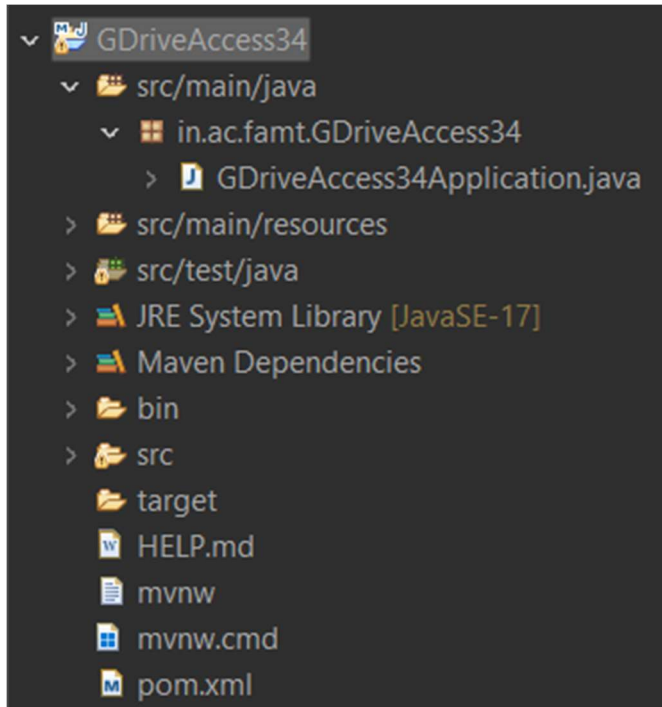


Practical No.06

Implementation of Cloud Computing Services

Q.1 Write a java program to access the files from your Google drive account and read and write the file contents from your program.



Code:

GDriveAccess34Application.java

```
package in.ac.famt.GDriveAccess34;
```

```
import org.springframework.boot.SpringApplication;
```

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

```
@SpringBootApplication
```

```
public class GDriveAccess34Application {
```

```
    public static void main(String[] args) {
```

```
        SpringApplication.run(GDriveAccess34Application.class, args);
```

```
    }
```

```
}
```

GDriveAccess34ApplicationTests.java

```
package in.ac.famt.GDriveAccess34;

import com.google.api.client.auth.oauth2.Credential;
import com.google.api.client.extensions.java6.auth.oauth2.AuthorizationCodeInstalledApp;
import com.google.api.client.extensions.jetty.auth.oauth2.LocalServerReceiver;
import com.google.api.client.googleapis.auth.oauth2.GoogleAuthorizationCodeFlow;
import com.google.api.client.googleapis.auth.oauth2.GoogleClientSecrets;
import com.google.api.client.googleapis.javanet.GoogleNetHttpTransport;
import com.google.api.client.http.AbstractInputStreamContent;
import com.google.api.client.http.InputStreamContent;
import com.google.api.client.http.javanet.NetHttpTransport;
import com.google.api.client.json.JsonFactory;
import com.google.api.client.json.jackson2.JacksonFactory;
import com.google.api.client.util.store.FileDataStoreFactory;
import com.google.api.services.drive.Drive;
import com.google.api.services.drive.DriveScopes;
import com.google.api.services.drive.model.File;
import com.google.api.services.drive.model.FileList;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.security.GeneralSecurityException;
import java.util.Collections;
import java.util.List;

//@SpringBootApplication
```

```

public class GDriveAccess34ApplicationTests {

    private static final JsonFactory JSON_FACTORY = JacksonFactory.getDefaultInstance();

    // Directory to store user credentials for this application.

    //private static final java.io.File CREDENTIALS_FOLDER = new
    java.io.File(System.getProperty("user.home"), "credentials");

    private static final java.io.File CREDENTIALS_FOLDER = new java.io.File("Z:\\MCA-DSCC");


    // Global instance of the scopes required by this program.

    private static final List<String> SCOPES = Collections.singletonList(DriveScopes.DRIVE);

    //https://developers.google.com/resources/api-
    libraries/documentation/drive/v2/java/latest/com/google/api/services/drive/DriveScopes.html


    private static Credential getCredentials(final NetHttpTransport HTTP_TRANSPORT) throws
    IOException {

        java.io.File clientSecretFilePath = new java.io.File("Z:\\MCA-DSCC\\credits.json");

        if (!clientSecretFilePath.exists()) {

            throw new FileNotFoundException("Please copy credentials.");

        }

        // Load client secrets.

        InputStream in = new FileInputStream(clientSecretFilePath);

        GoogleClientSecrets clientSecrets = GoogleClientSecrets.load(JSON_FACTORY, new
        InputStreamReader(in));


        // Build flow and trigger user authorization request.

        GoogleAuthorizationCodeFlow flow = new
        GoogleAuthorizationCodeFlow.Builder(HTTP_TRANSPORT, JSON_FACTORY,

            clientSecrets, SCOPES).setDataStoreFactory(new
        FileDataStoreFactory(CREDENTIALS_FOLDER))

```

```

        .setAccessType("offline").build());

//System.out.println("Flow info - " + flow.toString());

return new AuthorizationCodeInstalledApp(flow, new
LocalServerReceiver()).authorize("user");
}

public static void main(String[] args)throws IOException, GeneralSecurityException {

    // 1: Build a new authorized API client service.

    final NetHttpTransport HTTP_TRANSPORT =
GoogleNetHttpTransport.newTrustedTransport();

    // 2: Read client_secret.json file & create Credential object.

    Credential credential = getCredentials(HTTP_TRANSPORT);

    // 3: Create Google Drive Service.

    Drive service = new Drive.Builder(HTTP_TRANSPORT, JSON_FACTORY,
credential).setApplicationName("GDrive Access").build();

    System.out.println("----" + service.getApplicationName() + "----");

    // Print the names and IDs for up to 10 files.

    FileList result = service.files().list().setPageSize(2).setFields("nextPageToken, files(id,
name)").execute();

    String lastFile = "";

    List<File> files = result.getFiles();

    if (files == null || files.isEmpty()) {

        System.out.println("No files found.");

    } else {

        System.out.println("Files:");

        for (File file : files) {

            System.out.printf("%s (%s)\n", file.getName(), file.getId());

            lastFile = file.getId();

        }
    }
}

```

```

    }

    //Create Folder on Google Drive
    File fileMetadata = new File();
    fileMetadata.setName("MyFolderData");
    fileMetadata.setMimeType("application/vnd.google-apps.folder");
    //fileMetadata.setParents(folderIdParent);
    File file = service.files().create(fileMetadata).setFields("id, name").execute();
    if(file != null)
        System.out.println("Folder Created..");

    //Creating a file on GDrive
    java.io.File uploadFileContent = new java.io.File("Z:\\MCA-DSCC\\textUpload.txt");
    String contentType = "text/plain";
    AbstractInputStreamContent uploadStreamContent = new
    InputStreamContent(contentType,new FileInputStream(uploadFileContent));

    fileMetadata = new File();
    fileMetadata.setName("MyGDriveJavaFile.txt");

    file = service.files().create(fileMetadata, uploadStreamContent).setFields("id,
    webContentLink, webViewLink, parents").execute();

    if(file != null) {
        System.out.println("File Created..");
        System.out.println("WebContentLink: " + file.getWebContentLink() );
        System.out.println("WebViewLink: " + file.getWebViewLink() );
    }
}
}
}

```

Pom.xml

```

<!-- https://mvnrepository.com/artifact/com.google.apis/google-api-services-drive -->
    <dependency>

```

```
<groupId>com.google.apis</groupId>
<artifactId>google-api-services-drive</artifactId>
<version>v3-rev103-1.23.0</version>
</dependency>
```

```
-> <!-- https://mvnrepository.com/artifact/com.google.api-client/google-api-client -
```

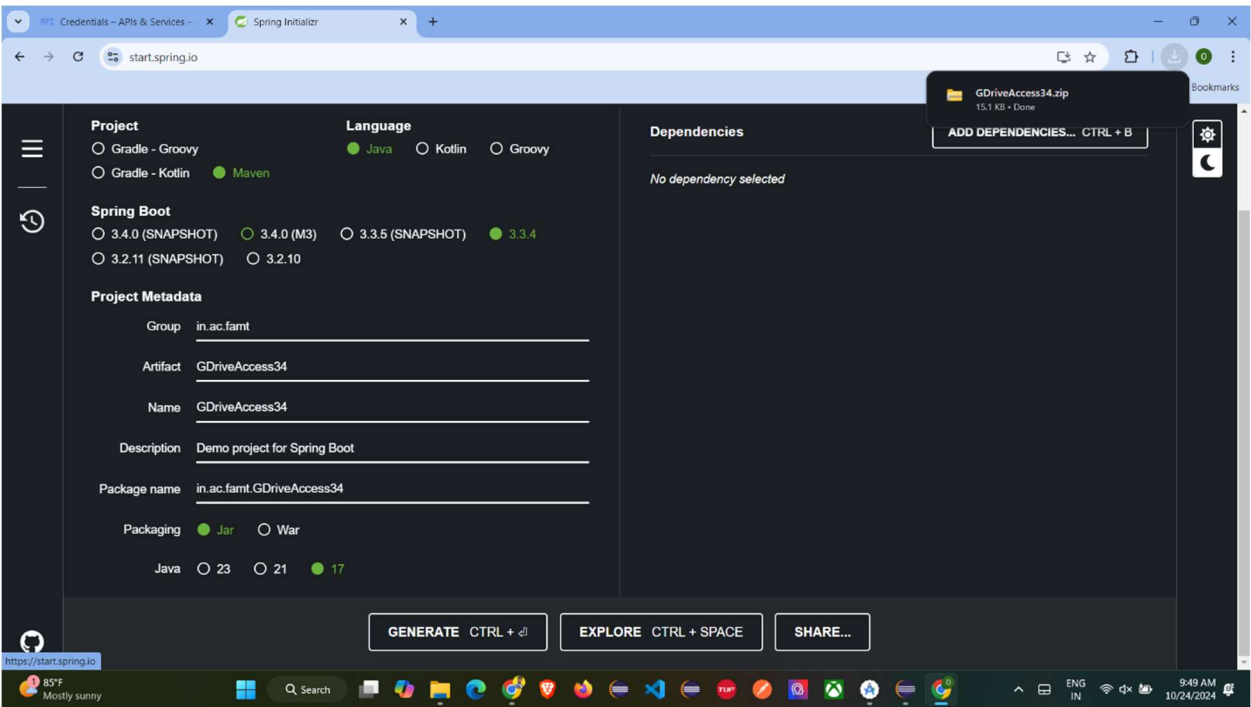
```
<dependency>
<groupId>com.google.api-client</groupId>
<artifactId>google-api-client</artifactId>
<version>1.23.0</version>
</dependency>
```

```
client-jetty -->
```

```
<dependency>
<groupId>com.google.oauth-client</groupId>
<artifactId>google-oauth-client-jetty</artifactId>
<version>1.23.0</version>
</dependency>
```

Output:

Download Spring Boot File



Home » [com.google.apis](#) » google-api-services-drive



Drive API V3 Rev197 1.25.0

Drive API V3 Rev197 1.25.0

License	Apache 2.0
Tags	google service api
Ranking	#4226 in MvnRepository (See Top Artifacts)
Used By	111 artifacts

dsc-439603

Search (/) for resources, docs, products, and more

Search

1

A

Credentials

+ CREATE CREDENTIALS

DELETE

RESTORE DELETED CREDENTIALS

Create credentials to access your enabled APIs. [Learn more](#)

API Keys

<input type="checkbox"/>	Name	Creation date ↓	Restrictions	Actions
No API keys to display				

OAuth 2.0 Client IDs

<input type="checkbox"/>	Name	Creation date ↓	Type	Client ID	Actions
<input type="checkbox"/>	Desktop client 1	Oct 24, 2024	Desktop	1088174696225-uan...	<div></div> <div></div> <div></div>

Service Accounts

[Manage service accounts](#)

<input type="checkbox"/>	Email	Name ↑	Actions
No service accounts to display			

OAuth client created

The client ID and secret can always be accessed from Credentials in APIs & Services



OAuth access is restricted to the [test users](#) listed on your [OAuth consent screen](#)

Client ID	1088174696225-uanha17lajdkkj0lrlee3ptkis6mqj3f.apps.googleusercontent.com
Client secret	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
Creation date	October 24, 2024 at 9:42:44 AM GMT+5
Status	<div>Enabled</div>

DOWNLOAD JSON

OK



Create OAuth client ID

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information. [Learn more](#) about OAuth client types.

Application type *

Desktop app ▾

Name *

Desktop client 1

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

Note: It may take 5 minutes to a few hours for settings to take effect

CREATE

CANCEL



Credentials

+ CREATE CREDENTIALS

DELETE

RESTORE DELETED CREDENTIALS

Create credentials to access your project

API Keys



Name

No API keys to display

OAuth 2.0 Client IDs



Name

Creation date ↓

No OAuth clients to display

API key

Identifies your project using a simple API key to check quota and access

OAuth client ID

Requests user consent so your app can access the user's data

Service account

Enables server-to-server, app-level authentication using robot accounts

Help me choose

Asks a few questions to help you decide which type of credential to use

Authorized domains

Not provided

Contact email addresses

a230157@famt.ac.in

Scopes

EDIT

API ↑	Scope	User-facing description
No rows to display		

Test users

EDIT

1 user (1 test, 0 other) / 100 user cap

Filter

Enter property name or value

User information

a230157@famt.ac.in

BACK TO DASHBOARD

dsc-439603

Search (/) for resources, docs, products, and more



Edit app registration

OAuth consent screen — Scopes — Test users — 4 Summary

OAuth consent screen

EDIT

User type

External

App name

GoogleDrive2024AB

Support email

a230157@famt.ac.in

App logo

Not provided

Application homepage link

Not provided

Application privacy policy link

Not provided

Application terms of service link

Not provided

Test users

While publishing status is set to "Testing", only test users are able to access the app. Allowed user cap prior to app verification is 100, and is counted over the entire lifetime of the app. [Learn more](#)

+ ADD USERS

Filter Enter property name or value



User information

a230157@famt.ac.in



SAVE AND CONTINUE

CANCEL

[Console](#) to check if your domains are authorized. [Learn more](#) about the authorized domain limit.

+ ADD DOMAIN

Developer contact information

Email addresses *

a230157@famt.ac.in



These email addresses are for Google to notify you about any changes to your project.

SAVE AND CONTINUE

CANCEL

Google Cloud

dsc-439603

Search (/) for resources, docs, products, and more

Search

APIs & Services

Enabled APIs & services

Library

Credentials

OAuth consent screen

Page usage agreements

Edit app registration

1 OAuth consent screen

2 Scopes

3 Test users

4 Summary

App information

This shows in the consent screen, and helps end users know who you are and contact you

App name *

GoogleDrive2024AB

The name of the app asking for consent

User support email *

a230157@famt.ac.in

For users to contact you with questions about their consent. [Learn more](#)

API APIs & Services

- Enabled APIs & services
- Library
- Credentials
- OAuth consent screen
- Page usage agreements

OAuth consent screen

Choose how you want to configure and register your app, including your target users. You can only associate one app with your project.

User Type

☐ Internal ☒ ?

Only available to users within your organization. You will not need to submit your app for verification. [Learn more about user type](#)

External ?

Available to any test user with a Google Account. Your app will start in testing mode and will only be available to users you add to the list of test users. Once your app is ready to push to production, you may need to verify your app. [Learn more about user type](#)

CREATE

Let us know what you think about our OAuth experience



Google Drive API

Google Enterprise API

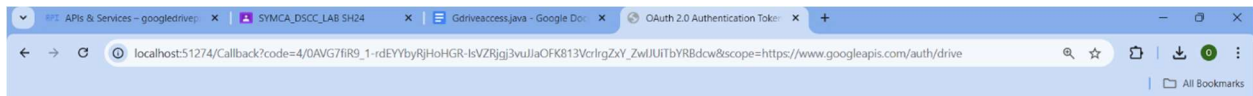
Create and manage resources in Google Drive.

MANAGE

[TRY THIS API](#) 

API Enabled

```
WARNING: unable to change permissions for owner: Z:\MCA-DSCC
10:30:20.889 [main] INFO org.mortbay.log -- Logging to Logger[org.mortbay.log] via org.mortbay.log.S
10:30:20.895 [main] INFO org.mortbay.log -- jetty-6.1.26
10:30:20.920 [main] INFO org.mortbay.log -- Started SocketConnector@localhost:51274
Please open the following address in your browser:
  https://accounts.google.com/o/oauth2/auth?access_type=offline&client_id=950807015856-
  tv1jbfd3ucr890lq71bakn4c1cq8s6ba.apps.googleusercontent.com&redirect_uri=http://localhost:51274/Call
  ponse_type=code&scope=https://www.googleapis.com/auth/drive
Attempting to open that address in the default browser now...
10:30:52.564 [main] INFO org.mortbay.log -- Stopped SocketConnector@localhost:51274
----GDrive Access----
Files:
Practical No. 9 (1r6mL9_VH3E2kj9QWz2Rngn9ZPKwSEIGUSazIZDs5pzY)
Practical No. 8 (194euEiDmK2DNIUY53p-rVPJBVhHrqEwE1ZHyd2V29xI)
```



Received verification code. You may now close this window.

```
<terminated> GDriveAccess34ApplicationTests [Java Application] C:\software\eclipse-java-2024-06-R-win32-x86_64\eclipse\plugins\org.eclipse.just.openjdk.hotspot.jre.full.win32.x86_64.21.0.3.v20240426-1530\jre\bin\javaw.exe (Oct 24, 2024, 10:39:38 AM)
Oct 24, 2024 10:39:38 AM com.google.api.client.util.store.FileDataStoreFactory setPermissionsToOwnerOnly
WARNING: unable to change permissions for everybody: Z:\MCA-DSCC
Oct 24, 2024 10:39:38 AM com.google.api.client.util.store.FileDataStoreFactory setPermissionsToOwnerOnly
WARNING: unable to change permissions for owner: Z:\MCA-DSCC
----GDrive Access----
Files:
MyFolderUsingJava (1-Qz7_wY76FHJw6TF1laLYcvXLWXjneTF)
MyFolderUsingJava (16whH0kg8BTfuRHnf18uUNPmxXAI3S7DG)
Folder Created..
File Created..
WebContentLink: https://drive.google.com/uc?id=160WTYtqXf1KNZN84-UbCOicwbx_pLeAR&export=download
WebViewLink: https://drive.google.com/file/d/160WTYtqXf1KNZN84-UbCOicwbx_pLeAR/view?usp=drivesdk
```

