, and the `createMenu()` function creates a new menu with the specified name (Custom Menu). The `addItem()` function adds menu items to the menu, with the first parameter being the name of the menu item and the second parameter being the function to be executed when the menu item is clicked. The `addToUi()` function adds the menu to the user interface.

## How can I protect a range in a Google Sheet using Google Apps Script?

```
function protectRange() {  
  var sheet = SpreadsheetApp.getActiveSheet();  
  var range = sheet.getRange('A1:B10');  
  var protection =  
range.protect().setDescription('Protected Range');  
  protection.removeEditors(protection.getEditors());  
  if (protection.canDomainEdit()) {  
    protection.setDomainEdit(false);  
  }  
}
```

Explanation: This code protects a range of cells (A1:B10) in the active sheet of a Google Sheet. The `getActiveSheet()` function returns the active sheet, and the `getRange()` function returns the range of cells to be protected. The `protect()` function protects the range and returns a protection object, which is stored in the `protection` variable. The `setDescription()` function sets a description for the protection. The `removeEditors()` function removes all editors from the protection, and the `canDomainEdit()` function checks if the domain is allowed to edit the protection. If the domain is allowed to edit the protection, the `setDomainEdit()` function sets the domain to not be able to edit the protection.

## How can I get the current date and time using Google Apps Script?

```
function getCurrentDateTime() {
```

```
var now = new Date();  
Logger.log(now);  
}
```

Explanation: This code gets the current date and time using the Date() constructor and logs it to the execution log using the Logger.log() function.

## How can I copy a sheet within a Google Sheet using Google Apps Script?

```
function copySheet() {  
  var ss = SpreadsheetApp.getActiveSpreadsheet();  
  var sheetToCopy = ss.getSheetByName('Sheet1');  
  var newSheet = sheetToCopy.copyTo(ss).setName('Copy  
of Sheet1');  
}
```

Explanation: This code copies a sheet (Sheet1) in the active Google Sheet to a new sheet with the name Copy of Sheet1. The getActiveSpreadsheet() function returns the active spreadsheet, the getSheetByName() function returns the sheet to be copied, and the copyTo() function copies the sheet to the active spreadsheet and returns the new sheet. The setName() function sets the name of the new sheet.

## How can I get the last row of data in a Google Sheet using Google Apps Script?

```
function getLastRow() {
```

```
var sheet = SpreadsheetApp.getActiveSheet();
var range = sheet.getDataRange();
var lastRow = range.getLastRow();
Logger.log(lastRow);
}
```

Explanation: This code gets the last row of data in the active sheet of a Google Sheet. The `getActiveSheet()` function returns the active sheet, and the `getDataRange()` function returns the range of cells that contain data in the sheet. The `getLastRow()` function returns the last row of the range, which is the last row of data in the sheet. The `Logger.log()` function logs the last row to the execution log.

## How can I send an email from a Google Sheet using Google Apps Script?

```
function sendEmail() {
  var recipient = 'recipient@example.com';
  var subject = 'Email Subject';
  var body = 'Email Body';
  MailApp.sendEmail(recipient, subject, body);
}
```

Explanation: This code sends an email to a recipient with the specified email address (`recipient@example.com`), subject (Email Subject), and body (Email Body) using the `sendEmail()` function of the `MailApp` class.

## How can I add a new row of data to a Google Sheet using Google Apps Script?

```
function addRow() {  
  var sheet = SpreadsheetApp.getActiveSheet();  
  sheet.appendRow(['New Value 1', 'New Value 2']);  
}
```

Explanation: This code adds a new row to the active sheet of a Google Sheet with the values New Value 1 in column A and New Value 2 in column B using the `appendRow()` function of the sheet object.

## How can I get the value of a cell in a Google Sheet using Google Apps Script?

```
function getCellValue() {  
  var sheet = SpreadsheetApp.getActiveSheet();  
  var cell = sheet.getRange('A1');  
  var value = cell.getValue();  
  Logger.log(value);  
}
```

Explanation: This code gets the value of cell A1 in the active sheet of a Google Sheet. The `getActiveSheet()` function returns the active sheet, and the `getRange()` function returns the cell at the specified coordinates. The `getValue()` function returns the value of the cell, which is stored in the value variable. The `Logger.log()` function logs the value to the execution log.

## How can I delete a row of data from a Google Sheet using Google Apps Script?

```
function deleteRow() {  
  var sheet = SpreadsheetApp.getActiveSheet();  
  var rowToDelete = 2;  
  sheet.deleteRow(rowToDelete);  
}
```

Explanation: This code deletes the second row of data from the active sheet of a Google Sheet using the `deleteRow()` function of the sheet object.

## How can I add a custom function to a Google Sheet using Google Apps Script?

```
function myFunction(input) {  
  return input * 2;  
}
```

Explanation: This code defines a custom function named `myFunction` that takes one input parameter and returns the input value multiplied by 2. This function can be used in a Google Sheet by typing `=myFunction(A1)` in a cell, where A1 is the cell containing the input value.