Step 1: Open the below URL:

https://console.cloud.google.com/apis/library

Step 2: Search and open Google Calendar API library

Graphical user interface, application, Teams

Description automatically generated

Step 3: Select Google Calendar API to Check if this API enabled, if not then need to enable this api, then click on Manage

Graphical user interface, text, application

Description automatically generated

Step 4: Now click on Credential link in the left hand side and then click on Create Credentials

Graphical user interface, application

Description automatically generated

Select OAuth client ID

Graphical user interface, text, application, email

Description automatically generated

Step 5: Select Web application as Application type

Graphical user interface, text, application, email

Description automatically generated

Step 6: You can provide any name instead of <Web client 2>. Keep Authorized Redirect URIs blank , this will require later, and then click on Create.

Once you click on Create, system will ask to create OAuth Consent Screen, please create consent screen and complete the process.

Graphical user interface, text, application, email

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generated

Step 7: Now after creation of Consent screen your App is ready. Now copy the Client ID and Client Secret

Step 8: Go to Salesforce page and now create an Auth.Provider

Provide Consumer Key 🡺 Client ID

Consumer Secret 🡺 Client Secret

And the other fields like Authorize Endpoint URL etc.

Graphical user interface, text, application, email

Description automatically generated

Now after save Auth.Providers, go to the bottom of the page and copy callback url and then open the app you created just few steps before, then go to the Authorized redirect URIs and paste it over there.

Graphical user interface, text, email

Description automatically generated

Step 9: Now create Named Credential.

Select Identity Type = <named principal>

Select Authentication protocol = <OAuth 2.0>

Provide value of Scope = <openid https://www.googleapis.com/auth/calendar>

and then click on Save button

Graphical user interface, text, application, email

Description automatically generated

Step 10: Try to execute the following piece of code from Anonymous window

String reqbody = '{' + '\n' ;

reqbody+= ' "end": { ' + '\n' ;

reqbody+= ' "dateTime": "2021-11-15T09:00:00-05:00", ' + '\n' ;

reqbody+= ' "timeZone": "Asia/Kolkata" ' + '\n' ;

reqbody+= ' }, ' + '\n';

reqbody+= ' "start": { ' + '\n';

reqbody+= ' "dateTime": "2021-11-15T22:00:00-05:00", ' + '\n';

reqbody+= ' "timeZone": "Asia/Kolkata" ' + '\n';

reqbody+= ' }, ' + '\n';

reqbody+= ' "description": "Sample Event", ' + '\n';

reqbody+= ' "summary": "Google Calendar Demo Testing" ' + '\n';

reqbody+= ' } ' ;

String calendarId = 'xxxxxxx@gmail.com'; // here you have to provide your google username

//String scope= 'https://www.googleapis.com/auth/calendar';

//String endPoint = 'https://www.googleapis.com/calendar/v3/calendars/' + calendarId + '/events';

Http http = new Http();

HttpResponse HttpRes = new HttpResponse();

HttpRequest httpReq = new HttpRequest();

httpReq.setEndpoint('callout:SalesforceGoogleCalendarAPI' + '/' + calendarId + '/events' );

httpReq.setMethod('POST');

httpReq.setHeader('Content-Type', 'application/json');

httpReq.setBody(reqbody);

try{

HttpRes = http.send(httpReq);

if(httpRes.getStatusCode() == 200){

System.debug('HtteRes '+HttpRes.getBody());

}else{

System.debug( 'error code ' + HttpRes.getBody() );

}

}catch(Exception e){

}