Lock Sheet - Script Lock Options

In Google Apps Script, a script lock is a feature that helps prevent concurrent access to data by different users or processes. Script locks can be used to synchronize access to shared resources, such as a file or a sheet, to ensure that data is not overwritten or corrupted by multiple users or scripts running simultaneously.

There are two types of script locks in Google Apps Script:

Lock Service: The Lock Service provides a way to obtain a lock on a specific resource, such as a file or a sheet, to prevent other users or scripts from modifying it while the lock is held. The lock can be acquired either in shared mode, which allows other users or scripts to read the resource but not modify it, or in exclusive mode, which prevents any other access to the resource while the lock is held.

Properties Service: The Properties Service provides a way to store and retrieve key-value pairs associated with a specific user or script. The properties can be accessed and modified by any script running under the same user account or project, making it a convenient way to share data between different parts of a script or different scripts running in the same environment. The Properties Service also provides a way to obtain a script lock on the properties, which can be used to synchronize access to shared data between different scripts.

Examples of how to use the Lock Service and Properties Service in Google Apps Script:

Lock Service Example

The following example demonstrates how to use the Lock Service to obtain a lock on a specific resource, such as a sheet, to prevent other users or scripts from modifying it while the lock is held.

```
function myFunction() {
  var sheet = SpreadsheetApp.getActiveSheet();
  var lock = LockService.getScriptLock();

  try {
    lock.waitLock(30000); // Wait up to 30 seconds to
acquire lock.
    // Do some work on the sheet here...
  } catch (e) {
    Logger.log('Failed to acquire lock: ' + e);
  } finally {
    lock.releaseLock();
  }
}
```

In this example, the getScriptLock() method is used to obtain a lock on the script. The waitLock() method is then used to wait for up to 30 seconds to acquire the lock. If the lock cannot be acquired within that time period, the script will throw an exception. Once the lock is acquired, the script can perform some work on the sheet. Finally, the releaseLock() method is called to release the lock.

Properties Service Example

The following example demonstrates how to use the Properties Service to store and retrieve key-value pairs associated with a specific user or script.

```
function myFunction() {
  var key = 'myKey';
  var value = 'myValue';
  var properties =
PropertiesService.getUserProperties();

  properties.setProperty(key, value);

  // Retrieve the value of the property.
  var retrievedValue = properties.getProperty(key);
  Logger.log(retrievedValue); // Output: myValue
}
```

In this example, the getUserProperties() method is used to obtain a reference to the user properties associated with the script. The setProperty() method is then used to store a key-value pair in the properties. The getProperty() method is then used to retrieve the value of the property with the specified key. The value is then logged to the console using the Logger.log() method.