Integrating Google Calendar API with Salesforce via OAuth 2.0

1. Google Cloud Project Setup

- Go to https://console.cloud.google.com
- Create a new project (e.g., 'Salesforce Integration')
- Navigate to 'APIs & Services > Credentials'
- Create OAuth 2.0 Client ID (Web Application)
- Set the Authorized Redirect URI to:

  https://<your-org>.my.salesforce.com/services/authcallback/<AuthProviderName>
- Enable Google Calendar API in 'Library'

2. Create Auth Provider in Salesforce

- Go to Setup > Auth Providers > New
- Type: OpenID Connect
- Name: GoogleTest (example)
- Authorize URL: https://accounts.google.com/o/oauth2/auth?access_type=offline&prompt=consent
- Token URL: https://oauth2.googleapis.com/token
- User Info URL: https://openidconnect.googleapis.com/v1/userinfo
- Client ID & Secret: From Google Cloud Console
- Scope: openid email profile https://www.googleapis.com/auth/calendar.readonly
- Save the Auth Provider and copy the Callback URL

3. Update Google Console Redirect URI

- Go back to Google Console
- Paste the Salesforce callback URI into the Authorized Redirect URIs
- Save changes

4. Create Named Credential in Salesforce

- Go to Setup > Named Credentials > New
- Label: GoogleCalendarNamed
- URL: https://www.googleapis.com/calendar/v3/

- Identity Type: Named Principal
- Authentication Protocol: OAuth 2.0
- Authentication Flow Type: Authorization Code
- Auth Provider: GoogleTest
- Scope: https://www.googleapis.com/auth/calendar.readonly
- Save and click 'Allow Salesforce to access this Named Credential'
- Confirm no warning about missing refresh token

5. Call Google API from Apex

Sample Apex:

```
HttpRequest req = new HttpRequest();
req.setEndpoint('callout:GoogleCalendarNamed/calendars/primary/events?maxResults=5');
req.setMethod('GET');
Http http = new Http();
HttpResponse res = http.send(req);
System.debug(res.getBody());
```

6. Common Troubleshooting Tips

- Bad_Id: Auth Provider name does not match callback path
- redirect_uri_mismatch: Redirect URI not registered in Google Console
- No_Oauth_State: You manually triggered the auth URL; use Named Credential auth flow instead
- No Refresh Token: Add access_type=offline and prompt=consent to authorize URL