

Da Google Doc con "GGeditor" → via GitHub → a Read-the-Docs

φ HTML

```

```

φ HTML

```
<span style="background-color: #6462d1; color: #ffffff; display: inline-block; padding: 0px 5px; border-radius: 5px;"><small>Tutorial</small></span>


```

Le spiegazioni contenute in questo tutorial rappresentano una via per migliorare la qualità dei documenti pubblicati sul web e sono la traduzione in italiano del [tutorial GGeditor](#). Il tutorial spiega come usare lo strumento di **Google doc**, ed un componente aggiuntivo, per pubblicare documenti con lo stile di **Read the Docs**, utilizzando anche **GitHub**.

Questo documento nasce subito dopo la produzione del tutorial "[Come abbiamo creato un «Read the Docs» per pubblicare documenti pubblici su Docs Italia](#)".

① Per questo tutorial i seguenti ringraziamenti particolari

Ringraziare queste persone è importante, perché con i loro approfondimenti hanno permesso di ampliare le possibilità d'uso di ``Read the Docs`` come piattaforma di pubblicazione documentale:

[Hsin Yuan Yeh](#), [Andrea Borruso](#), [Giovan Battista Vitrano](#), [Salvatore Fiandaca](#), [Pablo Persico](#), [Marina Bassi](#), [Andrea Ivan Baldassarre](#), [Daniele Rizzo](#), [Maurizio Costa](#), [Michela Stentella](#).

φ HTML

```

```

|

Vantaggi dell'uso di "Read the Docs"

L'uso di "Read the Docs" come piattaforma di pubblicazione di documenti online ha i seguenti vantaggi rispetto al formato PDF:

HTML

```
<p><span style="background-color: #6462d1; color: #ffffff; display: inline-block; padding: 3px 8px; border-radius: 10px;">Responsive</span> </p>
```

```
<p><span style="background-color: #105618; color: #ffffff; display: inline-block; padding: 3px 8px; border-radius: 10px;">Funzioni avanzate di ricerca testo</span> </p>
```

```
<p><span style="background-color: #14c9ab; color: #ffffff; display: inline-block; padding: 3px 8px; border-radius: 10px;">Fornisce testo in HTML, EPUB e PDF</span> </p>
```

```
<p><span style="background-color: #e86514; color: #ffffff; display: inline-block; padding: 3px 8px; border-radius: 10px;">Codice sorgente del testo online</span> </p>
```

```
<p><span style="background-color: #c914c0; color: #ffffff; display: inline-block; padding: 3px 8px; border-radius: 10px;">E' elegante e bello da vedere</span> </p>
```

Per i nostalgici e dipendenti da documenti in formato PDF (non accessibili comodamente da dispositivi mobili), la documentazione esposta su "Read the Docs" permette di scaricare il contenuto dell'intero documento pubblicato online sia in formato PDF che EPUB oltre che HTML.

HTML

```

```

Changelog

Registro di tutti gli aggiornamenti | modifiche principali apportate al progetto

Aggiornamento N. 1 - gennaio_2020

Su Google doc cliccando su "installa componente aggiuntivo" non trovate più "GGeditor" neanche sul Marketplace di Google (capita dall'agosto 2019 per una ridefinizione dei termini d'uso di Google). Passate alla procedura del successivo aggiornamento.

In alternativa all'installazione del componente aggiuntivo su Google doc (dall'elenco dei componenti aggiuntivi forniti da Google), le funzioni svolte da GGeditor possono anche essere assicurate creando un Google doc che contiene uno script con il codice sorgente del componente **GGeditor**. Una volta creato lo script (con la procedura di seguito illustrata) avviando il comando "Commit to Github" (percorso: componenti aggiuntivi / GGeditor / Commit to Github), è possibile creare automaticamente un file in formato ``.RST`` su Github partendo dal contenuto editato su Google doc.



I file contenuti nello script da creare su Google doc

Lo script è costituito dai seguenti files che si trovano dentro il repository https://github.com/cirospat/GGeditor_script:

- conversion.html
- explicitMarkup.html
- generator.gs
- github.html
- properties.gs
- reSTMetadata.gs
- settings.html
- sidebar.html
- 程式碼.gs (程式碼 in cinese significa :guilabel:`codice`)

Per creare lo script su Google doc, andare su **menu strumenti / < > editor di script**. Nella pagina dello script copiare il codice dei 9 file di cui al repository https://github.com/cirospat/GGeditor_script dando lo stesso nome dei 9 file di cui sopra. Allo script così creato date il nome ``GGeditor``.

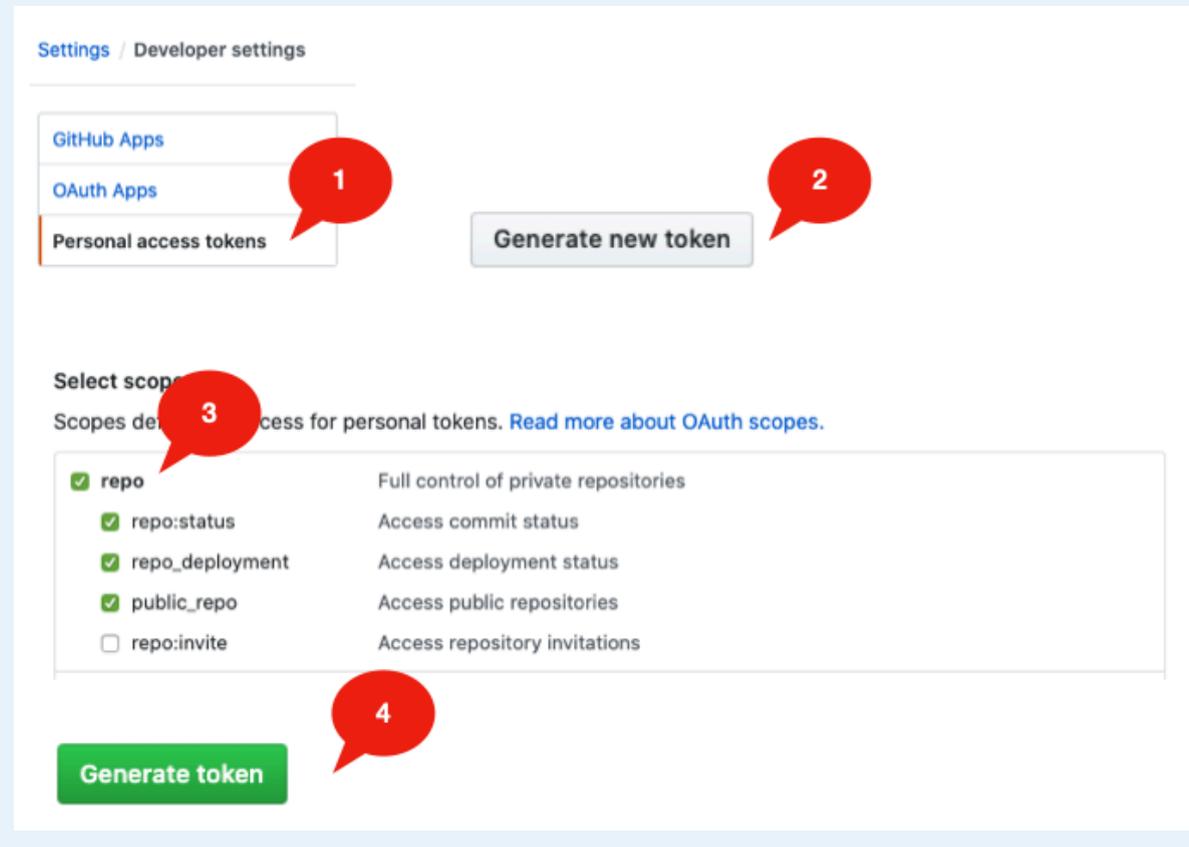
Messaggio "**Bad Credential**" ("Credenziali errate di Github") su GGeditor. Github ha deprecato la sua API di autenticazione per "*nome utente*" e "*password*", che è la causa principale del problema del messaggio "Bad Credential" ("Credenziali non valide").

[La soluzione è facile](#), basta sostituire la password con cui si entra nell'account Github con il "**token di accesso personale**" quando si esegue il commit in GGEditor nel Google doc. I passi da seguire sono i seguenti:

1. Vai alla [pagina delle impostazioni in Github.com](#) e **crea un token di accesso personale** in Github.com ([How to by Github](#)). Quindi copia il token di accesso personale negli appunti.

2. Apri un documento Google e rimuovi tutte le credenziali archiviate precedentemente in GGEDitor, quindi aggiungi un nuovo account Github con il token copiato come password.

Guarda i passi da compiere nelle schermate di Github:



① Aggiornamento N. 4 - ottobre_2021

Messaggio di “non compilazione” del progetto su readthedocs.org e soluzione (news di fine ottobre 2021). Guarda anche la relativa [issue su Github](#).

Da fine ottobre 2021 su readthedocs.org compare una non compilazione del progetto (**build failed**) legata alla versione di ``Sphinx``. La soluzione è inserire nel file ``requirements.txt`` la seguente sintassi:

```
``docutils<0.18``
```

come di seguito illustrato.

- 1- avere un [file](#) denominato ``requirements.txt`` con il seguente codice:

```
.. code-block:: python
```

```
:linenos:
```

```
sphinx-rtd-theme  
sphinx  
recommonmark  
markdown  
sphinx-markdown-tables  
docutils<0.18
```

2- avere un file ``.readthedocs.yaml`` con il seguente codice:

```
.. code-block:: python
```

```
:linenos:
```

```
version: 2  
python:  
  install:  
    - requirements: requirements.txt
```

① Aggiornamento N. 5 - maggio_2022

Fortunatamente il :guilabel:`componente aggiuntivo` per Google Doc ``GG editor`` è di nuovo disponibile nello store di Google per l'installazione. Vai al seguente link: <https://workspace.google.com/u/1/marketplace/app/ggeditor/644886139871>

① Messaggio di errore "undefined", soluzione

Nelle azioni di commit sul componente GGeditor potrebbe comparire un messaggio di "undefined" ed un messaggio di error in lingua taiwanese.

Ho aperto una issue (<https://github.com/iapyeh/GGeditor/issues/1>) sul suo progetto Github e Hsin Yuan Yeh (autore di GGeditor) ha provveduto ad analizzare tempestivamente il problema ed ha fornito immediatamente una soluzione che è stata implementata nel codice sorgente di **GGeditor**.

Praticamente se dovesse capitare di leggere un messaggio "undefined" o un messaggio in lingua taiwanese del tipo

↓

無法讀取 null 的
「19rZn4reioTxYQL2dLiVbxOHrMnw2llq_5r6xtLpPqVQ」屬
性。

OK

La cosa da fare è:

andare in “**Componenti aggiuntivi**”, poi “**GGeditor**” e quindi su “**Setting**” dove si trova un tasto rosso di “**Reset**” che cancella tutte le informazioni di collegamento agli account Github e ai relativi file nel repository.

Settings

Accounts

Document

About

The "Reset" button will clean up all your stored information of accounts and bindings.

RESET

Tutto ritorna in ordine e funziona correttamente nella procedura di Commit da Google Doc a Github. Ovviamente bisogna ricollegare il componente aggiuntivo GGeditor all'account di Github per poter continuare a effettuare i commits. Grazie Hsin Yuan Yeh per questa tempestiva soluzione al problema.

L'utilità di GGeditor per i progetti di documentazione online

Il componente aggiuntivo **GGeditor** rappresenta uno strumento molto utile e comodo in quanto i servizi di Google Drive oggi sono molto usati anche nelle Pubbliche Amministrazioni, oltre che dai privati, per la facilità d'uso e per la funzionalità di condivisione dei documenti in gruppo.

Il lavoro principale che svolge il componente aggiuntivo **GGeditor** è quello di trasformare il testo editato su un foglio di Google doc in un file con linguaggio ``.RST`` dentro il repository

di **Github**. Github a sua volta permette la compilazione automatica dello stesso documento su **Read the Docs** in pagine ``HTML`` . Sembra una cosa difficile nella descrizione, ma se lo faccio [io](#) lo possono fare tutti, con un pizzico di pazienza e curiosità.

φ HTML

```
<iframe width="100%" height="500" src="https://www.youtube.com/embed/502D4h5hI18" frameborder="0" allow="autoplay; encrypted-media" allowfullscreen></iframe>  
Breve video introduttivo (2'10")
```

|

Le principali funzioni e punti di forza di GGeditor

- Facile inizio per chi non ha dimestichezza con i file ``RST``, anche per chi non ha idea dei marcatori di ``RST``.
- Alimentato da Google Docs. Quasi la totalità di quello che vedi su Google Docs è quello che ottieni su Read the Docs. Lo stesso è per l'intero gruppo di lavoro.
- Un click per commissionare il lavoro sul repository di Github.
- Puoi vedere in anteprima il file ``RST`` generato dall'interno di Google Docs e scaricarlo nel tuo PC.
- Supporta headings, bold, italic, hyperlink, subscript e superscript.
- Supporta note a margine, immagini, liste di articolo e tabelle.
- Supporta caratteri a larghezza intera (CKJ) nelle intestazioni e nelle tabelle.
- Supporta i link interni ai bookmarks, headings e le Google Docs tabelle native di contenuti (in document table of contents).
- Supporta i link relativi ai file ``RST`` generati dai Google Docs all'interno della stessa directory e sotto-directory Google Docs.
- Supporta la tabella dei contenuti (cross-document table of content ``(.. toctree::)``) per fare generare l'indice a Read The Docs.
- Supporta tutti gli stili di "admonitions" di Read The Docs.
- Supporta account multipli per compilare i file nei repository di diversi account Github.
- Supporta la conversione di tabelle con i tags ``HTML`` to let look-and-feel e la stessa cosa è possibile per i blogger.

```
.. toctree::  
:maxdepth: 2
```

```
Home <https://googledocs.readthedocs.io>
```

```
.. toctree::  
:maxdepth: 3  
:caption: Indice
```

```
gdocs-rtd
tutorial
come-usarlo
tabelle
inserire_immagini_video
lavoro-github
lavoro-rtd
user-guide
hypothesis-partecipazione
sintassi-rst
sintassi-md
pubblicare-su-docs-italia
licenza
opendatasicilia
```

① Tip

⇒ il contenuto di questa pagina che stai leggendo è editato in [questo Google Doc](#) 📄 ... dai un'occhiata per capire meglio come il testo di Google doc viene esposto su pagine ``HTML`` di `:guilabel:Read the Docs`

φ HTML

```
<script id="dsq-count-scr" src="//guida-readthedocs.disqus.com/count.js" async></script>

<div id="disqus_thread"></div>
<script>

/**
 * RECOMMENDED CONFIGURATION VARIABLES: EDIT AND UNCOMMENT THE SECTION BELOW TO INSERT
 * DYNAMIC VALUES FROM YOUR PLATFORM OR CMS.
 * LEARN WHY DEFINING THESE VARIABLES IS IMPORTANT:
 * https://disqus.com/admin/universalcode/#configuration-variables*/
 */

var disqus_config = function () {
  this.page.url = PAGE_URL; // Replace PAGE_URL with your page's canonical URL variable
  this.page.identifier = PAGE_IDENTIFIER; // Replace PAGE_IDENTIFIER with your page's
  unique identifier variable
};
```

```
*/  
(function() { // DON'T EDIT BELOW THIS LINE  
var d = document, s = d.createElement('script');  
s.src = 'https://guida-readthedocs.disqus.com/embed.js';  
s.setAttribute('data-timestamp', +new Date());  
(d.head || d.body).appendChild(s);  
})();  
</script>  
<noscript>Please enable JavaScript to view the <a  
href="https://disqus.com/?ref_noscript">comments powered by Disqus.</a></noscript>
```