

1. Create a new GitHub repository.

- Clone the repository to your local machine using SSH (generate an SSH key if needed, add the public key to your GitHub account).
- Create a new branch named after your username (e.g., Tutedude).
- Add your Flask project files to this branch.

Commit the changes and merge the branch into the main branch.

```
MINGW64:/c:/Users/Kunal Sandeep Jadhav/onedrive/desktop/devops/git_tutorial/Git_task_1

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_task_1 (Tutedude)
$ git add .
error: 'flask_api_app/' does not have a commit checked out
fatal: adding files failed

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_task_1 (Tutedude)
$ cd flask_api_app

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_task_1/flask_api_app (master)
$ rm -rf .git

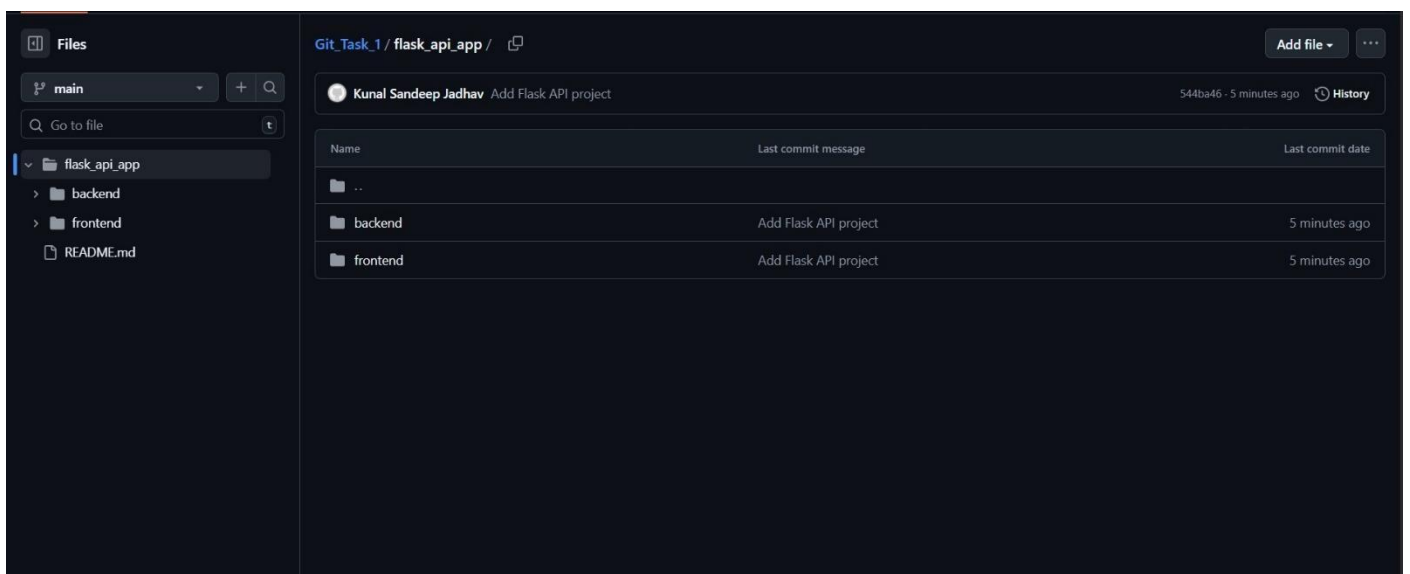
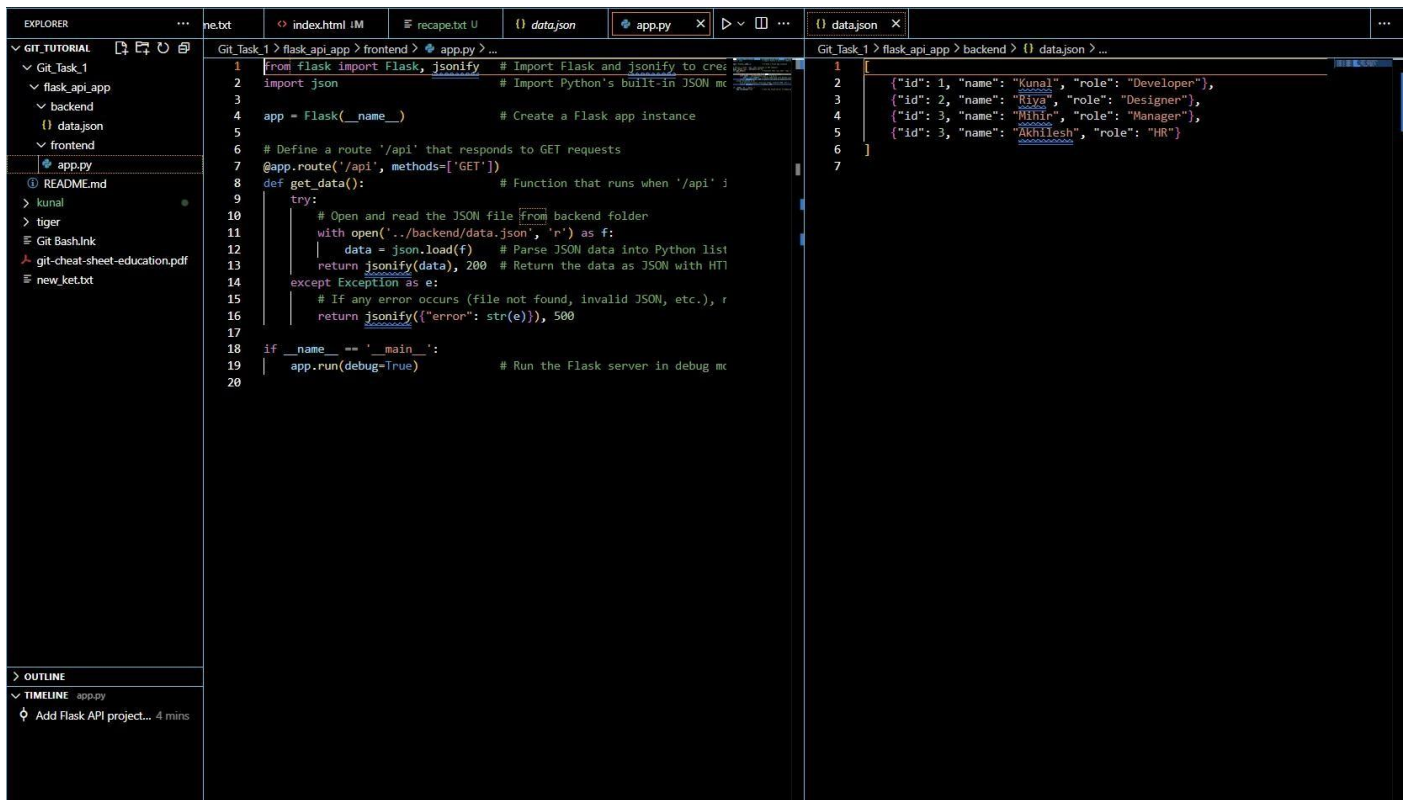
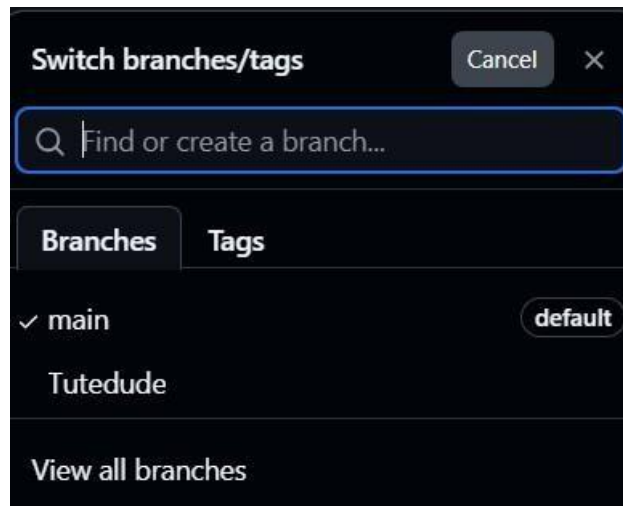
Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_task_1/flask_api_app (Tutedude)
$ cd ..

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_task_1 (Tutedude)
$ git add .

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_task_1 (Tutedude)
$ git commit -m "Add Flask API project"
[Tutedude 544ba46] Add Flask API project
 2 files changed, 25 insertions(+)
 create mode 100644 flask_api_app/backend/data.json
 create mode 100644 flask_api_app/frontend/app.py

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_task_1 (Tutedude)
$ git push origin Tutedude
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 12 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (7/7), 1.06 KiB | 271.00 KiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'Tutedude' on GitHub by visiting
remote:   https://github.com/KUNALKJ28/Git_Task_1/pull/new/Tutedude
remote:
To github.com:KUNALKJ28/Git_Task_1.git
 * [new branch]      Tutedude -> Tutedude

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_task_1 (Tutedude)
$ |
```



2. Create a new branch named <your_name>_new (e.g., Tutedude_new).

- . Update the content of the JSON file used for the / api route in this branch.
- . Merge the <your_name>_new branch into the main branch.
- . If there are conflicts during the merge, resolve them by accepting the changes from the <your_name>_new branch.
- . Add the resolved changes to the staging area, commit them, and push the updates to the remote repository.

```

MINGW64/c:/Users/Kunal Sandeep Jadhav/onedrive/desktop/devops/git_tutorial/Git_Task_1
Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main|MERGING)
$ git commit -m "Merge Branch 'Tutedude_new' and resolve conflicts"
[main fcf4c25] Merge Branch 'Tutedude_new' and resolve conflicts

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main)
$ git push
To github.com:KUNALKJ28/Git_Task_1.git
 ! [rejected]        main -> main (non-fast-forward)
error: failed to push some refs to 'github.com:KUNALKJ28/Git_Task_1.git'
hint: Updates were rejected because the tip of your current branch
hint: is behind
hint: its remote counterpart. If you want to integrate the remote c
hint: hanges,
hint: use 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for d
hint: etails.

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main)
$ git pull
Auto-merging flask_api_app/backend/data.json
CONFLICT (add/add): Merge conflict in flask_api_app/backend/data.json
Automatic merge failed; fix conflicts and then commit the result.

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main|MERGING)
$ git push
To github.com:KUNALKJ28/Git_Task_1.git
 ! [rejected]        main -> main (non-fast-forward)
error: failed to push some refs to 'github.com:KUNALKJ28/Git_Task_1.git'
hint: Updates were rejected because the tip of your current branch is behind
hint: its remote counterpart. If you want to integrate the remote changes,
hint: use 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

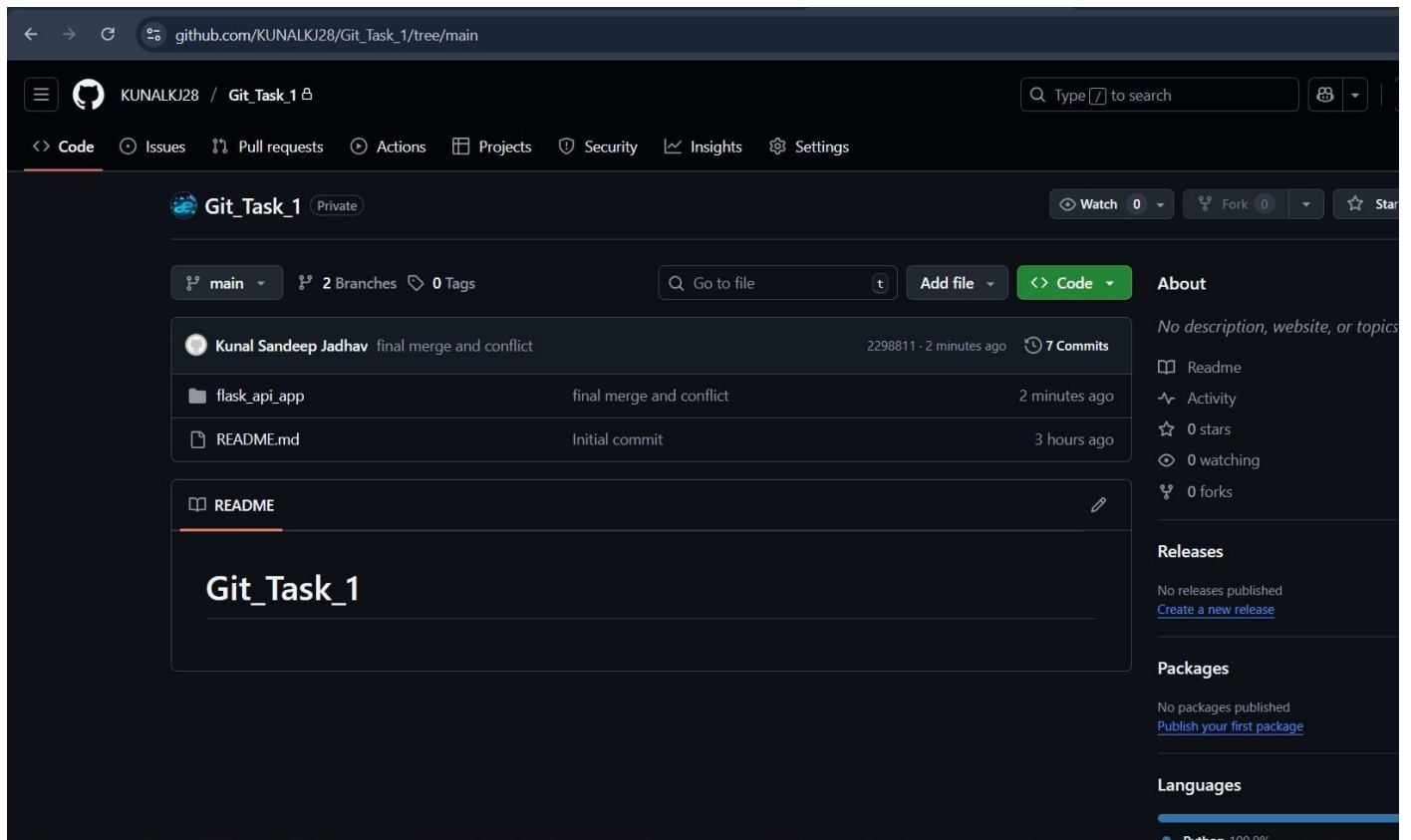
Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main|MERGING)
$ git add .

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main|MERGING)
$ git commit -m "final merge and conflict"
[main 2298811] final merge and conflict

Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main)
$ git push
Enumerating objects: 25, done.
Counting objects: 100% (25/25), done.
Delta compression using up to 12 threads
Compressing objects: 100% (13/13), done.
Writing objects: 100% (20/20), 1.74 KiB | 593.00 KiB/s, done.
Total 20 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), done.
To github.com:KUNALKJ28/Git_Task_1.git
   b24f1bb..2298811  main -> main

```





3. Branch Creation:

- . Create two branches: master_1 and master_2 from the main branch.
- . Feature Development in master 1:
 - . In the master_1 branch, create a To-Do Page in the frontend.
 - o The page should contain a form with the following fields:

Item Name . Item

Description
- . Backend API in master 2:
 - . In the master 2 branch, create a backend route named / submittodoitem.
 - . This route will:
 - o Accept itemName and itemDescription via a POST request.
 - Store these details in a MongoDB database.
- . Merging Changes:
 - . Merge the changes from both master_1 and master_2 into the main branch.

```

MINGW64:/c:/Users/Kunal Sadeep Jadhav/onedrive/devops/git_tutorial/Git_Task_1
Kunal Sadeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop
$ cd devops

Kunal Sadeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops
$ cd git_tutorial

Kunal Sadeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial
$ cd Git_Task_1

Kunal Sadeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (master_2)
$ git switch main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

Kunal Sadeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main)
$ git merge master_1
Updating 2298811..36f16f3
Fast-forward
 flask_api_app/frontend/app.py | 6 +++++
 flask_api_app/frontend/template/todo.html | 26 ++++++++++++++++++++++
 2 files changed, 31 insertions(+), 1 deletion(-)
 create mode 100644 flask_api_app/frontend/template/todo.html

Kunal Sadeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main)
$ git merge master_2
Already up to date.

Kunal Sadeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main)
$ git add .

Kunal Sadeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main)
$ git commit -m "MERGE: Integrate frontend and backend features"
[main c674d4c] MERGE: Integrate frontend and backend features
 1 file changed, 32 insertions(+)
 create mode 100644 flask_api_app/backend/app.py

Kunal Sadeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main)
$ git push
Enumerating objects: 17, done.
Counting objects: 100% (17/17), done.
Delta compression using up to 12 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (12/12), 1.95 KiB | 284.00 KiB/s, done.
Total 12 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:KUNALKJ28/Git_Task_1.git
 2298811..c674d4c main -> main

Kunal Sadeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1 (main)
$ |

```


Create a New To-Do Item

Item Name:

Item Description:

Add Item

```
Git_Task_1 > flask_api_app > backend > app.py > submit_todo_item
1 # =====
2 #     1. IMPORTS
3 # =====
4 from flask import Flask, jsonify, render_template, request, redirect, url_for
5 from pymongo import MongoClient
6 from dotenv import load_dotenv
7 import os
8 import json
9
10 # =====
11 #     2. APP & DATABASE SETUP
12 # =====
13 app = Flask(__name__, template_folder='../frontend/templates')
14
15 # Load environment variables
16 load_dotenv()
17 MONGO_URI = os.getenv("MONGO_URI")
18
19 # MongoDB Connection
20 client = MongoClient(MONGO_URI)
21 db = client['todo_database']
22 collection = db['items']
23
24 # =====
25 #     3. ROUTES
26 # =====
27 @app.route('/')
28 def home():
29     return redirect(url_for('todo_page'))
30
31 @app.route('/todo')
32 def todo_page():
33     return render_template('todo.html')
34
35 @app.route('/submittodoitem', methods=['POST'])
36 def submit_todo_item():
37     try:
38         itemName = request.form['itemName']
39         itemDescription = request.form['itemDescription']
40
41         todo_item = {
42             "name": itemName,
43             "description": itemDescription
44         }
45
```

```

45 |
46 |         collection.insert_one(todo_item)
47 |         return redirect(url_for('todo_page'))
48 |     except Exception as e:
49 |         return jsonify({"error": str(e)}), 500
50 |
51 | @app.route('/api', methods=['GET'])
52 | def get_data():
53 |     try:
54 |         with open('data.json', 'r') as f:
55 |             data = json.load(f)
56 |             return jsonify(data), 200
57 |     except Exception as e:
58 |         return jsonify({"error": str(e)}), 500
59 |
60 | # =====
61 | #         4. RUN THE SERVER
62 | # =====
63 | if __name__ == '__main__':
64 |     app.run(debug=True, use_reloader=False)

```

cloud.mongodb.com/v2/68d8003a83d84b6ea30f5e80#/explorer/68d8015f06a42d7cdb43632c/todo_database/items/find

Atlas Kunal's Org Access Manager Billing All Clusters Get Help Kunal

Project 0 Data Services Charts

Overview

DATABASE

Clusters

Data Explorer

SERVICES

Atlas Search

Stream Processing

Triggers

Migration

Data Federation

SECURITY

Quickstart

Backup

Database Access

Network Access

Advanced

Goto

Data Explorer PREVIEW

CLUSTERS (1)

Search clusters

kunal

- admin
- local
- sample_mflix
- test
- todo_database
 - Items

Items

kunal > todo_database > Items

Documents 0 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

EXPLAIN RESET FIND Options

ADD DATA UPDATE DELETE

25 1 - 2 of 2

```

_id: ObjectId('68dff94bf739fc9c407d9f3')
name: "Book "
description: "i want to buy it!"

```

```

_id: ObjectId('68dff9b9bf739fc9c407d9f4')
name: "bike"
description: "today evening i want to go to the ride "

```

System Status: All Good

©2025 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

```
Git_Task_1 > flask_api_app > frontend > templates > < todo.html > html > body > form > label
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>New To-Do Item</title>
6   <style>
7     body { font-family: sans-serif; margin: 2em; }
8     form { display: flex; flex-direction: column; width: 300px; }
9     label { margin-top: 1em; }
10    input, textarea { padding: 0.5em; margin-top: 0.5em; }
11    button { margin-top: 1.5em; padding: 0.7em; cursor: pointer; }
12  </style>
13 </head>
14 <body>
15   <h1>Create a New To-Do Item</h1>
16   <form action="/submittodoitem" method="POST">
17     <label for="itemName">Item Name:</label>
18     <input type="text" id="itemName" name="itemName" required>
19
20     <label for="itemDescription">Item Description:</label>
21     <textarea id="itemDescription" name="itemDescription" rows="4" required></textarea>
22
23     <button type="submit">Add Item</button>
24   </form>
25 </body>
26 </html>
```

4. Enhancing the To-Do Form in master_1:

- In the master_1 branch, add the following fields to the To-Do form:
 - **Item ID** ○ **Item**
UUID
 - **Item Hash**
- **Committing in Sequence:**
- Add and commit each field separately in the following order:
 - **First commit:** Add **Item ID** field.
 - **Second commit:** Add **Item UUID** field.
 - **Third commit:** Add **Item Hash** field.
- **Merging to main:**
- Merge the master_1 branch into the main branch.
- **Git Reset and Commit Deletion:**
- In the main branch, use **Git Reset** to roll back to the commit where only the **Item ID** field was added.
- Use `git reset --soft` to ensure changes remain staged.
- Re-commit this state to the main branch.
- Merge this updated state to the main branch.

- **Rebasing Changes:**
- **Rebase the updated changes in the main branch to the master_1 branch. Clarification:**
 - During rebasing, **preserve individual commits** to maintain the commit history for each change (i.e., do not squash commits).
 - Use git rebase main master_1 to integrate changes from the main branch back into the master_1 branch.

MINGW64:/c:/Users/Kunal Sandeep Jadhav/onedrive/desktop/devops/git_tutorial/Git_Task_1/flask_api_app

```
Kunal Sandeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1/flask_api_app (main)
$ git checkout -b master_1
```

fatal: a branch named 'master_1' already exists

```
Kunal Sandeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1/flask_api_app (main)
$ git add path/to/yourfile.html
git commit -m "Add Item ID field to To-Do form"
git push origin master_1
warning: could not open directory 'flask_api_app/path/to/': No such file or directory
fatal: pathspec 'path/to/yourfile.html' did not match any files
On branch main
Your branch is up to date with 'origin/main'.
```

```
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   frontend/templates/todo.html
```

```
no changes added to commit (use "git add" and/or "git commit -a")
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'master_1' on GitHub by visiting:
remote:   https://github.com/KUNALKJ28/Git_Task_1/pull/new/master_1
remote:
To github.com:KUNALKJ28/Git_Task_1.git
 * [new branch]      master_1 -> master_1
```

```
Kunal Sandeep Jadhav@LAPTOP-0EHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1/flask_api_app (main)
$ git add path/to/yourfile.html
git commit -m "Add Item UUID field to To-Do form"
git push origin master_1
warning: could not open directory 'flask_api_app/path/to/': No such file or directory
fatal: pathspec 'path/to/yourfile.html' did not match any files
On branch main
Your branch is up to date with 'origin/main'.
```

```
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   frontend/templates/todo.html
```

```
no changes added to commit (use "git add" and/or "git commit -a")
Everything up-to-date
```

```
Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1/flask_api_app (main)
$ git add path/to/yourfile.html
git commit -m "Add Item Hash field to To-Do form"
git push origin master_1
warning: could not open directory 'flask_api_app/path/to/': No such file or directory
fatal: pathspec 'path/to/yourfile.html' did not match any files
On branch main
Your branch is up to date with 'origin/main'.
```

```
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   frontend/templates/todo.html
```

```
no changes added to commit (use "git add" and/or "git commit -a")
Everything up-to-date
```

```
Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1/flask_api_app (main)
$ git log --oneline --decorate -n 3
a5f481a (HEAD -> main, origin/master_1, origin/main, origin/HEAD, master_1) FEAT: Add Item UUID and Hash fields
278fea0 FEAT: Add Item ID field
36f16f3 FEAT: Add frontend To-DO page
```

```
Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1/flask_api_app (main)
$ git log --oneline --graph --all
* a5f481a (HEAD -> main, origin/master_1, origin/main, origin/HEAD, master_1) FEAT: Add Item UUID and Hash fields
* 278fea0 FEAT: Add Item ID field
* 36f16f3 FEAT: Add frontend To-DO page
* 2298811 (master_2) final merge and conflict
```

```

\
 * b24f1bb Merge pull request #1 from KUNALKJ28/Tutedude
 * 544ba46 (origin/Tutedude, Tutedude) Add Flask API project
 * fcf4c25 Merge Branch 'Tutedude_new' and resolve conflicts
\
 * 4be6851 (Tutedude_new) Update Json
\
 * 2c14ecc upadte Json
\
 * 0aa4d40 Initial commit
```

```
Kunal Sandeep Jadhav@LAPTOP-OEHUN47H MINGW64 ~/onedrive/desktop/devops/git_tutorial/Git_Task_1/flask_api_app (main)
$ |
```

Git_Task_1 > flask_api_app > frontend > templates > todo.html > html > body > form

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>New To-Do Item</title>
6      <style>
7          body { font-family: sans-serif; margin: 2em; }
8          form { display: flex; flex-direction: column; width: 300px; }
9          label { margin-top: 1em; }
10         input, textarea { padding: 0.5em; margin-top: 0.5em; }
11         button { margin-top: 1.5em; padding: 0.7em; cursor: pointer; }
12     </style>
13 </head>
14 <body>
15     <h1>Create a New To-Do Item</h1>
16     <form action="/submittodoitem" method="POST">
17         <label for="itemId">Item ID:</label>
18         <input type="text" id="itemId" name="itemId" required>
19
20         <label for="itemUUID">Item UUID:</label>
21         <input type="text" id="itemUUID" name="itemUUID" required>
22
23         <label for="itemHash">Item Hash:</label>
24         <input type="text" id="itemHash" name="itemHash" required>
25
26         <label for="itemName">Item Name:</label>
27         <input type="text" id="itemName" name="itemName" required>
28
29         <label for="itemDescription">Item Description:</label>
30         <textarea id="itemDescription" name="itemDescription" rows="4" required></textarea>
31
32
33         <button type="submit">Add Item</button>
34     </form>
35 </body>
36 </html>
```

DEVOPS Lecture Page

Git and github - Google Docs

KUNALKJ28/Git_Task_1 at masl

github.com/KUNALKJ28/Git_Task_1/tree/master_1

KUNALKJ28 / Git_Task_1

Type to search

Code

Issues

Pull requests

Actions

Projects

Security

Insights

Settings

Git_Task_1

Private

Watch 0

Fork 0

Star 0

master_1

3 Branches

0 Tags

Go to file

Add file

Code

About

This branch is up to date with main .

Contribute

Kunal Sandeep Jadhav

FEAT: Add Item UUID and Hash fields

a5f481a · 2 days ago

10 Commits

flask_api_app

FEAT: Add Item UUID and Hash fields

2 days ago

README.md

Initial commit

3 days ago

README

Git_Task_1

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Languages

Dart 63.4%

HTML 36.6%

KUNALKJ28 / Git_Task_1

Type to search

Code

Issues

Pull requests

Actions

Projects

Security

Insights

Settings

Commits

master_1

All users

All time

Commits on Oct 4, 2025

FEAT: Add Item UUID and Hash fields

Kunal Sandeep Jadhav committed 2 days ago

a5f481a

FEAT: Add Item ID field

Kunal Sandeep Jadhav committed 2 days ago

278fea0

Commits on Oct 2, 2025

FEAT:Add frontend To-DO page

Kunal Sandeep Jadhav committed 3 days ago

36f16f3

final merge and conflict

Kunal Sandeep Jadhav committed 3 days ago

2298811

Merge Branch 'Tutedude_new'and resolve conflicts

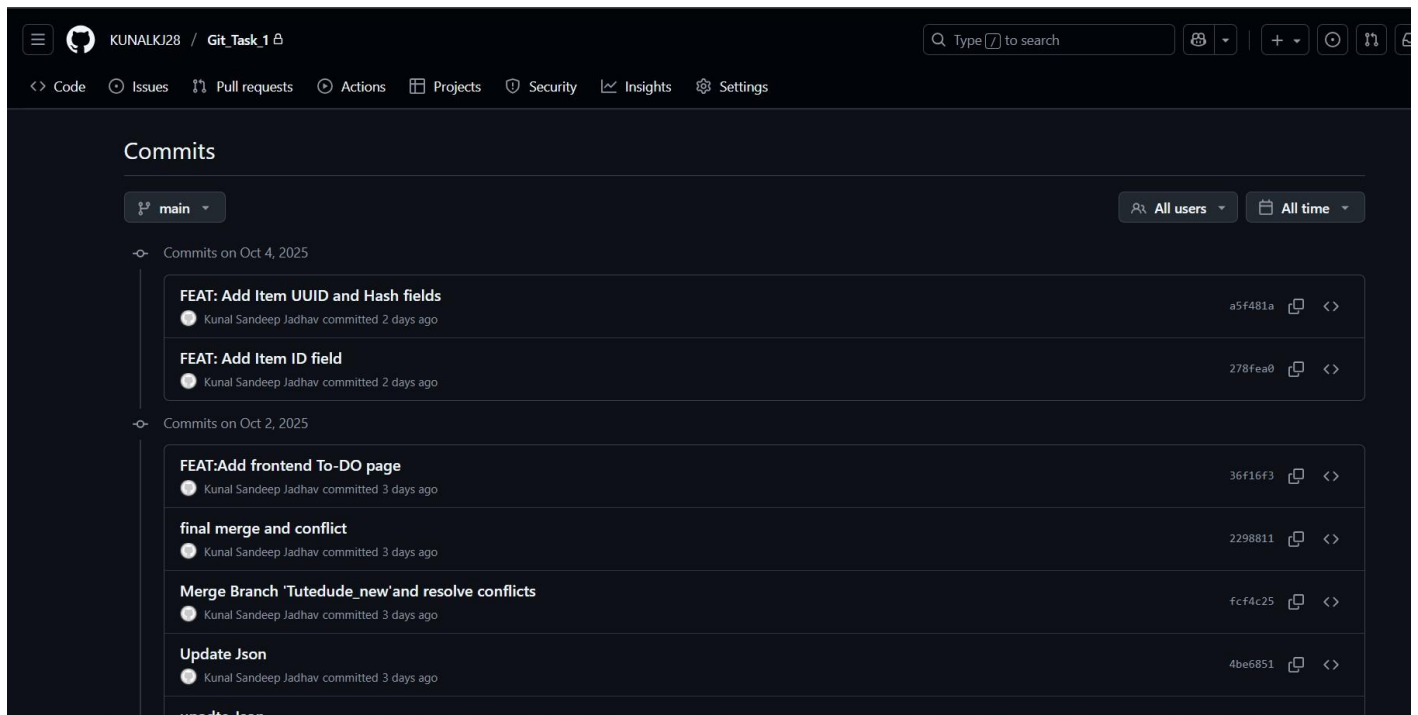
Kunal Sandeep Jadhav committed 3 days ago

fcf4c25

Update Json

Kunal Sandeep Jadhav committed 3 days ago

4be6851



Task GitHub repo - https://github.com/KUNALKJ28/Git_Task_1

I had initially added all the To-Do form fields in a single commit. I have now sorted them properly into separate commits for each field (Item ID, Item UUID, and Item Hash) as per the task requirements. The Git history has been corrected, and the branches (main and master_1) now reflect the intended structure