

GABRIELE ROMANATO

Menu

Bash: formattare la risposta JSON delle API REST

Nel mondo moderno dello sviluppo software, l'interazione con API REST è una necessità comune. Le API REST permettono ai diversi sistemi di comunicare tra loro in modo standardizzato. In questo articolo, vedremo come implementare uno script Bash che effettui chiamate verso API REST e restituisca l'output JSON formattato.

Prima di iniziare, assicurati di avere i seguenti strumenti installati nel tuo sistema:

1. **curl**: un tool da riga di comando per trasferire dati con URL.
2. **jq**: un processore di JSON da riga di comando che consente di analizzare, filtrare e trasformare dati JSON.

Per installare curl e jq su un sistema basato su Debian, puoi usare i seguenti comandi:

```
sudo apt-get update
sudo apt-get install curl jq
```

Per iniziare, definiamo le variabili necessarie per la nostra chiamata API. Ad esempio, l'URL dell'API e l'endpoint specifico che vogliamo chiamare:

```
API_URL="https://api.example.com/v1"
ENDPOINT="/data"
```

Usiamo `curl` per effettuare la chiamata API. Supponiamo che l'API richieda un token di autenticazione, che possiamo passare come header:

```
response=$(curl -s -H "Authorization: Bearer  
YOUR_ACCESS_TOKEN" "${API_URL}${ENDPOINT}")
```

Il flag `-s` rende `curl` silenzioso (suppress progress meter), il che rende l'output più pulito.

Usiamo `jq` per formattare l'output JSON. Il nostro obiettivo è renderlo leggibile:

```
formatted_response=$(echo "$response" | jq '.')
```

Infine, stampiamo l'output formattato:

```
echo "$formatted_response"
```

Mettiamo insieme tutti i passaggi:

```
#!/bin/bash  
  
# Definizione delle variabili  
API_URL="https://api.example.com/v1"  
ENDPOINT="/data"  
TOKEN="YOUR_ACCESS_TOKEN"  
  
# Esecuzione della chiamata API  
response=$(curl -s -H "Authorization: Bearer $TOKEN"  
"${API_URL}${ENDPOINT}")
```

```
# Verifica se la chiamata ha avuto successo
if [ $? -eq 0 ]; then
    # Formattazione dell'output JSON
    formatted_response=$(echo "$response" | jq '.')
    # Visualizzazione dell'output formattato
    echo "$formatted_response"
else
    echo "Errore nella chiamata API" >&2
fi
```

Conclusione

Abbiamo creato uno script Bash semplice ma efficace per effettuare chiamate verso API REST e restituire l'output JSON formattato. Questo script può essere facilmente esteso per includere ulteriori funzionalità come la gestione degli errori, il supporto per diversi metodi HTTP (GET, POST, PUT, DELETE), e l'analisi più complessa dei dati JSON. Con `curl` e `jq`, hai potenti strumenti a disposizione per interagire con le API direttamente dalla riga di comando.