USER'S GUIDE FROM AUTHOR

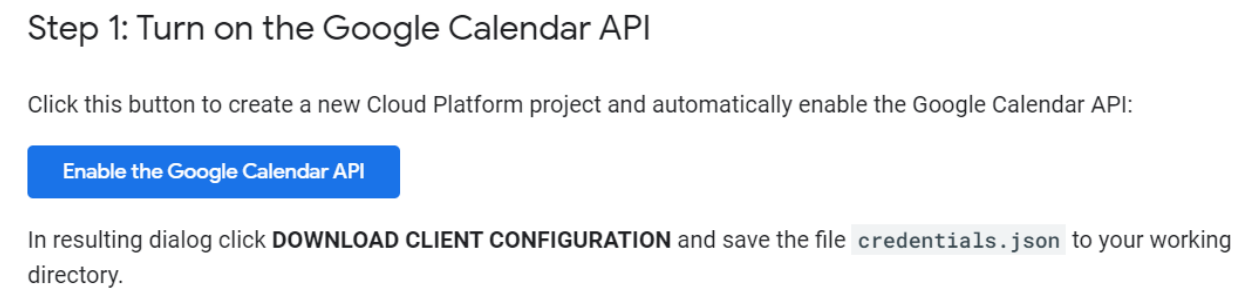
Project Source

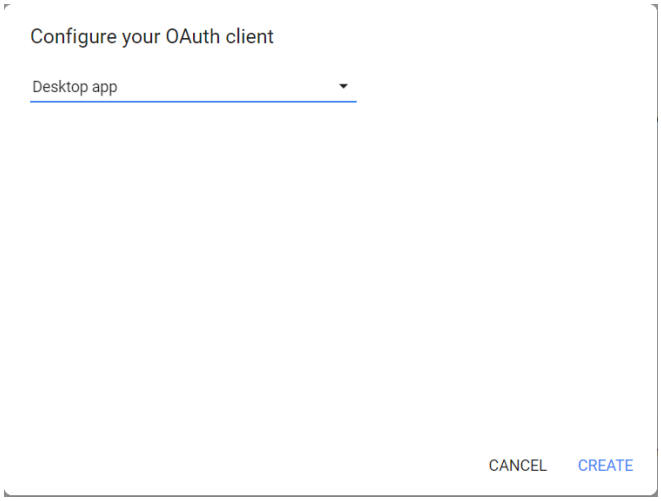
WHAT IS THIS PROJECT ABOUT?

Once I heard, Programmers will spend hours to automate something, they could have done it manually in minutes. Attending online classes daily is tough. Getting classes' notifications and looking for a meet link is frustrating sometimes. This project is about automating that. You run the code, your daily events on google calendar will be fetched and then before 2 minutes of any event, you'll get a notification in your computer's screen and a related link will be opened in your browser immediately. Before moving forward, I would like to point out one most important thing, This algorithm will fetch only G-meet links, but for any other type of schedules, you can add them manually in this CLI (Command Line Interface) application.

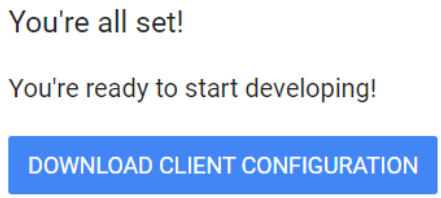
Proper Installation Guidelines

First goto, Google calendar API. Click one 'Enable the Google Calendar API'.





Enter a new project name, whatever you want and click 'Next'. Configure your OAuth client as 'Desktop App' and click on 'CREATE'. You're all set.



Click on 'DOWNLOAD CLIENT CONFIGURATION' and move this file to the folder where all .py of this project has been saved. Note that, the file name

of the downloaded client file should be 'credentials.json'. You're almost good to go. When you run an operation to fetch data from google calendar for the first time, you have to sign in. Sign-in through your account, you want to fetch the calendar data.

pip install

The following packages are used in the project. You need to install them if you haven't yet.

- webbrowser
- os
- sys
- time
- datetime
- datefinder
- json
- threading
- random
- pyperclip
- plyer
- re

You can easily google their installation and most of them come pre-installed with python.

Running the project and Menu

Open cmd and run the 'main.py' file. If you see this, then you're good to go.

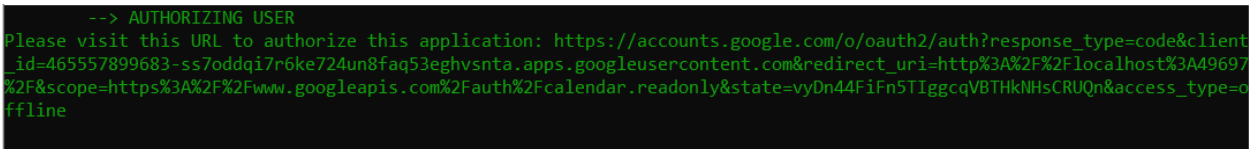
```
--> WELCOME TO YOUR ONLINE CLASSES MANAGER.
--> This app is still in beta version. Please report bugs and feedbacks to help us to improve this application.
--> Press ENTER continue
```

Hit 'enter' to continue.

```
**********************************************************************
*                            MENU                                    *
**********************************************************************
* Please select the task you want to perform                         *
*                                                                    *
*             1. Get today's schedule from google calender.          *
*             2. Add an event in your schedule locally.              *
*             3. Quick google calender events' automation            *
*             4. Generate a strong password                          *
*             5. Add an event in your google calender                *
*             6. Bug Report and Feedback                             *
*             7. Exit                                                *
**********************************************************************

    --> ENTER YOUR CHOICE (1-7):
```

Get today's schedule from google calendar : In this field, your system will access google calendar server and fetch today's events. When you load this field for the first time you'll be redirected to the sign-in page. You have to sign into your account and



allow it to access your google calendar. After this you're good to go. Once the data is fetched, You can add and delete events which you'll find at the very next operation in this field. At the end, you'll be asked to start 'Automation' and by agreeing on it, and a separate thread will be created. You'll get notifications and links will be opened in the browser according to your events' starting time. You'll be redirected to the menu and you're advised to not to close or exit from the menu. You can minimise it though.

Add an event in your schedule locally: In this field you can add events just for today. If you have added extra events by default, you can delete them too, before continuing for automation. After everything is set, You can start automating things.

Quick google calendar event automation: In this field, You don't need to edit, add or delete anything. So, e.g. if you have all events set up in your google calendar for the whole day, You can automate those events just by choosing this operation. It's a quick way to achieve a goal.

Generate a strong password: In this field, you can generate a strong password very quickly. You just need to give the length of password you want to generate. Then if you want to change default settings for password (i.e. upper letters, lower letters, numbers and special characters all will be included) you can change it. You can turn off maximum two of them. Once the password is generated (which is instant) you'll be notified and the generated password will be copied to your clipboard. You just need to paste it.

Add an event in your google calendar: Opens the official website of google calendar. In future, Manual options may be added.

Bug Report and Feedback: This area will open a github page of issues and a google form, where you can report a bug or give me feedback.

Exit: This field terminates the code. It is highly advised that if you have some background running automations, don't exit or cross the cmd/terminal, otherwise those processes will be killed.

ADDITIONAL AND HIDDEN FEATURES: Multithreading is a big highlight of this project, So multiple scheduled events can run parallely. On completion of each automation task, you'll be notified that all tasks have been completed. So, even if you have started automation of events, You can add other events manually or by editing the google calendar anytime, they will not affect each other. Note that, I haven't added any feature to stop an ongoing automation process, so be careful before starting it, otherwise you'll get irrelevant notifications and interference. You have to kill the whole code in order to get rid of it.

FUTURE UPDATES: I'm a student. This was a fun project to work on. This thing is working pretty well unless you decide to kill the code. In future, If I have people who want to work on this project, GUI is going to be the first thing that I'm gonna add. Bugs can be fixed. More features may be introduced. C++ or Java versions may be released. Or a web app may be released. It all depends on how people are gonna respond and community growth.

At the end, This code is open source. Feel free to tweek, or change things in it. Github contributions will be appreciated. And HAPPY CODING.