

# Utilizzo delle API per il trasferimento dati da Google Sheet

Edoardo Matterazzo

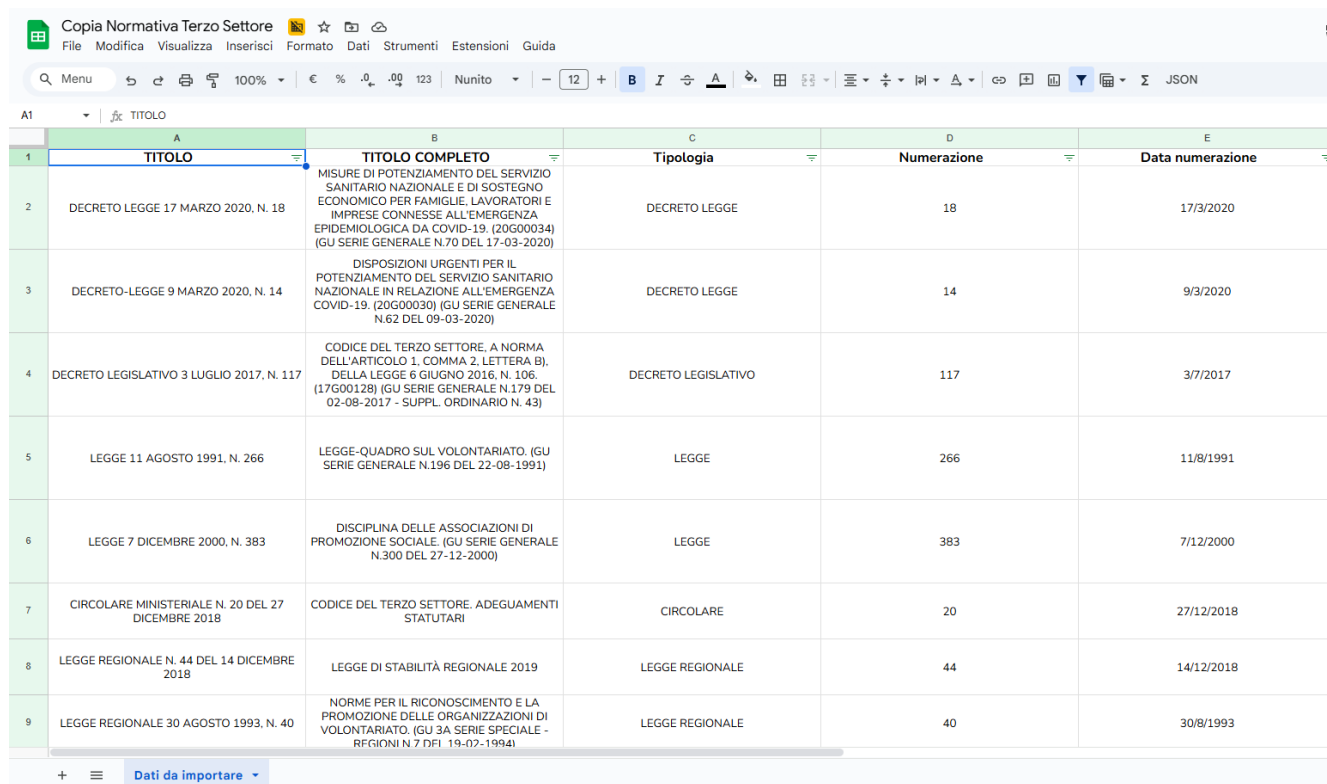
## 1 Inizializzazione

Prima di procedere, è fondamentale verificare che il file da cui si intendono prelevare i dati sia in formato **Foglio Google** e non nel formato **xlsx**. Qualora il file fosse in formato **Excel**, è necessario convertirlo selezionando **File** → **Salva come foglio Google**. In questo modo il documento sarà compatibile con le funzionalità di automazione e condivisione messe a disposizione da **Google Sheets**.

All'interno dei fogli di lavoro devono essere considerate due pagine distinte, ognuna con un ruolo specifico:

1. **Pagina di origine** (Figura 1): contiene i dati da condividere. È importante che la prima riga sia riservata ai titoli delle colonne, così da mantenere una struttura chiara e facilmente interpretabile dallo script.
2. **Pagina di destinazione** (Figura 2): riceverà in automatico i dati aggiornati dalla pagina di origine, garantendo una sincronizzazione costante e senza interventi manuali.

Questa organizzazione consente di separare in modo netto i dati di input da quelli di output, migliorando sia la leggibilità del file che la gestione dei futuri aggiornamenti.



1	A	B	C	D	E
	TITOLO	TITOLO COMPLETO	Tipologia	Numerazione	Data numerazione
2	DECRETO LEGGE 17 MARZO 2020, N. 18	MISURE DI POTENZIAMENTO DEL SERVIZIO SANITARIO NAZIONALE E DI SOSTEGNO ECONOMICO PER FAMIGLIE, LAVORATORI E IMPRESE CONNESSE ALL'EMERGENZA EPIDEMIOLOGICA DA COVID-19. (20G00034) (GU SERIE GENERALE N.70 DEL 17-03-2020)	DECRETO LEGGE	18	17/3/2020
3	DECRETO-LEGGE 9 MARZO 2020, N. 14	DISPOSIZIONI URGENTI PER IL POTENZIAMENTO DEL SERVIZIO SANITARIO NAZIONALE IN RELAZIONE ALL'EMERGENZA COVID-19. (20G00030) (GU SERIE GENERALE N.62 DEL 09-03-2020)	DECRETO LEGGE	14	9/3/2020
4	DECRETO LEGISLATIVO 3 LUGLIO 2017, N. 117	CODICE DEL TERZO SETTORE, A NORMA DELL'ARTICOLO 1, COMMA 2, LETTERA B), DELLA LEGGE 6 GIUGNO 2016, N. 106. (17G00128) (GU SERIE GENERALE N.179 DEL 02-08-2017 - SUPPL. ORDINARIO N. 43)	DECRETO LEGISLATIVO	117	3/7/2017
5	LEGGE 11 AGOSTO 1991, N. 266	LEGGE-QUADRO SUL VOLONTARIATO. (GU SERIE GENERALE N.196 DEL 22-08-1991)	LEGGE	266	11/8/1991
6	LEGGE 7 DICEMBRE 2000, N. 383	DISCIPLINA DELLE ASSOCIAZIONI DI PROMOZIONE SOCIALE. (GU SERIE GENERALE N.300 DEL 27-12-2000)	LEGGE	383	7/12/2000
7	CIRCOLARE MINISTERIALE N. 20 DEL 27 DICEMBRE 2018	CODICE DEL TERZO SETTORE. ADEGUAMENTI STATUTARI	CIRCOLARE	20	27/12/2018
8	LEGGE REGIONALE N. 44 DEL 14 DICEMBRE 2018	LEGGE DI STABILITÀ REGIONALE 2019	LEGGE REGIONALE	44	14/12/2018
9	LEGGE REGIONALE 30 AGOSTO 1993, N. 40	NORME PER IL RICONOSCIMENTO E LA PROMOZIONE DELLE ORGANIZZAZIONI DI VOLONTARIATO. (GU 3A SERIE SPECIALE - RFGIONI N.7 DFI 19-02-1994)	LEGGE REGIONALE	40	30/8/1993

Figura 1: Pagina contenente i dati da prelevare.

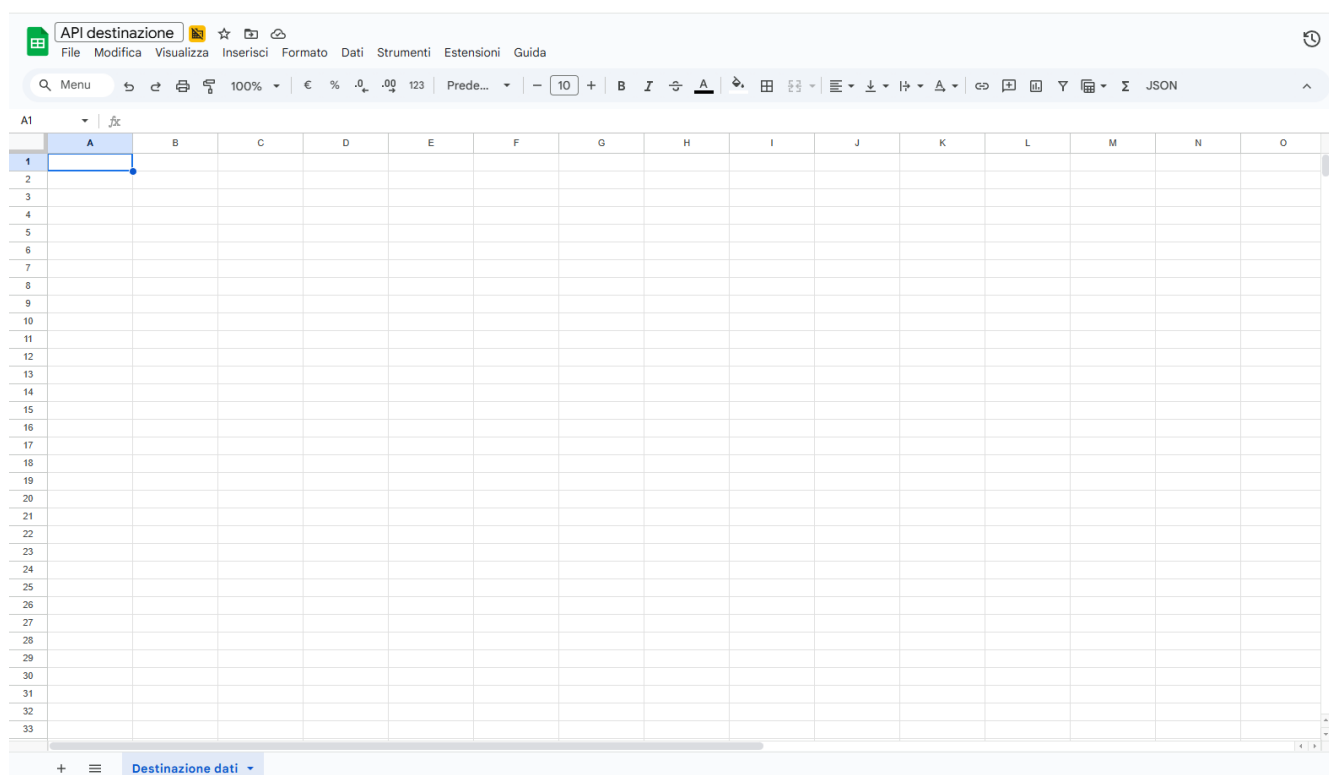


Figura 2: Pagina in cui verranno trasferiti i dati.

## 2 Creazione il link per accedere ai dati

Per trasferire i dati presenti in una pagina di **Google Sheets** sotto forma di tabella, è necessario creare un link che ne descriva la struttura. Dal menu del foglio in cui si vogliono inserire o prelevare i dati, selezionare **Estensioni** → **Apps Script** (Figura 3a). Si aprirà così la schermata mostrata in Figura 3b, contenente del codice di esempio. Questo codice deve essere sostituito con lo script JavaScript disponibile al seguente link: [Scarica \\_Json\\_Script](#). Lo script può essere incollato al posto del codice predefinito ed è possibile rinominare sia il titolo del file (ad esempio: "1) Salva\_Json") sia il progetto che lo contiene (ad esempio: "API"). In Figura 4a è riportata una parte dello script. Gli elementi evidenziati in rosso indicano le variabili personalizzabili dall'utente:

- **USERNAME**: nome utente per l'accesso.
- **PASSWORD**: password per l'accesso.
- **SPREADSHEET\_ID**: identificativo univoco del file **Google Sheets**.
- **SHEET\_NAME**: nome del foglio **Google Sheet** da cui estrarre i dati.

**SPREADSHEET\_ID** sarà fondamentale per identificare il file di lavoro in modo univoco e dovrà essere modificato qualora si volessero prendere i dati da un nuovo file. Questo permette di evitare complicazioni nel caso in cui all'interno di **Google Drive** siano presenti più file con lo stesso nome.

Per fare questo, bisognerà accedere all'URL del foglio, simile a:

```
https://spreadsheets/d/1hZ8Zh-t61feoTtyi5xZvp2yX75qHuN4CGK5zMmYqAqI/edit..
```

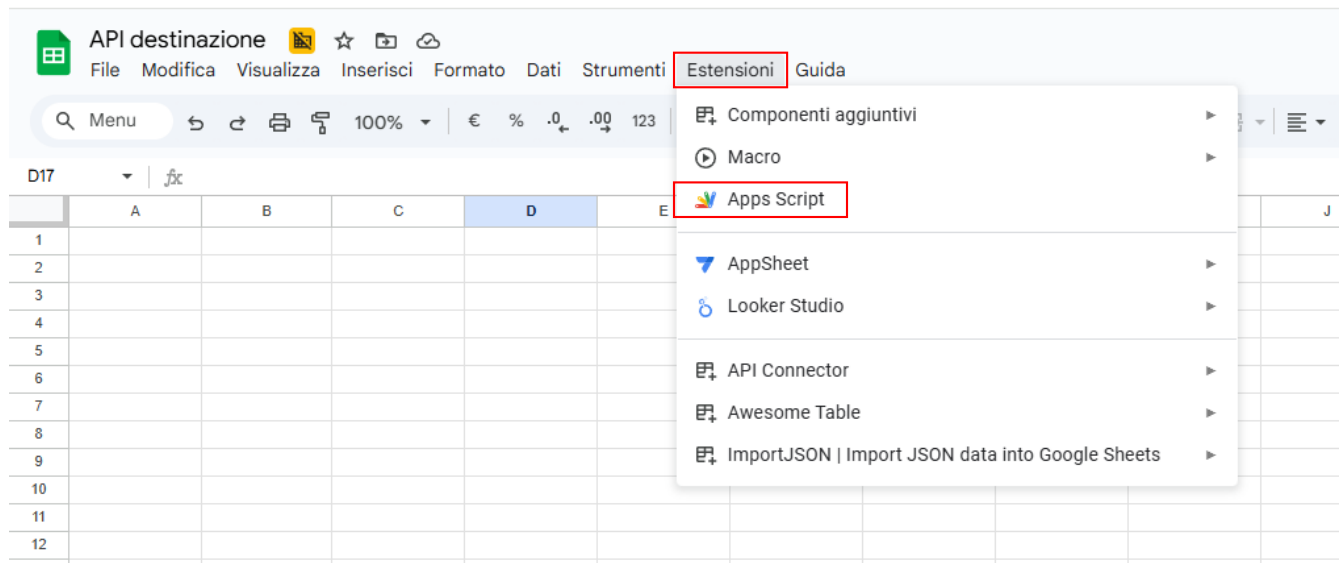
e prendere la parte tra `"/d/"` e `"/edit"`.

Una volta configurato lo script, occorre procedere con il **Deployment**, seguendo la procedura illustrata in Figura 4. In questa fase è necessario specificare:

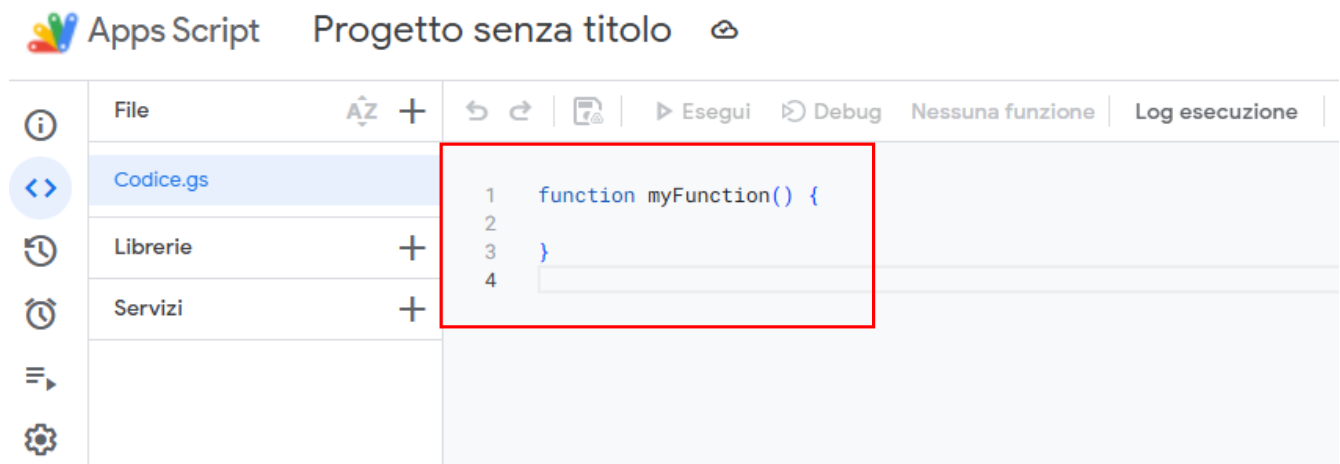
- il nome del Deployment;

- chi può accedere (impostare su "Chiunque")

Al termine, viene generato un endpoint API pubblicato come **Web App**, accessibile tramite un link dedicato. Infine, accedendo a **Gestione Deployment** e selezionando quello più recente (o quello desiderato), sarà possibile recuperare in corrispondenza della voce "Applicazione web" il collegamento diretto all'endpoint.



(a) Sezione **Estensioni** della barra dei menù.



(b) Pagina **Apps Script** all'apertura (default).

Figura 3: Creazione del link contenente la struttura (in formato JSON) della tabella da trasferire.

### 3 Trasferimento dei dati

All'interno della pagina **Apps Script** è necessario creare un nuovo script e assegnargli un nome significativo (ad esempio "Importa.Dati"). Una volta fatto ciò, il codice predefinito generato in automatico deve essere sostituito con quello disponibile al seguente link: [Importa.Dati.Script](#). Dopo aver incollato lo script, l'editor apparirà come in Figura 5. Questo script svolge tre operazioni principali:

1. **Connessione al Web App:** richiama l'endpoint API generato in precedenza, aggiungendo al termine dell'URL le credenziali di accesso (in formato: "?username=...&password=...") ;

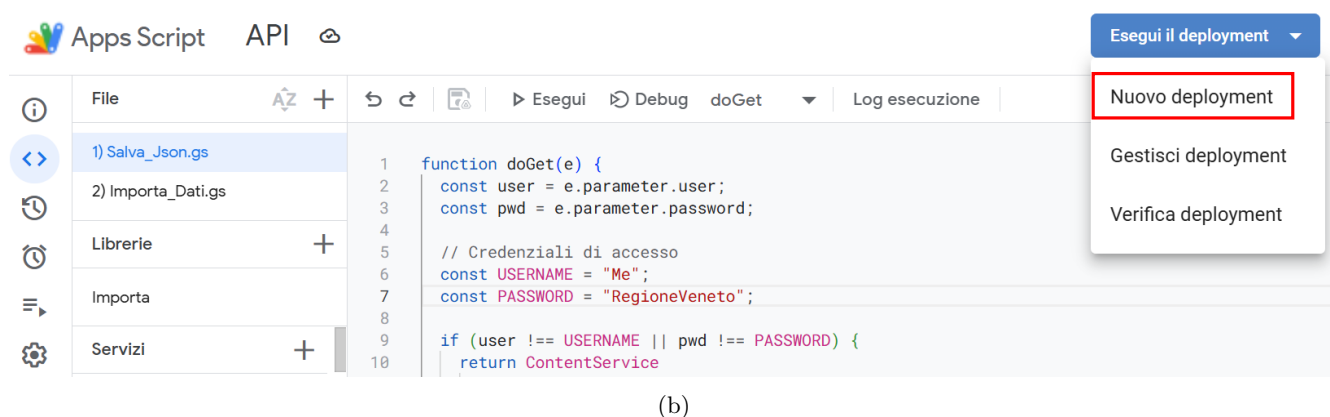
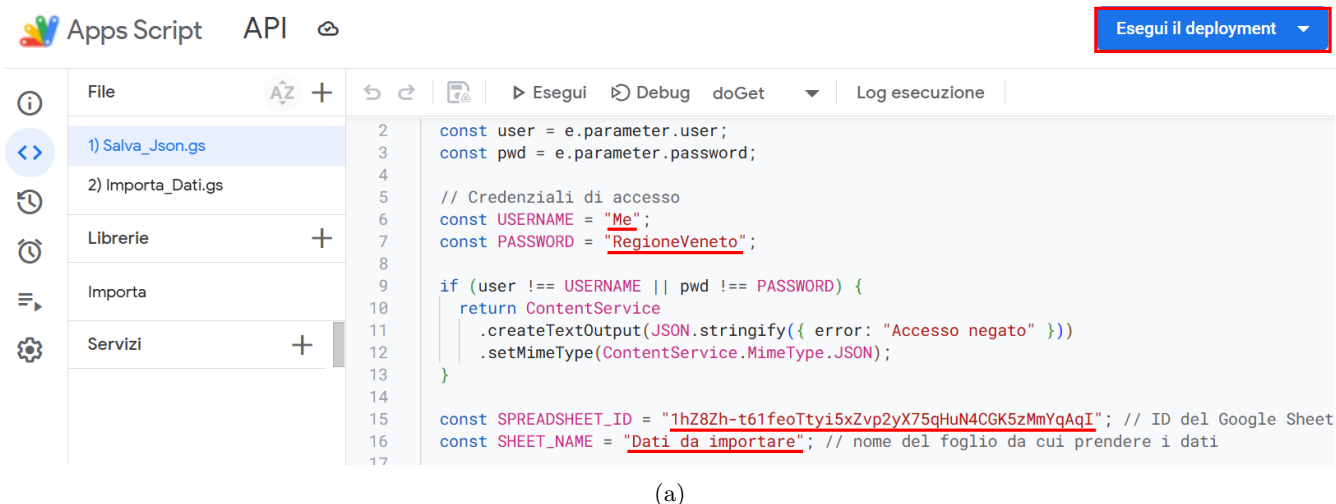


Figura 4: Creazione del link contenente la struttura (in formato JSON) della tabella da trasferire.

2. **Recupero dei dati:** riceve dal Web App i dati in formato JSON, ovvero una rappresentazione strutturata della tabella di partenza;
3. **Aggiornamento del foglio di destinazione:** svuota la pagina di Google Sheets selezionata e vi inserisce i nuovi dati, scrivendo per primi i titoli delle colonne e successivamente tutte le righe della tabella.

L'utente ha la possibilità di personalizzare alcuni elementi cardine dello script, evidenziati in rosso in Figura 5:

- il nome del foglio di destinazione, dove i dati verranno inseriti;
- le credenziali di accesso (username e password), necessarie per autenticare la chiamata al Web App.

Questa flessibilità consente di adattare facilmente lo script a contesti diversi, come l'importazione dei dati in un foglio specifico o la gestione di credenziali differenziate per più utenti. Inoltre, la separazione tra foglio di origine (che genera i dati) e foglio di destinazione (che li riceve) garantisce maggiore chiarezza nella gestione e riduce il rischio di errori dovuti a modifiche involontarie.

## 4 Caricamento giornaliero dei dati in modo automatico

Apps Script permette di eseguire uno script in modo completamente automatico, definendo giorno, data e ora in cui l'operazione deve avvenire. Per configurare un'attivazione temporizzata, procedere come segue:

1. Aprire il foglio Google desiderato, andare su **Estensioni** → **Apps Script** e verificare che sia presente la funzione di importazione (ad esempio `importaJSON`);

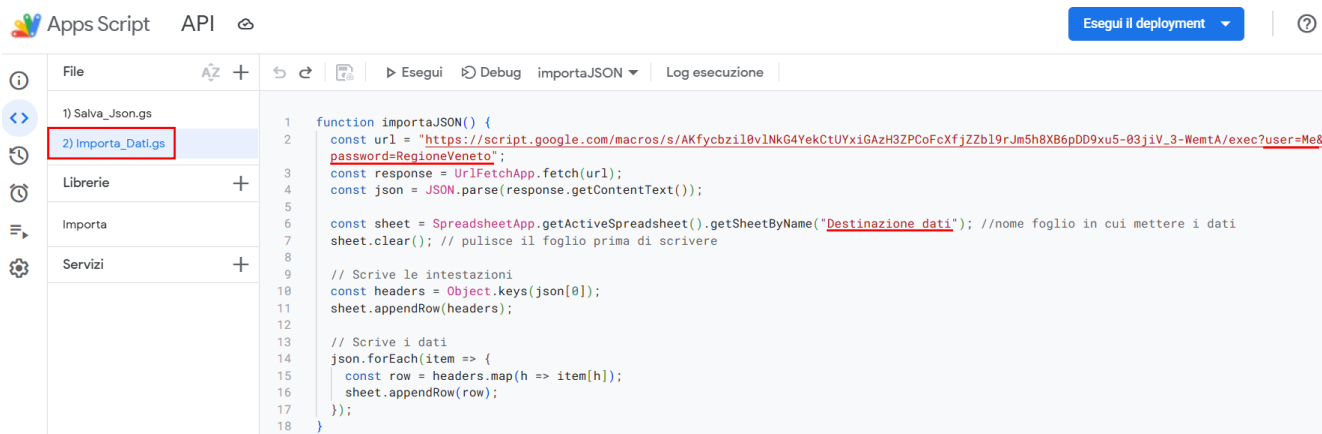


Figura 5: Script per caricare i dati sul nuovo foglio Google Sheets.

2. Nell'editor dello script, cliccare sull'icona a forma di orologio nella barra laterale sinistra per aprire la gestione degli attivatori (Figura 6);
3. Premere **Aggiungi attivatore** per crearne uno nuovo (Figura 7).

Si aprirà la finestra mostrata in Figura 8, dove sarà necessario configurare:

1. **Funzione da eseguire:** selezionare `importaJSON`;
2. **Deployment da eseguire:** scegliere `Head`;
3. **Origine dell'evento:** impostare su `evento vincolato a specifiche temporali`;
4. **Tipo di attivatore orario:** selezionare `Timer giornaliero`;
5. **Ora di esecuzione:** indicare l'orario in cui lo script dovrà essere eseguito ogni giorno.

Una volta completata la configurazione, lo script verrà eseguito automaticamente all'ora impostata, aggiornando quotidianamente la pagina di destinazione con i dati più recenti senza alcun intervento manuale.

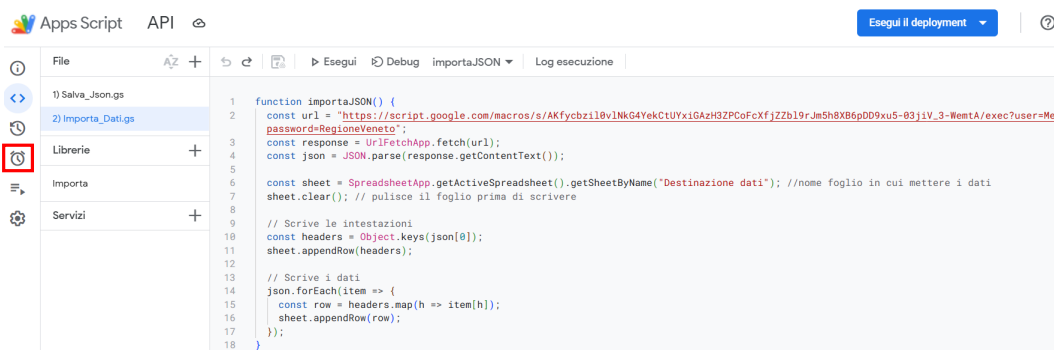


Figura 6: Accesso alla sezione attivatori dalla pagina Apps Script.

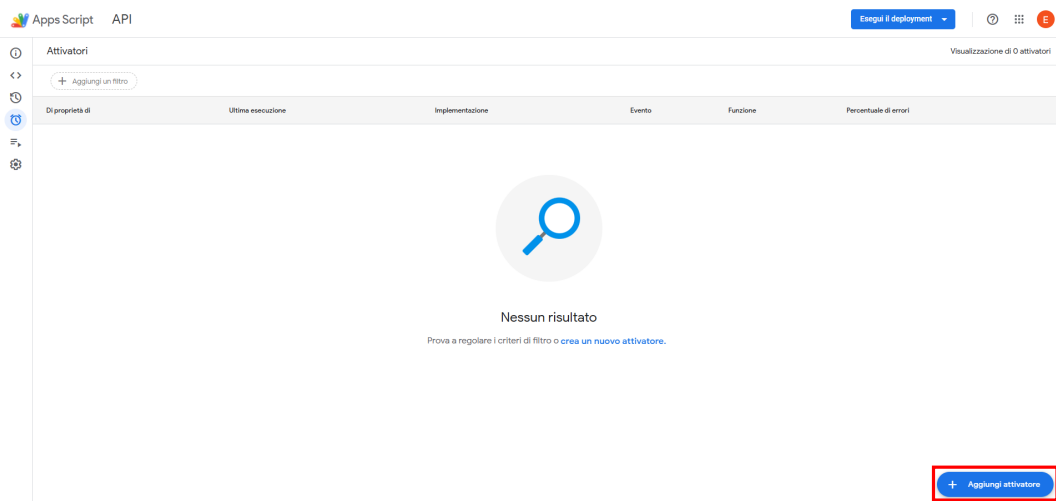


Figura 7: Aggiunta di un nuovo attivatore.

### Aggiungi Attivatore per API

Scegli quale funzione eseguire

doGet

Scegli quale deployment eseguire

Head

Seleziona l'origine dell'evento

Da foglio di lavoro

Seleziona il tipo di evento

All'apertura

Impostazioni di notifica di errore

Invia una notifica ogni giorno

Annulla

Salva

Figura 8: Impostazioni dell'attivatore.