

Описание:

Переменные:

IDuserOAuth - Идентификатор клиента OAuth Google Cloud. JSON формат.

credsfile - файл для сохранения информации об авторизации.

Классы:

Class GAPIWorkspace - управляет авторизацией Google API Workspace, работает без GUI.

Свойства:

creds - Словарь с данными, полученными от API в информации об авторизации. Или False, в случае неудачи.

Методы:

`__init__` - получает на входе 2 параметра: IDuserOAuth - обязательный, filename - не обязательный. На выходе получаем объект с creds или false при неудаче. Если указан необязательный параметр filename, проверяет содержащийся в файле токен и другие данные об авторизации. При необходимости получает новый, и обновляет его при каждой инициализации объекта.

Пример кода:

```
IDuserOAuth={OAuth json data}

credsfile='./creds.json'

gconn=GAPIWorkspace(IDuserOAuth, filename=credsfile)
```

Class GCalendar - работает с календарями Google без GUI.

Свойства:

data - словарь, содержит все поля текущего календаря со значениями, полученными через API. По умолчанию false. Значения применяются только тогда, когда получены данные от API.

error - информация об ошибке, если метод вернул False. По умолчанию False.

Методы:

`__init__` - на входе получает экземпляр класса GAPIWorkspace, содержащий creds. Создает объект класса без связи с каким либо календарем.

create - на входе словарь с данными для создания нового календаря, где все поля не обязательные, на выходе словарь data (далее идентификатор календаря) или False в случае неудачи. Значения по умолчанию для полей: name='Новое мероприятие', description='', timezone='GMT+02:00', share=[]. Метод создает новый календарь с указанными значениями полей.

delete - на входе идентификатор календаря, на выходе True или False, в зависимости от результата. Удаляет указанный календарь и делает data=False.

select - на входе словарь с данными для поиска, где поля не обязательные; на выходе идентификатор первого календаря, соответствующего условиям запроса или False. Метод производит поиск календаря по указанным полям и возвращает первое соответствие. Дополнительное поле descr_inc содержит подстроку для поиска в строке description.

`edit` - на входе обязательные параметры идентификатор календаря и словарь с данными для редактирования; на выходе идентификатор измененного календаря. Метод меняет те настройки календаря, которые указаны в словаре с данными. Все поля в словаре с данными не обязательные.

`eventlist` - на входе обязательные параметры идентификатор календаря и словарь с данными для выборки событий; на выходе список словарей, каждый из которых содержит поля, возвращаемые от API по каждому отдельному событию (идентификатор события). Каждый идентификатор события совместим с идентификатором события класса `GEvent` (описание далее). Словарь с данными для выборки содержит обязательное поле `from` в формате unix timestamp и не обязательные: поле `till` в формате unix timestamp, по умолчанию значение `now()`, не обязательное поле `limit`, по умолчанию 10. Метод выбирает список событий, соответствующих критериям выборки.

`get` - на входе обязательные параметры идентификатор календаря, на выходе идентификатор календаря или `False` в случае неудачи. Метод получает свойства `data` из API на основании идентификатора календаря.

Пример кода:

```
calendar=GCalendar(gconn)

data={'name':'Календарь123', 'description':'Описание календаря',
'timezone':'GMT+03:00', 'share':['user@gmail.com',]}

c1=calendar.create(data)

calendar.delete(c1)

data={'name':'Календарь456', 'descr_inc':'Описание календаря'}

c2=calendar.select(data)

data={'name':'МойКалендарь', 'description':'Описание календаря',
'timezone':'GMT+02:00', 'share':['anotheruser@gmail.com',]}

c2=calendar.edit(c2, data)

calendar.get(c2)

event=GEvent(calendar)

elist=[]

data={'from':'1729223248', 'till':'1729335458', 'limit':10}

elist=calendar.eventlist(c2, data)

for e in elist:

    e=event.edit(e, alarm={'type':'message', 'time':'1800'})
```

Class GEvent - работает с событиями в календаре Google без GUI.

Свойства:

`data` - словарь, содержит все поля текущего события со значениями, полученными через API. По умолчанию `false`. Значения применяются только тогда, когда получены данные от API.

`error` - информация об ошибке, если метод вернул `False`. По умолчанию `False`.

Методы:

`__init__` - на входе получает экземпляр класса `GCalendar`, содержащий `data`. Создает объект класса без связи с каким либо событием.

`create` - на входе словарь с данными для создания нового события, где все поля не обязательные, на выходе словарь `data` (далее идентификатор события) или `False` в случае неудачи. Список полей в примере указан не полный, дополнить самостоятельно. Метод создает новое событие с указанными значениями полей.

`delete` - на входе идентификатор события, на выходе `True` или `False`, в зависимости от результата. Удаляет указанное событие и делает `data=False`.

`select` - на входе словарь с данными для поиска, где поля не обязательные; на выходе идентификатор первого события, соответствующего условиям запроса или `False`. Метод производит поиск события по указанным полям и возвращает первое соответствие. Дополнительное поле `descr_inc` содержит подстроку для поиска в строке `description`. Поле `timestamp` в формате `unix timestamp`.

`edit` - на входе обязательные параметры идентификатор события и словарь с данными для редактирования; на выходе идентификатор измененного события. Метод меняет те настройки события, которые указаны в словаре с данными. Все поля в словаре с данными не обязательные.

Пример кода:

```
calendar.get(c2)

event=GEvent(calendar)

edata={'name':'Мероприятие1', 'timestamp':'1729223248', 'duration':'3600',
'description':'Описание мероприятия', 'alarm':[{'type':'message', 'time':'1800'}]}

e1=event.create(edata)

event.delete(e1)

edata={'name':'Мероприятие2', 'timestamp':'1729223248', 'descr_inc':'Описание
мероприятия'}

e2=event.select(edata)

edata={'name':'МоеМероприятие', 'timestamp':'1729335458', 'duration':'1800',
'description':'Описание мероприятия', 'alarm':[{'type':'message', 'time':'3600'}]}

e2=event.edit(e2, edata)
```