

STEPWISE INSTRUCTIONS TO MAKE REQUESTS TO GOOGLE CALENDAR API

(Generate Google meet link and notify all the attendees accordingly)

NOTE : Following document is customized as per the needs of Windows user only.

STEP 1 :

Installing PHP 5.4 or greater with the command-line interface (CLI) and JSON extension

Referred link : <https://www.sitepoint.com/how-to-install-php-on-windows/>

STEP 2:

Download The Composer dependency management tool

Referred link : <https://www.javatpoint.com/how-to-install-composer-on-windows>

STEP 3 :

Create a Google Cloud Platform project with the API enabled.

Referred link : <https://developers.google.com/workspace/guides/create-project>

After setting up the OAuth Consent Screen for the project, download the credentials.json file.

STEP 4:

Create a folder in C:/xampp/htdocs and place the downloaded credentials.json file in that particular file.

STEP 5 :

Install the Google Client Library .

Process - > Type the following command in windows command line(in that particular folder itself) -

\$composer require google/apiclient:^2.0

STEP 6 :

Create a file named quickstart.php in your working directory and copy in the following code:

-----COPY & PASTE THE BELOW LINES-----

```
<?php
```

```
use function GuzzleHttp\Promise\all;
if (php_sapi_name() != 'cli') {
    throw new Exception('This application must be run on the command line.');
```

```
}

require __DIR__ . '/vendor/autoload.php';
```

```
/**
```

```
 * Returns an authorized API client.
 * @return Google_Client the authorized client object
 */
```

```
function getClient()
```

```
{
    $client = new Google_Client();
    $client->setApplicationName('Google Calendar API PHP Quickstart');
    $client->setScopes(Google_Service_Calendar::CALENDAR);
    $client->setAuthConfig(__DIR__.'/credentials.json');
    $client->setAccessType('offline');
    $client->setPrompt('select_account consent');
```

```
    // Load previously authorized token from a file, if it exists.
    // The file token.json stores the user's access and refresh tokens, and is
    // created automatically when the authorization flow completes for the first
    // time.
```

```
    $tokenPath = 'token.json';
    if (file_exists($tokenPath)) {
        $accessToken = json_decode(file_get_contents($tokenPath), true);
        $client->setAccessToken($accessToken);
```

```
    }
```

```

// If there is no previous token or it's expired.
if ($client->isAccessTokenExpired()) {
    // Refresh the token if possible, else fetch a new one.
    if ($client->getRefreshToken()) {
        $client->fetchAccessTokenWithRefreshToken($client->getRefreshToken());
    } else {
        // Request authorization from the user.
        $authUrl = $client->createAuthUrl();
        printf("Open the following link in your browser:\n%s\n", $authUrl);
        print 'Enter verification code: ';
        $authCode = trim(fgets(STDIN));

        // Exchange authorization code for an access token.
        $accessToken = $client->fetchAccessTokenWithAuthCode($authCode);
        $client->setAccessToken($accessToken);

        // Check to see if there was an error.
        if (array_key_exists('error', $accessToken)) {
            throw new Exception(join(', ', $accessToken));
        }
    }
    // Save the token to a file.
    if (!file_exists(dirname($tokenPath))) {
        mkdir(dirname($tokenPath), 0700, true);
    }
    file_put_contents($tokenPath, json_encode($client->getAccessToken()));
}
return $client;
}

```

```

// Get the API client and construct the service object.
$client = getClient();
$service = new Google_Service_Calendar($client);

```

```

// Print the next 10 events on the user's calendar : SEE FROM QUICKSTART.PHP
// Creating Meet Event

```

```

$event = new Google_Service_Calendar_Event(array(
    'summary' => 'Google Meet Invite Created Using Php by Rahul Kumar',
    'location' => 'Ipac Jubileehills Hyderabad',
    'description' => 'Description: Mr. XYZ has an interview with Mr. Pradip Nandwana,
IPAC',
    'start' => array(
        'dateTime' => '2021-11-11T09:00:00-00:00',
        'timeZone' => 'Asia/Kolkata',
    ),
    'end' => array(
        'dateTime' => '2021-11-11T09:30:00-00:00',
        'timeZone' => 'Asia/Kolkata',
    ),
    'recurrence' => array(
        'RRULE:FREQ=DAILY;COUNT=1'
    ),
    'attendees' => array(
        array('email' => 'rahulkumariiit9999@gmail.com'),
        array('email' => 'rahul.kumar@indianpac.com'),
        //array('email'=>'appresearch34268@gmail.com'),
        array('email'=>'sourabh.patil@indianpac.com'),
    ),
    'conferenceData' => [
        'createRequest' => [
            'requestId' => 'testing123',
            'conferenceSolutionKey' => ['type' => 'hangoutsMeet']
        ]
    ],
    'reminders' => array(
        'useDefault' => FALSE,
        'overrides' => array(
            array('method' => 'email', 'minutes' => 24 * 60),
            array('method' => 'popup', 'minutes' => 10),
        ),
    ),
));

$calendarId = 'primary';

```

```
$event = $service->events->insert($calendarId, $event,array('conferenceDataVersion'
=> 1,'sendUpdates'=>'all'));
printf('Event created: %s\n', $event->htmlLink);
```

```
?>
```

```
//-----
```

STEP 7 :

Make the following changes in following files to handle the mentioned errors -

7.1 : Uncaught TypeError()

Reference solutionlink :

<https://stackoverflow.com/questions/50898330/guzzle-error-count-parameter-must-be-an-array-or-an-object-that-implements-co/50903672#50903672?newreg=dd979eea3d614bc690008ed8de9c8e46>

7.2 : ssl error in CurlHandler

Reference solutionlink :

<https://stackoverflow.com/questions/29822686/curl-error-60-ssl-certificate-unable-to-get-local-issuer-certificate?noredirect=1&lq=1>

7.3 Implode error in Resource.php

Reference solutionlink :

<https://stackoverflow.com/questions/65523083/error-on-integrating-php-and-google-sheets-api>

STEP 8 :

Now open your console and run the following command -

\$php quickstart.php

Note : If you are running the application for the first time, you will get an “authurl” in console, open that link in chrome and get the success code and paste it back in console to get the final link of the created event[Part of OAuth2.0 required for Google API Integration]

-----END OF THE DOCUMENT-----