



DGIN 5201

Digital Transformation

Lab 4

GitLab and Git

Assignment Submission

Time and date:
11:35–12:25 and
13:05–13:55, 31-Jan-2025
Location: Goldberg CS
134 and 143

Image: DALL-E. Bing Image Creator. Generated by AI

Lab Overview: GitLab and Git

- Introduction to Git, GitLab, and GitHub
- Hands-on exercise with GitLab server in FCS
- Finalizing submission of labs to GitLab
- Last step of Assignment 1
- Your project repositories will be set up, and you will be encouraged to use it
- Some project deliverables will be required using your project repository in GitLab

What is GitLab?

- It is based on Git, a source version control system
- A source version control system is used
 - ▶ to store and manage different versions of code
 - ▶ to provide collaborative platform for software developers
- GitLab is based on Git and provides a web interface
- Similar to GitHub in this sense
- Provides Continuous Integration (CI) and Continuous Delivery (CD) of code
- A lot of material on Git and GitLab can be found on the Web

Step 1. Logging into DalFCS GitLab Website

- Open your Web browser and go to:
`https://git.cs.dal.ca`

DalFCS Git

Git repos for individual and group use.

Login using your [CSID](#) username & password. You can also check/update your login credentials and check if your account has become locked (i.e. due to repeated password errors) at the [CSID](#) page.

Contact the [DalFCS Helpdesk](#) at cshelp@cs.dal.ca for support requests, questions, etc.

If necessary, visit [Email Reconfirm](#) page to confirm your email address.

CSID Standard

Username

Password

Remember me

Sign in

Step 2. Find your CSID Course Project in DGIN5201 Group

- The next step is to find your course project
- It is in the course group for the current term
- Its name is your CSID
- The next slide shows how the page with your projects approximately looks like

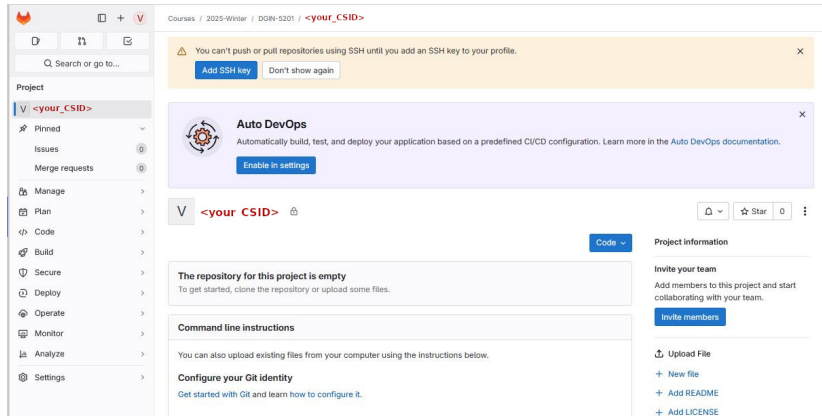
Approximate Page with Your Projects

- an approximate look of the page with your projects

The screenshot displays a web application interface. On the left is a sidebar with a navigation menu under the heading "Your work". The menu items are: Projects (highlighted), Groups, Issues, Merge requests, To-Do List, Milestones, Snippets, Activity, Workspaces, Environments, Operations, and Security. The main content area is titled "Your work > Projects". Below the title, there are filters: "Yours 1", "Starred 0", and "Pending deletion". A "Filter by name" input field is on the right. Below the filters, there are two tabs: "All" (selected) and "Personal". A single project is listed with a red square icon, the path [Courses / 2024-Winter / dgin5201 / <your CSID>](#), and a "Maintainer" role. A tooltip with the text "<your CSID>" is visible over the path.

Your Course GitLab Project (Repository)

- It is named as your CSID and in the course group (2025-winter/dgin-5201/<your_csid>)
- URL: `https://git.cs.dal.ca/courses/2025-winter/dgin-5201/<your_csid>`

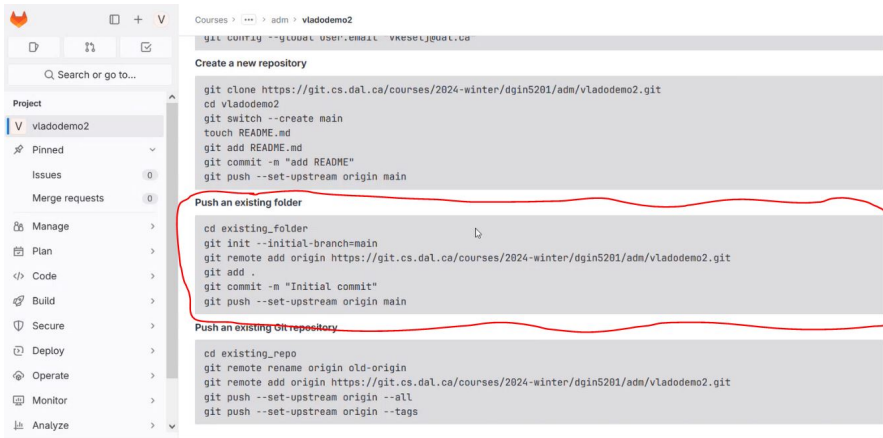


The screenshot shows the GitLab interface for a repository. The breadcrumb path is "Courses / 2025-Winter / DGIN-5201 / <your_CSID>". A notification banner at the top states: "You can't push or pull repositories using SSH until you add an SSH key to your profile." with buttons for "Add SSH key" and "Don't show again". Below this is an "Auto DevOps" section with a gear icon and the text "Automatically build, test, and deploy your application based on a predefined CI/CD configuration. Learn more in the Auto DevOps documentation." with an "Enable in settings" button. The repository name is displayed as "<your CSID>". A "Code" dropdown menu is visible. The main content area shows "The repository for this project is empty" with instructions to clone or upload files. Below this are sections for "Command line instructions" and "Configure your Git identity". On the right, the "Project information" section includes "Invite your team" with an "Invite members" button, and "Upload File" with options for "New file", "Add README", and "Add LICENSE". A left sidebar contains navigation options like "Pinned", "Issues", "Merge requests", "Manage", "Plan", "Code", "Build", "Secure", "Deploy", "Operate", "Monitor", "Analyze", and "Settings".

Step 3: Uploading your Files from timberlea

- In this step you should upload your lab files from timberlea into the GitLab server
- The instructions are shown in the GitLab page
- You should also open another command-line window for ssh login to timberlea

Instructions to Upload our Files



The screenshot shows a terminal window with a sidebar on the left. The sidebar contains a search bar and a list of project actions: Project (vladodemo2), Pinned, Issues (0), Merge requests (0), Manage, Plan, Code, Build, Secure, Deploy, Operate, Monitor, and Analyze. The main terminal area shows the following content:

```
Courses > ... > adm > vladodemo2
```

```
git config --global user.email vkesel@cs.dal.ca
```

Create a new repository

```
git clone https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2.git
cd vladodemo2
git switch --create main
touch README.md
git add README.md
git commit -m "add README"
git push --set-upstream origin main
```

Push an existing folder

```
cd existing_folder
git init --initial-branch=main
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2.git
git add .
git commit -m "Initial commit"
git push --set-upstream origin main
```

Push an existing Git repository

```
cd existing_repo
git remote rename origin old-origin
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2.git
git push --set-upstream origin --all
git push --set-upstream origin --tags
```

Login to timberlea Server

- In another window, we login to timberlea server
- It will probably be a command-line window or terminal window in which we type the command:

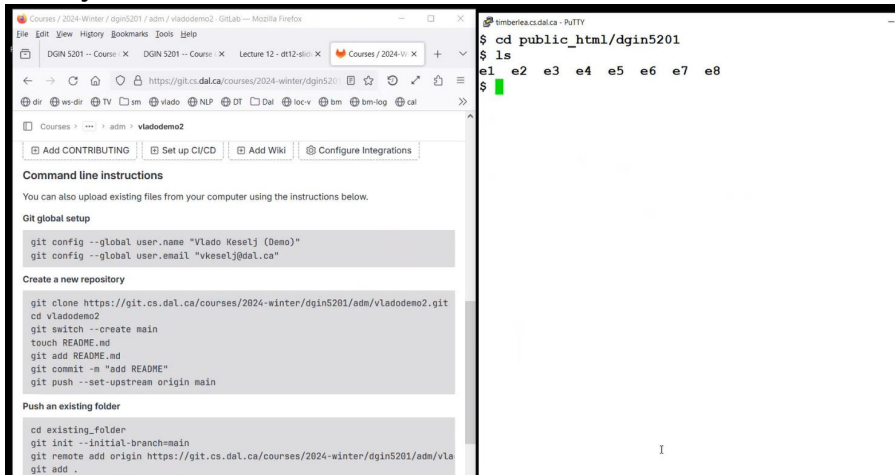
```
ssh <your_csid>@timberlea.cs.dal.ca
```

where instead of `<your_csid>` you should use your own CSID

- or maybe you can use PuTTY, mobaxterm, or other SSH application
- You should try to have two windows: the web browser with GitLab, and the command-line window

Going to your Labs Directory

- Change directory to `public_html/dgin5201` and list contents
- your screen with two windows could look as follows:



The screenshot shows two windows side-by-side. The left window is a Mozilla Firefox browser displaying the GitHub repository page for 'timberlea.cs.dal.ca / courses/2024-winter/dgin5201'. The right window is a PuTTY terminal session.

Browser Window:

- Address bar: `https://git.cs.dal.ca/courses/2024-winter/dgin5201`
- Path: `Courses > ... > adm > vladodemo2`
- Buttons: `Add CONTRIBUTING`, `Set up CI/CD`, `Add Wiki`, `Configure Integrations`
- Section: **Command line instructions**
- Text: "You can also upload existing files from your computer using the instructions below."
- Section: **Git global setup**
- Code block:

```
git config --global user.name "Vlado Keselj (Demo)"
git config --global user.email "vkeselj@dal.ca"
```
- Section: **Create a new repository**
- Code block:

```
git clone https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2.git
cd vladodemo2
git switch --create main
touch README.md
git add README.md
git commit -m "add README"
git push --set-upstream origin main
```
- Section: **Push an existing folder**
- Code block:

```
cd existing_folder
git init --initial-branch=main
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vla
git add .
```

Terminal Window:

- Shell: `timberlea.cs.dal.ca - PuTTY`
- Command: `$ cd public_html/dgin5201`
- Command: `$ ls`
- Output: `e1 e2 e3 e4 e5 e6 e7 e8`
- Prompt: `$`

Create Directory `git`

- make directory `git`
- make sure that the directory has permissions `rx` for the user
- the next screenshot shows how it would look on my screen

```
$ ls
e1 e2 e3 e4 e5 e6 e7 e8
$ mkdir git
$ ls -ltra
total 8
drwx--x--x  2 vladodemo csfac   43 Feb  6 13:16 e1
drwx--x--x  3 vladodemo csfac   59 Feb  6 13:27 e2
drwx--x--x  3 vladodemo csfac   22 Feb  8 08:49 ..
drwx--x--x  3 vladodemo csfac  121 Feb  8 13:59 e3
drwx--x--x  3 vladodemo csfac  121 Feb  8 13:59 e4
drwx--x--x  3 vladodemo csfac  178 Feb  8 15:26 e5
drwx--x--x  3 vladodemo csfac  209 Feb  8 15:39 e6
drwx--x--x  3 vladodemo csfac  209 Feb 13 14:56 e7
drwx--x--x  3 vladodemo csfac 4096 Feb 13 15:34 e8
drwx--x--x  2 vladodemo csfac    6 Feb 15 14:57 git
drwx--x--x 13 vladodemo csfac  120 Feb 15 14:57 .
$ █
```

Copy directories e1...e8 into directory git

- We first rsync directory e1:

```
timberlea.cs.dal.ca - PuTTY
$ ls
e1 e2 e3 e4 e5 e6 e7 e8
$ mkdir git
$ ls -ltra
total 8
drwx--x--x  2 vladodemo csfac    43 Feb  6 13:16 e1
drwx--x--x  3 vladodemo csfac    59 Feb  6 13:27 e2
drwx--x--x  3 vladodemo csfac    22 Feb  8 08:49 ..
drwx--x--x  3 vladodemo csfac   121 Feb  8 13:59 e3
drwx--x--x  3 vladodemo csfac   121 Feb  8 13:59 e4
drwx--x--x  3 vladodemo csfac   178 Feb  8 15:26 e5
drwx--x--x  3 vladodemo csfac   209 Feb  8 15:39 e6
drwx--x--x  3 vladodemo csfac   209 Feb 13 14:56 e7
drwx--x--x  3 vladodemo csfac  4096 Feb 13 15:34 e8
drwx-----  2 vladodemo csfac     6 Feb 15 14:57 git
drwx--x--x 13 vladodemo csfac   120 Feb 15 14:57 .
$ rsync -av e1/ git/e1/
```

Copy other Directories with rsync: e2

```
$ rsync -av e2/ git/e2/  
sending incremental file list  
created directory git/e2  
./  
index.html
```

I

```
timberlea.cs.dal.ca - PuTTY
index.html~

sent 255 bytes  received 86 bytes  682.00 bytes/sec
total size is 70  speedup is 0.21
$ rsync -av e2/ git/e2/
sending incremental file list
created directory git/e2
./
index.html
index.html~
material/
material/.htaccess
material/MDI-page.pdf

sent 794,495 bytes  received 136 bytes  1,589,262.00 bytes/
sec
total size is 793,957  speedup is 1.00
$ rsync -av e3/ git/e3/
sending incremental file list
created directory git/e3
./
.htaccess
.htaccess~
.htpasswd
index.html
index.html~
pw
material/
material/.htaccess
material/MDI-page.pdf
```



```
$ rsync -av e4/ git/e4/
```

```
$ rsync -av e5/ git/e5/  
sending incremental file list  
created directory git/e5  
./  
.htaccess  
.htaccess~  
.htpasswd  
index.html  
index.html~
```

```
$ rsync -av e6/ git/e6/  
sending incremental file list  
created directory git/e6  
./  
.htaccess  
.htaccess~  
.htpasswd
```

```
$ rsync -av e7/ git/e7/  
sending incremental file list  
created directory git/e7  
./  
.htaccess  
.htaccess~  
.htpasswd
```

```
$ rsync -av e8/ git/e8/  
sending incremental file list  
created directory git/e8  
./  
.htaccess  
htaccess.
```

Use highlighted commands to check your git directory

```
$ ls
e1 e2 e3 e4 e5 e6 e7 e8 git
$ cd git
$ ls
e1 e2 e3 e4 e5 e6 e7 e8
$ ls -l
total 4
drwx--x--x 2 vladodemo csfac 43 Feb 6 13:16 e1
drwx--x--x 3 vladodemo csfac 59 Feb 6 13:27 e2
drwx--x--x 3 vladodemo csfac 121 Feb 8 13:59 e3
drwx--x--x 3 vladodemo csfac 121 Feb 8 13:59 e4
drwx--x--x 3 vladodemo csfac 178 Feb 8 15:26 e5
drwx--x--x 3 vladodemo csfac 209 Feb 8 15:39 e6
drwx--x--x 3 vladodemo csfac 209 Feb 13 14:56 e7
drwx--x--x 3 vladodemo csfac 4096 Feb 13 15:34 e8
$ pwd
<your home directory .....> /public_html/dgin5201/git
```

Follow git Commands for Upload

We now follow git commands shows on the GitLab page
First, let us highlight which ones we will use:

```

Courses > ... > adm > vladodemo2

git clone https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2.git
cd vladodemo2
git switch --create main
touch README.md
git add README.md
git commit -m "add README"
git push --set-upstream origin main

Push an existing folder

cd existing_folder
git init --initial-branch=main
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vla
git add .
git commit -m "Initial commit"
git push --set-upstream origin main

Push an existing Git repository

cd existing_repo
git remote rename origin old-origin
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vla
git push --set-upstream origin --all
git push --set-upstream origin --tags

test.cgi
material/
material/.htaccess
material/MDI-page.pdf

sent 806,456 bytes  received 478 bytes  322,773.60 bytes/se
c
total size is 804,714  speedup is 1.00
$ ls
e1 e2 e3 e4 e5 e6 e7 e8 e8a git save
$ cd git
$ ls
e1 e2 e3 e4 e5 e6 e7 e8
$ ls -l
total 4
drwx--x--x 2 vladodemo csfac  43 Feb  6 13:16 e1
drwx--x--x 3 vladodemo csfac  59 Feb  6 13:27 e2
drwx--x--x 3 vladodemo csfac 121 Feb  8 13:59 e3
drwx--x--x 3 vladodemo csfac 121 Feb  8 13:59 e4
drwx--x--x 3 vladodemo csfac 178 Feb  8 15:26 e5
drwx--x--x 3 vladodemo csfac 209 Feb  8 15:39 e6
drwx--x--x 3 vladodemo csfac 209 Feb 13 14:56 e7
drwx--x--x 3 vladodemo csfac 4096 Feb 13 15:34 e8
$ pwd
/users/faculty/vladodemo/public_html/dgin5201/git

```

Type in Commands as Shown in GitLab page

Carefully copy commands from the GitLab page (these screenshots are from the last year):

Push an existing folder

```
cd existing_folder
git init --initial-branch=main
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vla
git add .
git commit -m "Initial commit"
git push --set-upstream origin main
```

Push an existing Git repository

```
cd existing_repo
git remote rename origin old-origin
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vla
git push --set-upstream origin --all
git push --set-upstream origin --tags
```

```
$ ls -l
total 4
drwx--x--x 2 vladodemo csfac 43 Feb 6 13:16 e1
drwx--x--x 3 vladodemo csfac 59 Feb 6 13:27 e2
drwx--x--x 3 vladodemo csfac 121 Feb 8 13:59 e3
drwx--x--x 3 vladodemo csfac 121 Feb 8 13:59 e4
drwx--x--x 3 vladodemo csfac 178 Feb 8 15:26 e5
drwx--x--x 3 vladodemo csfac 209 Feb 8 15:39 e6
drwx--x--x 3 vladodemo csfac 209 Feb 13 14:56 e7
drwx--x--x 3 vladodemo csfac 4096 Feb 13 15:34 e8
$ pwd
/users/faculty/vladodemo/public_html/dgin5201/git
$ git init --initial-branch=main
Initialized empty Git repository in /users/webhome/vladodem
o/dgin5201/git/.git/
$ git remote add origin https://git.cs.dal.ca/courses/2024-
winter/dgin5201/yourssid
```


Push an existing folder

```
git_folder
--initial-branch=main
> add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2.git

t -n "Initial commit"
--set-upstream origin main
< >
```

Push an existing Git repository

```
cd existing_repo
git remote rename origin old-origin
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vla
git push --set-upstream origin --all
git push --set-upstream origin --tags
< >
```

```
$ ls -l
total 4
drwx--x--x 2 vladodemo csfac  43 Feb  6 13:16 e1
drwx--x--x 3 vladodemo csfac  59 Feb  6 13:27 e2
drwx--x--x 3 vladodemo csfac 121 Feb  8 13:59 e3
drwx--x--x 3 vladodemo csfac 121 Feb  8 13:59 e4
drwx--x--x 3 vladodemo csfac 178 Feb  8 15:26 e5
drwx--x--x 3 vladodemo csfac 209 Feb  8 15:39 e6
drwx--x--x 3 vladodemo csfac 209 Feb 13 14:56 e7
drwx--x--x 3 vladodemo csfac 4096 Feb 13 15:34 e8
$ pwd
/users/faculty/vladodemo/public_html/dgin5201/git
$ git init --initial-branch=main
Initialized empty Git repository in /users/webhome/vladodem
o/dgin5201/git/.git/
$ git remote add origin https://git.cs.dal.ca/courses/2024-
winter/dgin5201/yourcsid.git
```

Push an existing folder

```
cd existing_folder
git init --initial-branch=main
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vla
git add .
git commit -m "Initial commit"
git push --set-upstream origin main
```

Push an existing Git repository

```
cd existing_repo
git remote rename origin old-origin
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vla
git push --set-upstream origin --all
git push --set-upstream origin --tags
```

```
drwx--x--x 3 vladodemo csfac 59 Feb 6 13:27 ez
drwx--x--x 3 vladodemo csfac 121 Feb 8 13:59 e3
drwx--x--x 3 vladodemo csfac 121 Feb 8 13:59 e4
drwx--x--x 3 vladodemo csfac 178 Feb 8 15:26 e5
drwx--x--x 3 vladodemo csfac 209 Feb 8 15:39 e6
drwx--x--x 3 vladodemo csfac 209 Feb 13 14:56 e7
drwx--x--x 3 vladodemo csfac 4096 Feb 13 15:34 e8
$ pwd
/users/faculty/vladodemo/public_html/dgin5201/git
$ git init --initial-branch=main
Initialized empty Git repository in /users/webhome/vladodem
o/dgin5201/git/.git/
$ git remote add origin https://git.cs.dal.ca/courses/2024-
winter/dgin5201/adm/vladodemo2.git
$ git add .
$ git commit -m "Initial commit"
```

```
Courses > ... > adm > vladodemo2

git clone https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2.git
cd vladodemo2
git switch --create main
touch README.md
git add README.md
git commit -m "add README"
git push --set-upstream origin main

Push an existing folder

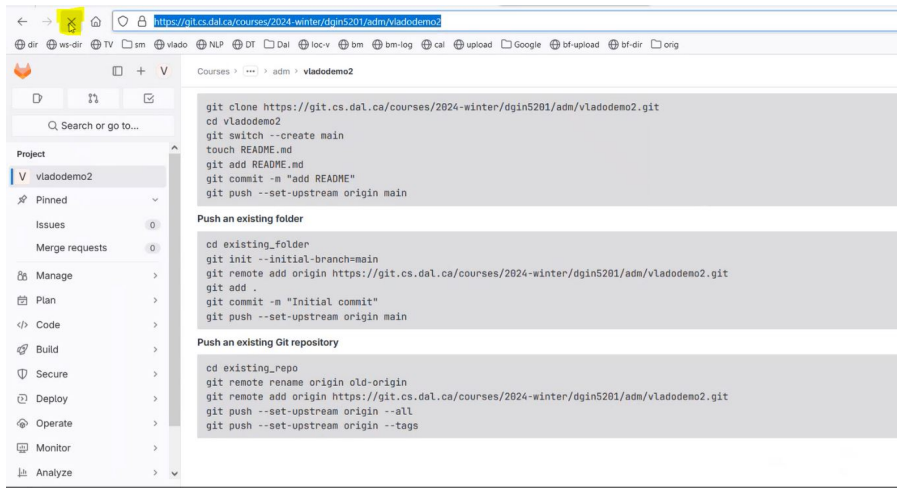
cd existing_folder
git init --initial-branch=main
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vla
git add .
git commit -m "Initial commit"
git push --set-upstream origin main

Push an existing Git repository

cd existing_repo
git remote rename origin old-origin
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vla
git push --set-upstream origin --all
git push --set-upstream origin --tags
```

```
create mode 100644 e8/material/MDI-page.pdf
create mode 100644 e8/pw
create mode 100755 e8/register-py.cgi
create mode 100755 e8/register-py.cgi~
create mode 100755 e8/register.cgi
create mode 100755 e8/register.cgi~
create mode 100755 e8/register.php
create mode 100755 e8/register.py
create mode 100644 e8/registrations-saved.txt
create mode 100755 e8/test.cgi
$ git push --set-upstream origin main
Username for 'https://git.cs.dal.ca': vladodemo
Password for 'https://vladodemo@git.cs.dal.ca':
Enumerating objects: 44, done.
Counting objects: 100% (44/44), done.
Delta compression using up to 32 threads
Compressing objects: 100% (40/40), done.
Writing objects: 100% (44/44), 766.54 KiB | 12.78 MiB/s, do
ne.
Total 44 (delta 21), reused 0 (delta 0), pack-reused 0
To https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/v
ladodemo2.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'or
igin'.
$
```

Refresh GitLab Page to Check Contents



The screenshot shows a web browser window with the URL `https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2`. The page content is organized into three sections, each with a title and a list of terminal commands:

- Project**

```
git clone https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2.git
cd vladodemo2
git switch --create main
touch README.md
git add README.md
git commit -m "add README"
git push --set-upstream origin main
```
- Push an existing folder**

```
cd existing_folder
git init --initial-branch=main
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2.git
git add .
git commit -m "Initial commit"
git push --set-upstream origin main
```
- Push an existing Git repository**

```
cd existing_repo
git remote rename origin old-origin
git remote add origin https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2.git
git push --set-upstream origin --all
git push --set-upstream origin --tags
```

Expected Contents in your GitLab Repository

The screenshot shows a web browser displaying the GitLab interface for a repository named 'vladodemo2'. The URL in the address bar is `https://git.cs.dal.ca/courses/2024-winter/dgin5201/adm/vladodemo2`. The page title is 'Courses > ... > adm > vladodemo2'. The main content area shows an 'Initial commit' by 'Vlado Keselj (Demo)' authorized 1 minute ago, with commit ID 'db48df63'. Below the commit information, there are buttons for 'History', 'Find file', 'Edit', 'Clone', and 'Add README', 'Add LICENSE', 'Add CHANGELOG', 'Add CONTRIBUTING', 'Add Kubernetes cluster', 'Set up CI/CD', and 'Add Wiki'. A table below lists the repository's contents:

Name	Last commit	Last update
📁 e1	Initial commit	1 minute ago
📁 e2	Initial commit	1 minute ago
📁 e3	Initial commit	1 minute ago
📁 e4	Initial commit	1 minute ago
📁 e5	Initial commit	1 minute ago
📁 e6	Initial commit	1 minute ago
📁 e7	Initial commit	1 minute ago

Browser address bar: <https://git.cs.dal.ca/courses/2024-winter/dgjn5201/adm/vladodemo2>

Project: vladodemo2

main vladodemo2 / +


History Find file Edit Clone




Add README Add LICENSE Add CHANGELOG Add CONTRIBUTING Add Kubernetes cluster Set up CI/CD Add Wiki

Configure Integrations

Name	Last commit	Last update
e1	Initial commit	1 minute ago
e2	Initial commit	1 minute ago
e3	Initial commit	1 minute ago
e4	Initial commit	1 minute ago
e5	Initial commit	1 minute ago
e6	Initial commit	1 minute ago
e7	Initial commit	1 minute ago
e8	Initial commit	1 minute ago

Project sidebar: Pinned, Issues (0), Merge requests (0), Manage, Plan, Code, Build, Secure, Deploy, Operate, Monitor, Analyze


+ V

Search or go to...

Project

V vladodemo2

- Pinned
- Issues 0
- Merge requests 0
- Manage
- Plan
- Code
- Build
- Secure
- Deploy
- Operate
- Monitor
- Analyze

Courses > ... > adm > vladodemo2

main vladodemo2 / e8 / +

Lock History Find file Edit ↓ ↓ Clone ↓

Name	Last commit	Last update
..		
material	Initial commit	1 minute ago
.htaccess	Initial commit	1 minute ago
.htaccess~	Initial commit	1 minute ago
.htpasswd	Initial commit	1 minute ago
index-php.html	Initial commit	1 minute ago
index-php.html~	Initial commit	1 minute ago
index-py.html	Initial commit	1 minute ago
index-py.html~	Initial commit	1 minute ago
index-py2.html	Initial commit	1 minute ago
index-py2.html~	Initial commit	1 minute ago

Lab 4 Finished

- With this you have finished Lab 4 and Assignment 1