Aim: Practice Source code management on GitHub

Description:

GitHub helps the company's long-standing efforts to accelerate development by breaking down communication barriers, shortening feedback loops, and automating tasks wherever possible.

What is DevOps?

DevOps is a software development and IT operations process that helps to improve the collaboration between developers and operators by automating the Deployment, configuration, monitoring, and management of applications.

Benefits of DevOps

1. **Faster Delivery:** DevOps enables organizations to release new products and updates faster and more frequently, which can lead to a competitive advantage.

2. **Improved Collaboration:** DevOps promotes collaboration between development and operations teams, resulting in better communication, increased efficiency, and reduced friction.

3. **Improved Quality:** DevOps emphasizes automated testing and continuous integration, which helps to catch bugs early in the development process and improve the overall quality of software.

4. **Increased Automation:** DevOps enables organizations to automate many manual processes, freeing up time for more strategic work and reducing the risk of human error.

5. **Better Scalability:** DevOps enables organizations to quickly and efficiently scale their infrastructure to meet changing demands, improving the ability to respond to business needs.

6. **Increased Customer Satisfaction:** DevOps helps organizations to deliver new features and updates more quickly, which can result in increased customer satisfaction and loyalty.

7. **Improved Security:** DevOps promotes security best practices, such as continuous testing and monitoring, which can help to reduce the risk of security breaches and improve the overall security of an organization's systems.

8. **Better Resource Utilization:** DevOps enables organizations to optimize their use of resources, including hardware, software, and personnel, which can result in cost savings and improved efficiency.

Implementation:

Step 1: Sign-in in GitHub Account:

Fine-grained personal access tokens (Preview)   Generate new token

These are fine-grained, repository-scoped tokens suitable for personal API use and for using Git over HTTPS.

Make sure to copy your personal access token now as you will not be able to see this again.   Never used   Delete

github_pat_11A5VTKSY0vYLEvRpYsMHG_565erc7AsfQXfGjhpWGT8D: ⧉

Expires **on Wed, Feb 26 2025**.

© 2025 GitHub, Inc.   Terms   Privacy   Security   Status   Docs   Contact   Manage cookies   Do not share my personal information

github_pat_11A5VTKSY0vYLEvRpYsMHG_565erc7AsfQXfGjhpWGT8Ditjvo5fNgA0Cqh YRRkEAaSANQFQ5Ter3JGlLg

Step 1A: Git Bash Sign-in

Commands:

git config --global user.name "Your Name"

git config --global user.email "your_email@example.com"

Step 2: Clone the Repository to Your Local Machine

Command: git clone <repository-URL>

```
geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops
$ git clone https://github.com/geeta-seshapalli/DevOpsLab.git
Cloning into 'DevOpsLab'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (5/5), done.

geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops
$
```

Step 3: Move to the Repository Directory

Command: cd repository-name

```
geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops
$ cd DevopsLab

geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$
```

Step 4: Create a New File in the Repository and Add the Source Code

Created BMI Calculator

📁 DesiQnA   📁 Profile   📁 Fin   🌐 ChatGPT   📁 Java   📁 Interview   📁 S2

BMI Calculator

Weight (kg):

55

Height (cm):

156

Calculate BMI

Your BMI is **22.60** (**Normal weight**).

Step 5: Stage the Changes

Command: git add filename.extension

```
geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$ git add bmi.html

geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$
```

To stage all changes, use: git add .

```
geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$ git add .

geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$
```

Step 6: Commit the Changes

Commit the staged changes with a meaningful message:

Command: git commit -m "initial source code"

```
geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$ git commit -m "initial source code"
[main edc4546] initial source code
 1 file changed, 118 insertions(+)
 create mode 100644 bmi.html

geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$
```
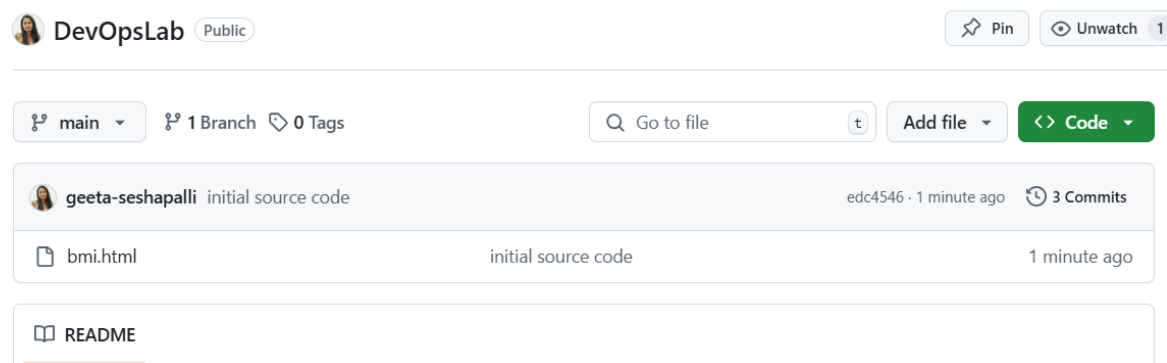
Step 7: Push the Changes to the Remote Repository

Command: git push origin main

```
geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$ git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 1.29 KiB | 1.29 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/geeta-seshapalli/DevOpsLab.git
   2e875ba..edc4546  main -> main

geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$ |
```

Changes are invisible in GitHub Repository.

DevOpsLab Public   📌 Pin   👁 Unwatch 1

⑁ main ▾   ⑁ 1 Branch ◯ 0 Tags   🔍 Go to file   t   Add file ▾   <> Code ▾

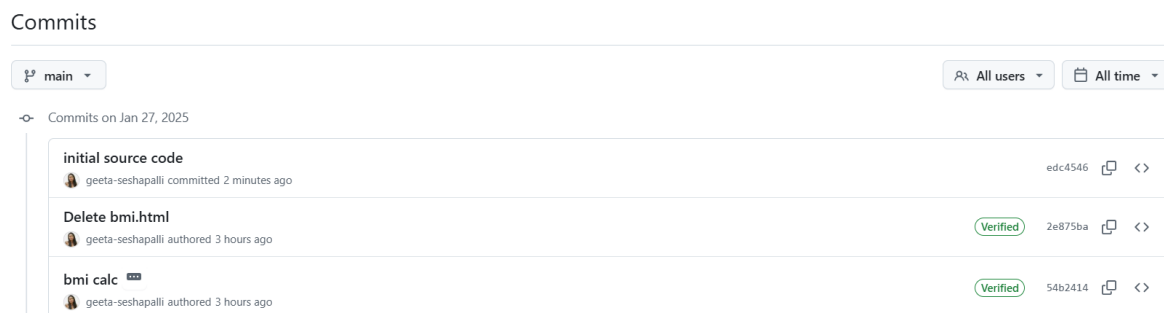👤 geeta-seshapalli initial source code   edc4546 · 1 minute ago   🕐 3 Commits

📄 bmi.html   initial source code   1 minute ago

📖 README

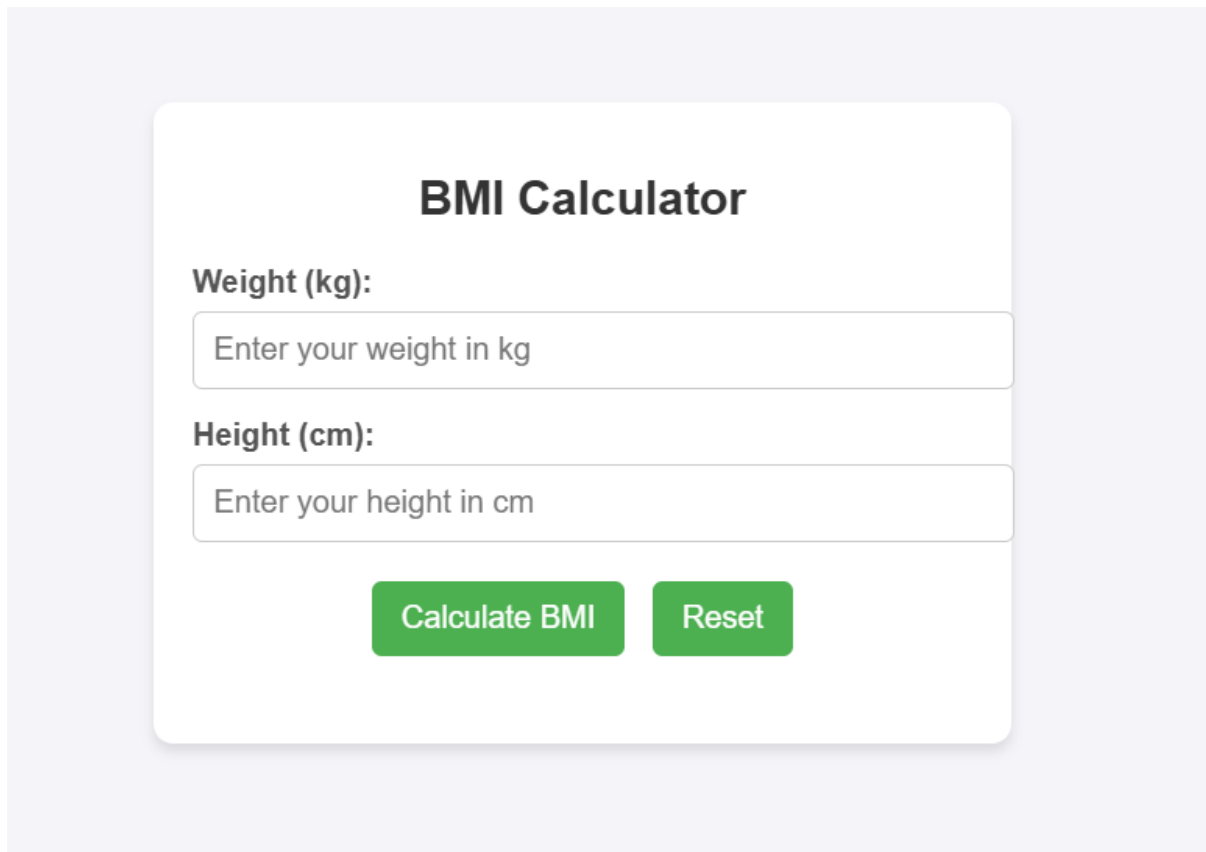*Commits are visible in GitHub Repository*

Commits

⑁ main ▾   👥 All users ▾   📅 All time ▾

-○- Commits on Jan 27, 2025

    initial source code   edc4546 📋 <>
    👤 geeta-seshapalli committed 2 minutes ago

    Delete bmi.html   Verified   2e875ba 📋 <>
    👤 geeta-seshapalli authored 3 hours ago

    bmi calc 💀   Verified   54b2414 📋 <>
    👤 geeta-seshapalli authored 3 hours ago

New Changes made in existing code locally, added reset button

Step 8: Check the Status of Your Repository

Command: git status

This will show a list of files that have been modified or are untracked.

Step 9: Stage the Changes

Command: git add filename.extension

```
geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$ git add bmi.html

geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$ git add .
```

Step 10: Commit the Changes

Command: git commit -m "Describe the changes made"

```
geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$ git commit -m "reset button added"
[main f81215e] reset button added
 1 file changed, 8 insertions(+)
```

Step 11: Push the Changes to GitHub

Command: git push origin main

```
geeta@Geeta MINGW64 ~/OneDrive/Desktop/Geeta/vjti/Sem2/Devops/DevopsLab (main)
$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 394 bytes | 394.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/geeta-seshapalli/DevOpsLab.git
   edc4546..f81215e  main -> main
```

Changes are visible in GitHub Repository

Commits

⑂ main ▾   ⚇ All users ▾   📅 All time ▾

-○-   Commits on Jan 27, 2025

reset button added
geeta-seshapalli committed 1 minute ago   f81215e ⎘ <>

initial source code
geeta-seshapalli committed 8 minutes ago   edc4546 ⎘ <>

Delete bmi.html
geeta-seshapalli authored 3 hours ago   Verified   2e875ba ⎘ <>

bmi calc 🏳️
geeta-seshapalli authored 3 hours ago   Verified   54b2414 ⎘ <>

👩 **DevOpsLab** (Public)   📌 Pin   👁 Unwatch 1

⑂ main ▾   ⑂ 1 Branch   ⎙ 0 Tags   🔍 Go to file   t   Add file ▾   <> Code ▾

👩 geeta-seshapalli reset button added   f81215e · 1 minute ago   🕐 4 Commits

📄 bmi.html   reset button added   1 minute ago

Conclusion:

This experiment demonstrated the essential workflow for managing and versioning source code using Git and GitHub. Here's the summarized conclusion:

1. **Repository Management**:
   Successfully created a new repository on GitHub and linked it to the local machine using Git.

2. **Version Control Workflow**:

   o **Cloning**: Retrieved the remote repository to the local system.

   o **Staging and Committing**: Staged and committed changes, ensuring a clear and trackable development process.

   o **Pushing Changes**: Synced local updates with the remote repository, making changes available to collaborators.

3. **Efficient Collaboration**:
   Git and GitHub provide a seamless way to collaborate on projects, manage code versions, and maintain an organized history of changes.

4. **Practical Understanding**:
   Gained hands-on experience with key Git commands (git clone, git add, git commit, git push), fostering a deeper understanding of version control best practices.

Key Takeaways:

- Git simplifies tracking changes in code and ensures no progress is lost.

- GitHub acts as a reliable platform for remote storage, collaboration, and backup.

- Writing meaningful commit messages improves project documentation and debugging.

Repository Link: https://github.com/geeta-seshapalli/DevOpsLab