



Calendar Skill

בדקו והוסיפו אירועים ליום גוגל באמצעות Apps Script

זמן התקנה: 10-15 דקות

מה הסkill הזה עושה?

הskill מאפשר לך לקרוא ולנהל את יום גוגל שלכם. תוכלן לשאול "מה יש לי היום?" או "הוסף פגישהמחר ב-10" וקלוד יטפל בה!



אירועים הקרובים

מה יש בשעות הקרובות



תצוגה שבועית

כל האירועים של השבוע



אירועי היום

צפיה בכל האירועים של היום



הוספת אירועים

יצירת אירועים חדשים

1 מה צריך לפני שמתחלים?

1

חשבון גугл

צריך חשבון גוגל עם יומן פעיל. זה בחיננו!

Node.js מותקן

בדקו שיש לכם Node.js מותקן:

```
node --version
```

טיפ: הסקיל משתמש ב- **clasp** - הכלי הרשמי של גוגל לניהול Apps Script מהטרמינל.



2 התקנת Clasp

2

Clasp הוא CLI של גוגל לניהול פרויקטי Apps Script:

```
התקנה גלובלית #  
npm install -g @google/clasp
```

התחברות לגוגל

```
clasp login
```

ייפתח דף פנוי להתחברות עם חשבון גוגל שלכם.

הפעלה API: אם מקבלים שגיאה, היכנסו ל-[הגדרות Apps Script](#) והפעילו את "Google API".



יצירת פרויקט Apps Script

3

צרו תיקייה חדשה

1

```
mkdir calendar-api && cd calendar-api
```

צרו פרויקט Apps Script

2

```
clasp create --title "Calendar API" --type webapp
```

appsscript.json | .clasp.json

זה ייצור קבצי

העתיקו את קוד ה-API

3

צרו קובץ `Code.gs` עם התוכן הבא:

```
/**  
 * Google Calendar API  
 * Deploy as Web App  
 */  
  
const AUTH_TOKEN = 'YOUR_SECRET_TOKEN_HERE';  
  
function doGet(e) {  
  const token = e.parameter.token;  
  if (token !== AUTH_TOKEN) {  
    return jsonResponse({ error: 'Unauthorized' });  
  }  
  
  const action = e.parameter.action || 'today';  
  let events;  
  
  switch (action) {  
    case 'today': events = getTodayEvents(); break;  
    case 'week': events = getWeekEvents(); break;  
    case 'upcoming':  
      const hours = parseInt(e.parameter.hours) || 4;
```

```
        events = getUpcomingEvents(hours);
        break;
    }

    return jsonResponse({ count: events.length, events });
}

//-ראו קוד מלא ב ... GitHub
```

הקוד המלא נמצא בקובץ `references/calendar-api.gs` בס킬.

4 הגדרו טוקן סודי

החליפו את `YOUR_SECRET_TOKEN_HERE` בסיסמה שתבחרו.

4

העלאה ופרסה

1

הלו את הקוד לגוגל

```
clasp push
```

2 פרסו כ-Web App

```
clasp deploy --description "Calendar API v1"
```

2

3 קבלו את ה-URL

```
clasp deployments
```

העתיקו את ה-URL שמסתיים ב- `/exec`

Deploy → לפתיחה בדף ופרסו שם דרך `clasp open`

אפשרות נוספת: הריצו



.New deployment

5 הגדרת הסקייל

אחרי שיש לכם URL וטוקן, בקשו מקלוד:

עם `calendar` עדכן את הסקייל:

URL: https://script.google.com/macros/s/YOUR_DEPLOYMENT_ID/exec

טוקון_שברחתם TOKEN:

או עדכנו ידנית את קובץ `h-pd.LL.SKILL` בתיקיית הסקייל.

דוגמאות לשימוש

אחרי ההגדרה, פשוט תשאלו את קלוד:

#בדיקה אירועים

"מה יש לי היום ביום"

"מה יש לי בשבוע"

"יש לי פגישה ב-4 שעות הקרובות"

#הוספה אירועים

"הוסף פגישה עם דני מחר ב-10:00 עד 11:00"

"קבע לי תוכורת לשיחה ביום רביעי ב-14:00"

פעולות זמינות

פרמטרים

טיאור

פעולה

אירועי היום

today

אירועי השבוע

week

(ברירת מחדל: 4 hours

hours

שעות קרובות

upcoming

end , start

טוויח ארויים

range

endTime , startTime , title

הוספה אירופ

add

פקודות Clasp שימושיות

פתרונות בעיות 

"Script API not enabled"

"Unauthorized"

- וודאו שהтокן בקוד תואם לטוקן שאתם שולחים
- בדקו שאין רוחקים מיותרים

"Not found"

- וודאו שאתם משתמשים ב-URL שמסתיים ב-`/exec`
- בדקו שהפרישה פעילה

אין הרשות ליום

וודאו שבפריסה בחורתם:

Execute as: **Me** •

Who has access: **Anyone** •

קוד מלא ב-GitHub 

← חזרה לדף הראשי

יש שאלות? פתחו issue ב-GitHub