

LAB 10 GIT ACTIONS

- >Deployment of docker full stack project using git actions.
- >git actions used for automatic deployment of image when there is change in the existing code no need of manual build of images again
- >Cloning existing repository
- >dir (to check directories)
- >cd GIT_ACTIONS-DOCKER-FULLSTACK (changing dir)

The screenshot shows a Windows Command Prompt window titled 'C:\Windows\System32\cmd.exe'. The command 'git clone https://github.com/suryakiran1993/GIT_ACTIONS-DOCKER-FULLSTACK.git' is run, cloning the repository into a directory named 'GIT_ACTIONS-DOCKER-FULLSTACK'. After cloning, the user navigates to this directory using 'cd GIT_ACTIONS-DOCKER-FULLSTACK'. The window also shows a file list for the directory, including files from September 18, 2025, and a total size of 179,617,148,928 bytes free. The taskbar at the bottom includes icons for File Explorer, Task View, and several pinned applications like Microsoft Edge, File Explorer, and others. The system tray shows the date as 18-09-2025 and the time as 11:32.

```
C:\Windows\System32\cmd.exe + - Microsoft Windows [Version 10.0-26100.6584]
(c) Microsoft Corporation. All rights reserved.

D:\3-1 CI CD DEVEOPS\DOCKER>git clone https://github.com/suryakiran1993/GIT_ACTIONS-DOCKER-FULLSTACK.git
Cloning into 'GIT_ACTIONS-DOCKER-FULLSTACK'...
remote: Enumerating objects: 95, done.
remote: Counting objects: 100% (95/95), done.
remote: Compressing objects: 100% (73/73), done.
remote: Total 95 (delta 21), reused 66 (delta 7), pack-reused 0 (from 0)
Receiving objects: 100% (95/95), 52.96 KiB | 2.04 MiB/s, done.
Resolving deltas: 100% (21/21), done.

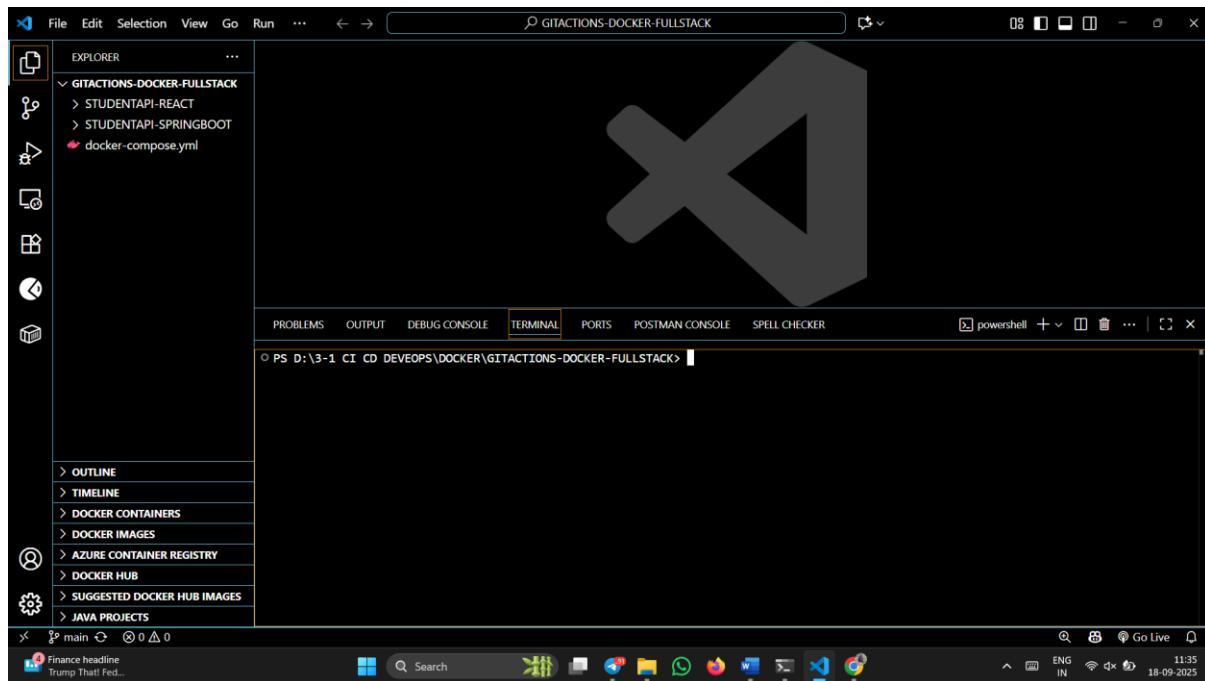
D:\3-1 CI CD DEVEOPS\DOCKER>dir
Volume in drive D is New Volume
Volume Serial Number is A822-90B5

Directory of D:\3-1 CI CD DEVEOPS\DOCKER

18-09-2025 11:29    <DIR>          .
10-09-2025 10:41    <DIR>          ..
13-09-2025 09:41    <DIR>          DOCKER-FULLSTACK
18-09-2025 11:29    <DIR>          GIT_ACTIONS-DOCKER-FULLSTACK
        0 File(s)           0 bytes
        4 Dir(s)   179,617,148,928 bytes free

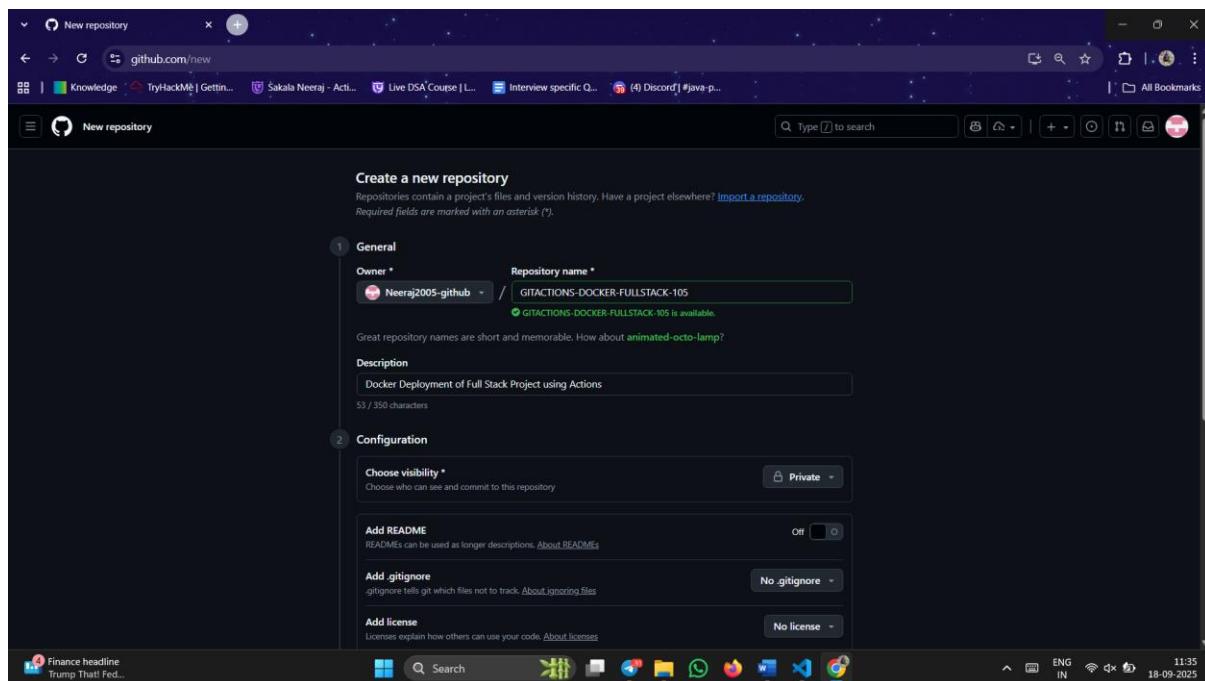
D:\3-1 CI CD DEVEOPS\DOCKER>cd GIT_ACTIONS-DOCKER-FULLSTACK
D:\3-1 CI CD DEVEOPS\DOCKER\GIT_ACTIONS-DOCKER-FULLSTACK>
```

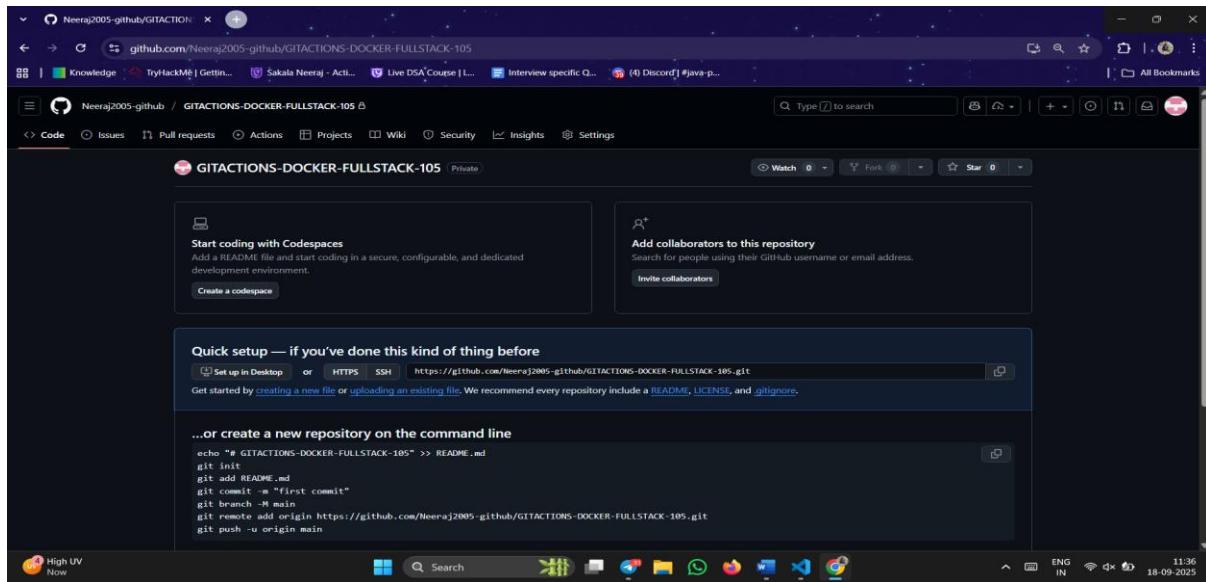
->project opened in vscode containing folders frontend with frontend.Dockerfile, backend with backend.Dockerfile and docker-compose.yml file



->Created a git repository name GITACtIONS-DOCKER-FULLSTACK-105

->visibility public





->Add the project from local to my git repository

->git remote remove origin (remove old origin)

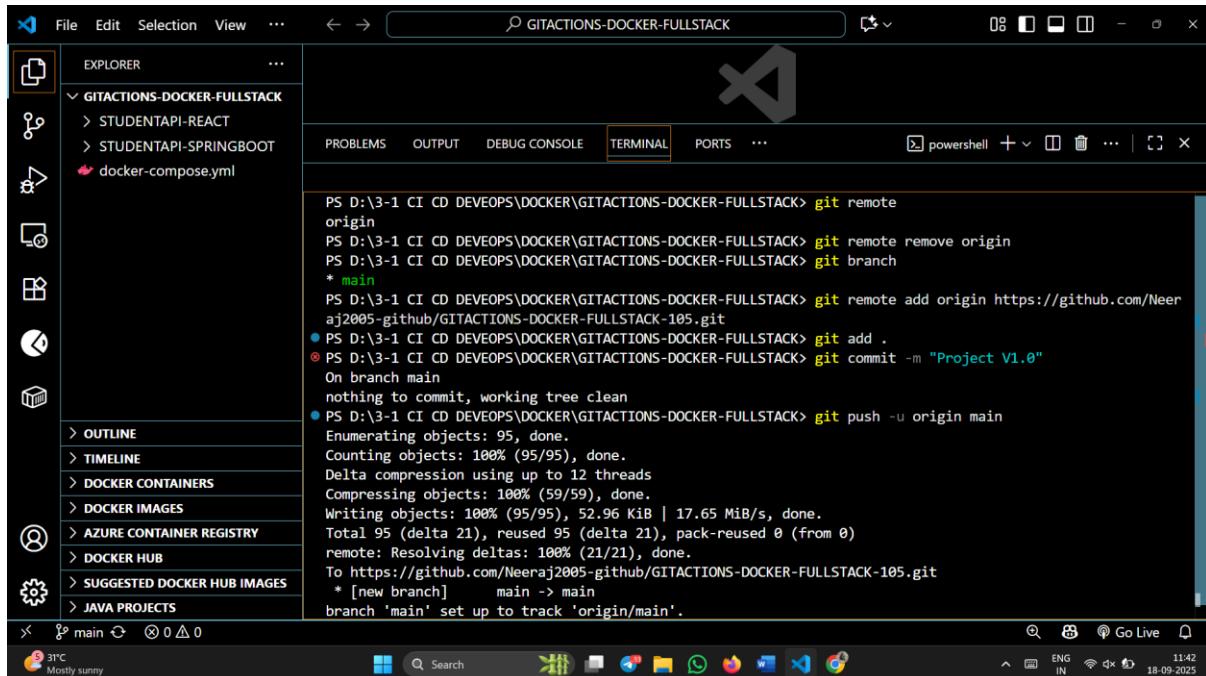
-> git add . (add files to staging area)

->git commit -m "" (commit message)

->git remote add origin ""(add to origin) and make sure to be in main branch (git branch)

->git branch -m main

->git push -u origin main (git push)



->github repo with project

The screenshot shows a GitHub repository page for 'GITACTIONS-DOCKER-FULLSTACK-105'. The repository has 1 branch and 0 tags. There are 11 commits in total, with the most recent being a update to docker-compose.yml by 'suryakiran1993' 2 days ago. Another commit by 'STUDENTAPI-REACT' updated application.properties 3 days ago. A third commit by 'STUDENTAPI-REACT' updated docker-compose.yml 2 days ago. The 'About' section describes the project as 'Docker Deployment of Full Stack Project using Actions'. The 'Languages' section shows a breakdown: JavaScript (40.0%), Java (35.7%), CSS (18.6%), Dockerfile (3.2%), and HTML (1.7%). The 'Suggested workflows' section is based on the tech stack.

Setting->secret and variables->actions (to add secret variables)

The screenshot shows the 'General' tab of the GitHub repository settings for 'GITACTIONS-DOCKER-FULLSTACK-105'. The 'General' section contains settings for repository name ('GITACTIONS-DOCKER-FULLSTACK-105'), access, and code automation. It includes a 'Template repository' checkbox, a 'Require contributors to sign off on web-based commits' checkbox, and a 'Default branch' dropdown set to 'main'. The 'Pages' section has a 'Preview' button. Other tabs available include 'Releases', 'Security', 'Advanced Security', 'Deploy keys', 'Environments', 'Codespaces', and 'Secrets and variables'.

->New repository secret add

The screenshot shows the GitHub Actions secrets settings page for the repository 'Neeraj2005-github/GITACIONS-DOCKER-FULLSTACK-105'. The left sidebar includes options like General, Access, Collaborators, Code and automation, Tags, Rules, Actions, Models, Webhooks, Copilot, Environments, Codespaces, and Pages. The main content area is titled 'Actions secrets and variables' and contains two sections: 'Environment secrets' and 'Repository secrets'. The 'Environment secrets' section shows a message 'This environment has no secrets.' and a 'Manage environment secrets' button. The 'Repository secrets' section shows a message 'This repository has no secrets.' and a 'New repository secret' button. The status bar at the bottom indicates it's 11:50 AM on 18-09-2025.

->Add dockerhub username as secretvariable with secret name as DOCKERHUB_USERNAME (can add any name but need to change in docker-image.yml)

The screenshot shows the 'Actions secrets / New secret' creation page. The 'Name' field is filled with 'DOCKERHUB_USERNAME'. The 'Secret' field contains the value 'neeraj2005'. A green 'Add secret' button is visible at the bottom of the secret input field. The left sidebar and status bar are identical to the previous screenshot.

->secret variable DOCKERHUB_USERNAME added

The screenshot shows the GitHub Actions secrets and variables page for a repository. The left sidebar has sections like General, Access, Collaborators, Code and automation, Branches, Tags, Rules, Actions (selected), Models, Webhooks, Copilot, Environments, Codespaces, Pages, Security, Advanced Security, Deploy keys, Secrets and variables (selected), Actions, and Codespaces. The main area is titled 'Actions secrets and variables'. It has two tabs: 'Secrets' (selected) and 'Variables'. Under 'Secrets', it says 'This environment has no secrets.' and has a 'Manage environment secrets' button. Under 'Repository secrets', there is a table with one row:

Name	Last updated
DOCKERHUB_USERNAME	now

->Dockerhub ->profile->Account settings->Personal access tokens

The screenshot shows the Docker Hub profile page for 'neeraj2005'. The left sidebar has sections like Repositories (selected), Collaborations, Settings, Default privacy, Notifications, Billing, Usage, Pulls, and Storage. The main area is titled 'Repositories' and lists two repositories: 'neeraj2005/springboot-app' (last pushed 7 days ago) and 'neeraj2005/my-reactapp' (last pushed 19 days ago). On the right side, there is a sidebar for 'neeraj2005' with options for 'What's new', 'My profile', 'Account settings' (selected), and 'Sign out'. The status bar at the bottom shows '11:59' and '18-09-2025'.

The screenshot shows the 'Account information' section of the Docker Hub settings. The 'Personal access tokens' tab is selected. A new token is being created with the following details:

- Full name: NEERAJ SAKALA
- Company: (empty)
- Location: (empty)
- Website: (empty)
- Gravatar email: (empty)

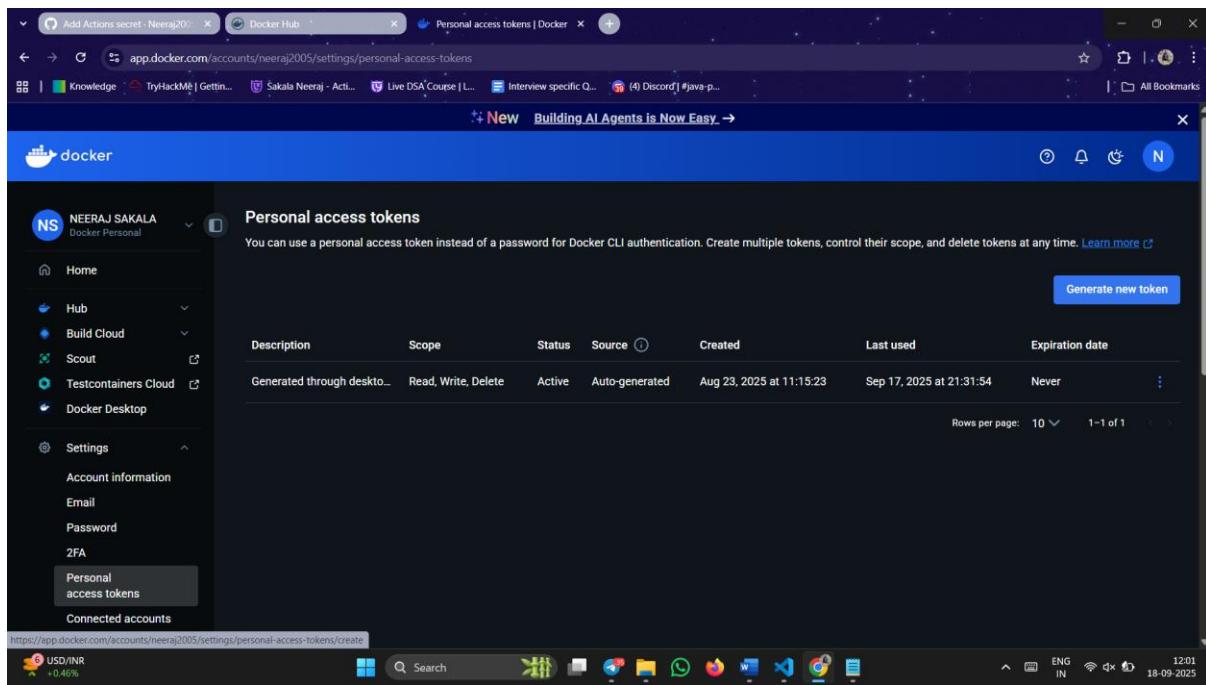
Buttons at the bottom are 'Cancel' and 'Save'. The Docker Hub sidebar on the left is visible.

->Personal access tokens and Create a new token with description and expiration date , make sure give Write,Read and Delete ->Generate

The screenshot shows the 'Create access token' page. The token details are as follows:

- Access token description: DOCKER FULL STACK
- Expiration date: None
- Access permissions: Read, Write, Delete

Buttons at the bottom are 'Cancel' and 'Generate'. The Docker Hub sidebar on the left is visible.

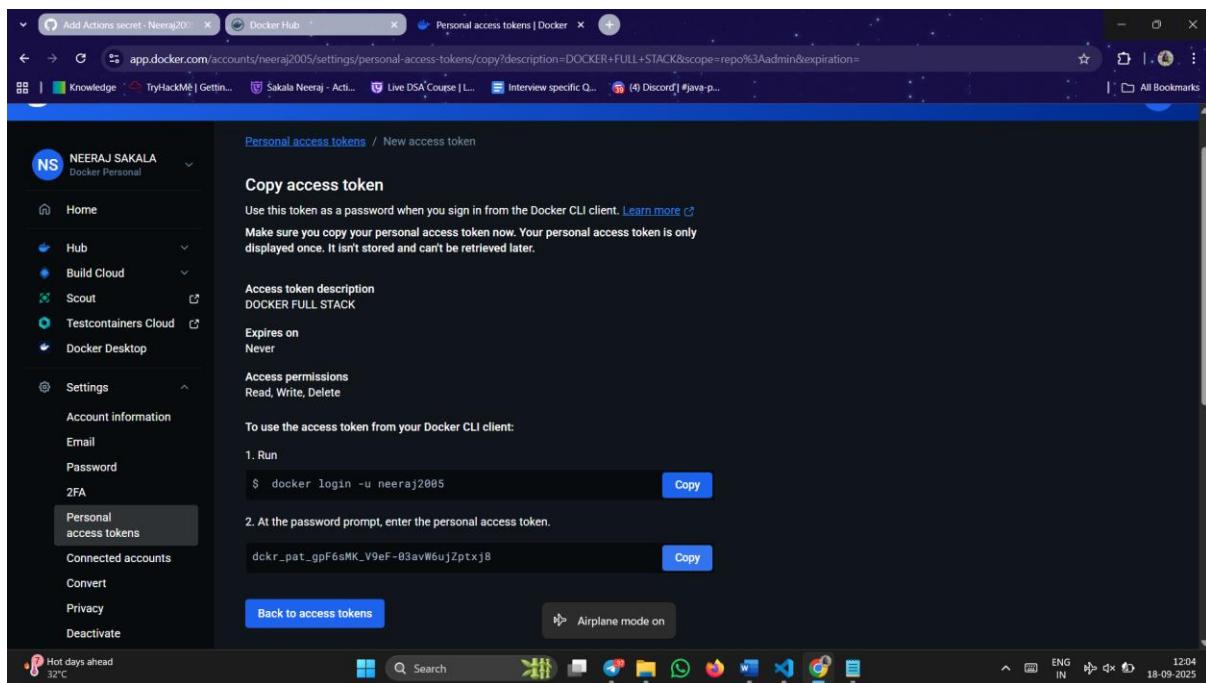


The screenshot shows the Docker Personal access tokens page. The left sidebar has a 'Personal access tokens' section selected. The main content area displays a table of tokens:

Description	Scope	Status	Source	Created	Last used	Expiration date
Generated through desktop...	Read, Write, Delete	Active	Auto-generated	Aug 23, 2025 at 11:15:23	Sep 17, 2025 at 21:31:54	Never

At the top right, there is a 'Generate new token' button. The bottom right shows 'Rows per page: 10' and '1-1 of 1'. The browser status bar at the bottom indicates '12:01 18-09-2025'.

->copy the personal access token for adding as secret variable token



The screenshot shows the Docker 'Copy access token' page. The left sidebar has a 'Personal access tokens' section selected. The main content area displays instructions for copying the token:

Use this token as a password when you sign in from the Docker CLI client. [Learn more](#)

Make sure you copy your personal access token now. Your personal access token is only displayed once. It isn't stored and can't be retrieved later.

Access token description: DOCKER FULL STACK

Expires on: Never

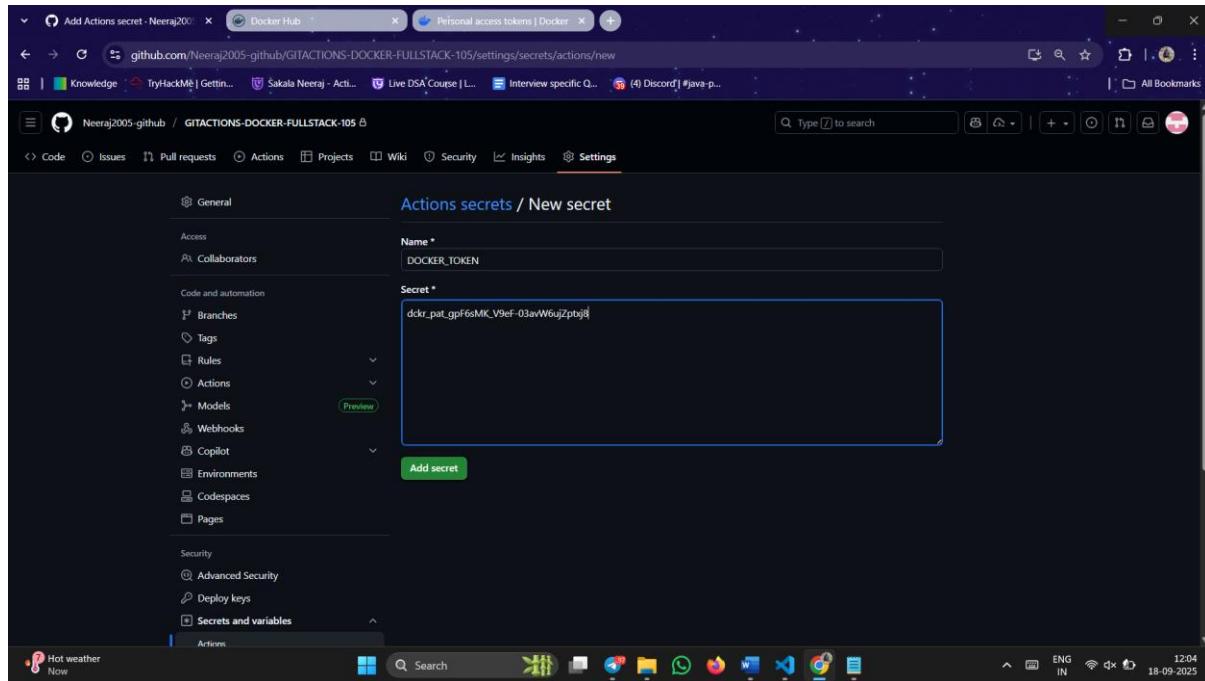
Access permissions: Read, Write, Delete

To use the access token from your Docker CLI client:

1. Run:
\$ docker login -u neeraj2005 Copy
2. At the password prompt, enter the personal access token.
dckr_pat_gpF6sMK_V9eF-03avW6ujZptxj8 Copy

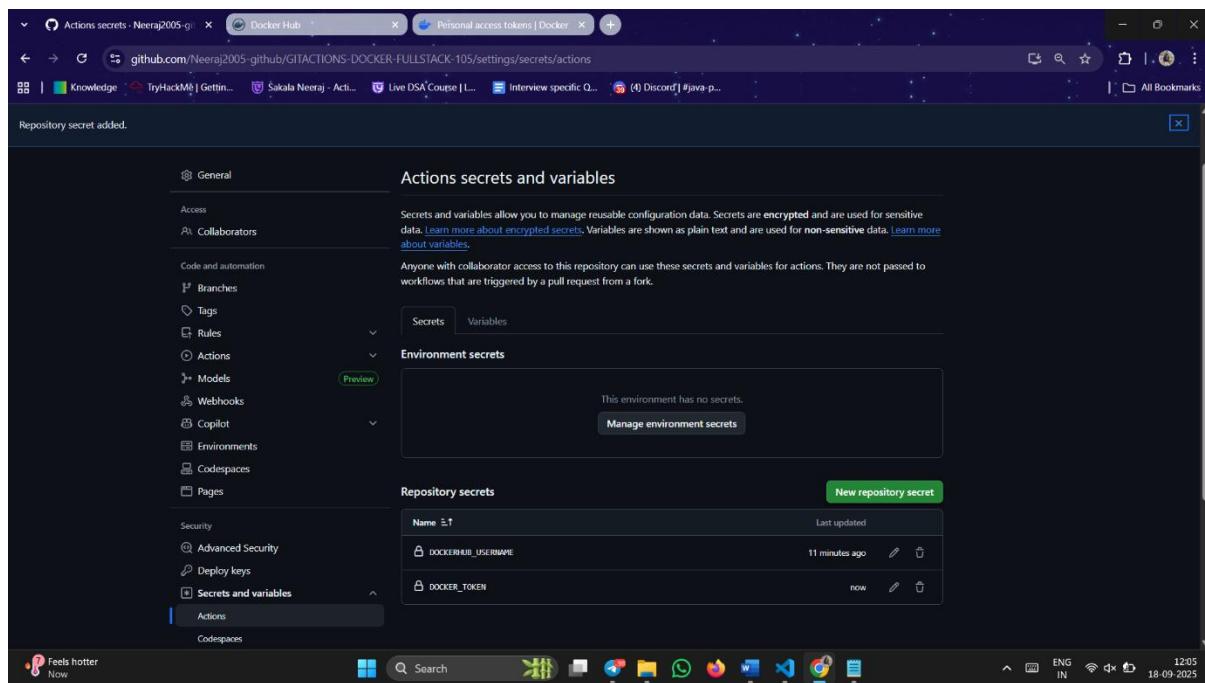
[Back to access tokens](#) Airplane mode on

->Add the copied token to secret variable as DOCKERHUB_SECRET properly without any space.



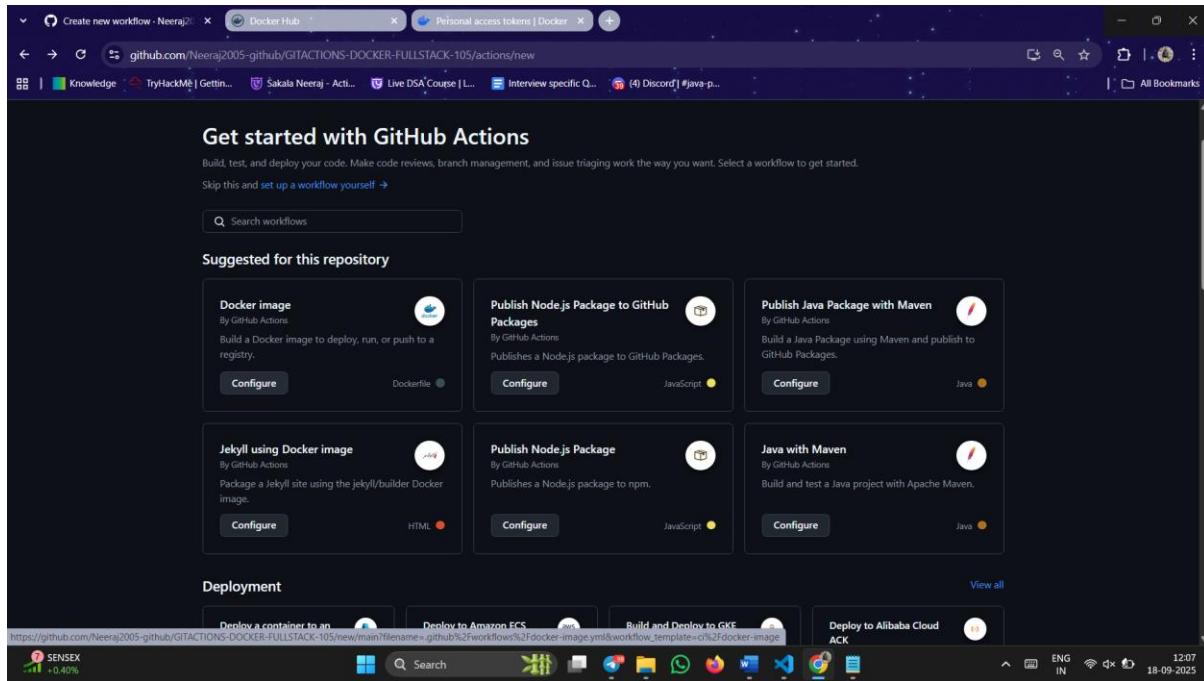
The screenshot shows the 'Actions secrets / New secret' page on GitHub. The 'Name' field is filled with 'DOCKER_TOKEN'. The 'Secret' field contains the copied Docker token: 'dckr_pat_gpf6sMK_V9ef-03avW6ujZpbj8'. A green 'Add secret' button is at the bottom right of the secret input area.

->Add secret

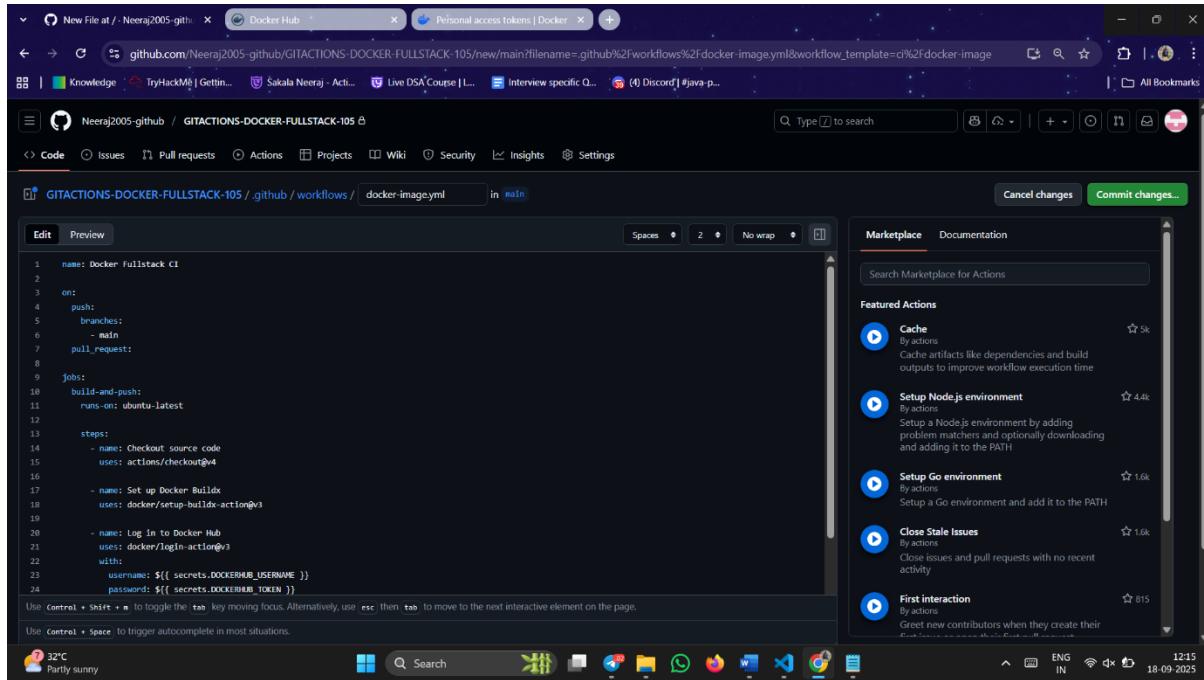


The screenshot shows the 'Actions secrets and variables' page. It displays two sections: 'Environment secrets' (which is empty) and 'Repository secrets'. Under 'Repository secrets', there are two entries: 'DOCKERHUB_USERNAME' (last updated 11 minutes ago) and 'DOCKER_TOKEN' (last updated now). A green 'New repository secret' button is located at the top right of the 'Repository secrets' table.

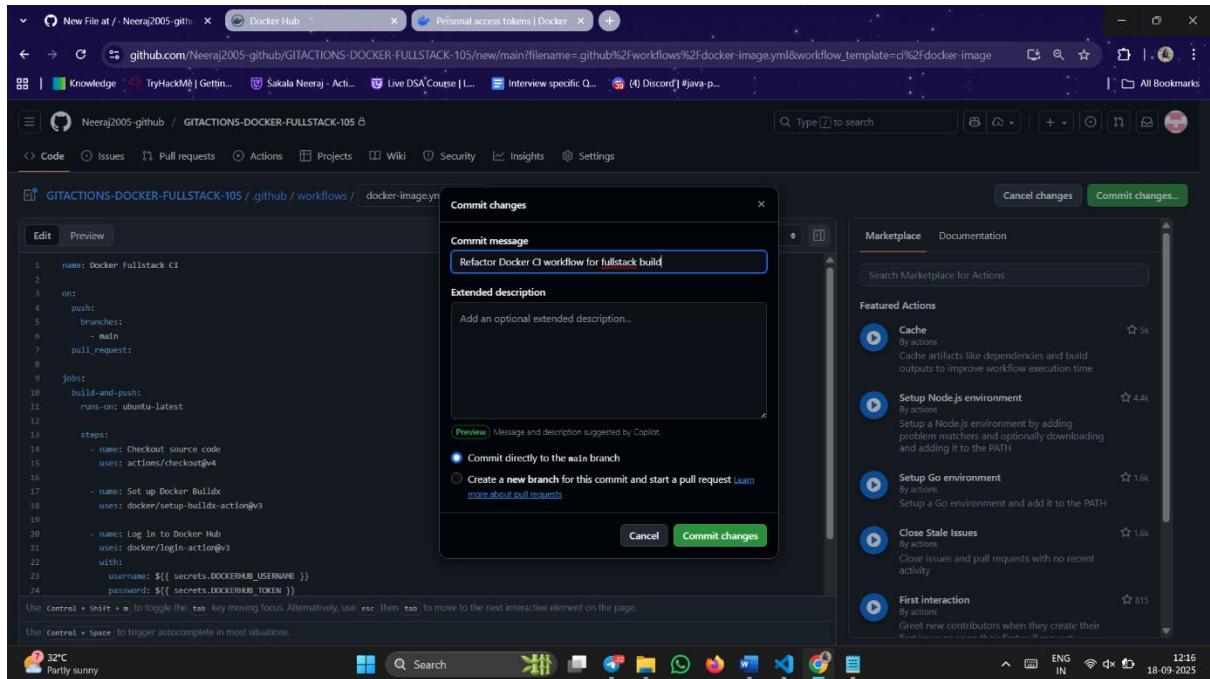
->Actions and configure Docker image as below:



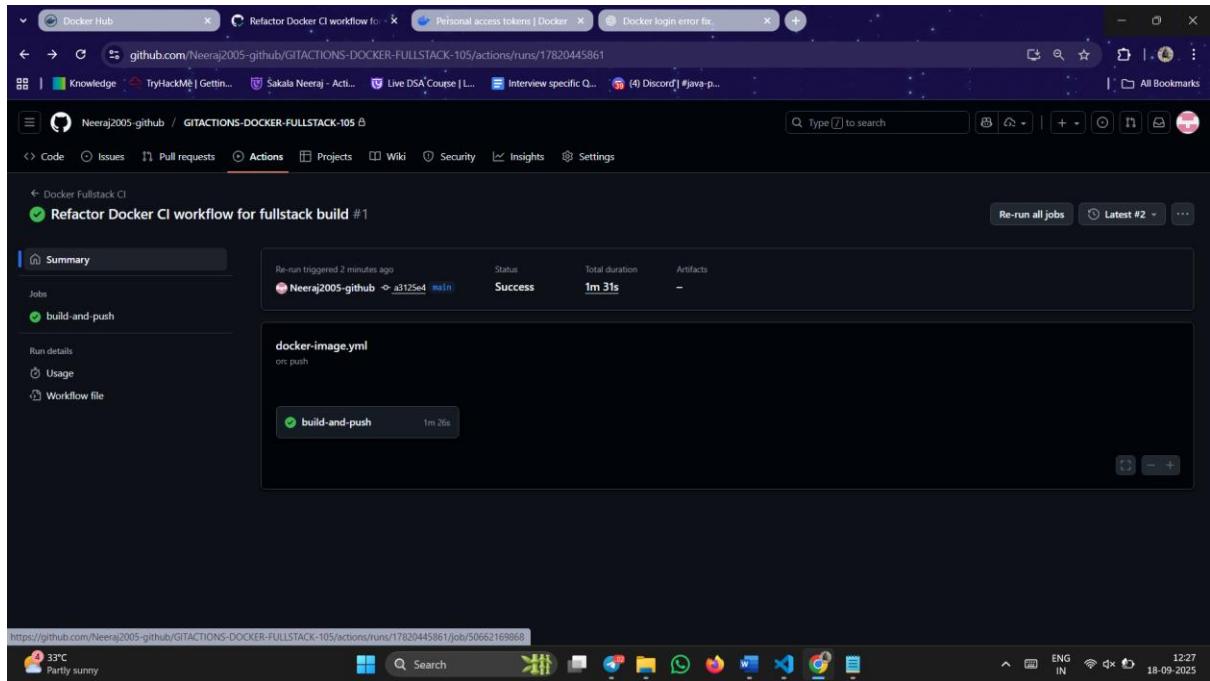
->The configured .yml file with secret variables and dockerhub username make sure to be correct names of variables created and image name



->commit changes



->go to actions and check for successful build



->Check images build in docker hub using git actions

The screenshot shows the Docker Hub interface. On the left, there's a sidebar for the user 'neeraj2005' with options like Repositories, Collaborations, Settings, Default privacy, Notifications, Billing, Usage, Pulls, and Storage. The main area is titled 'Repositories' and shows a list of four repositories: 'neeraj2005/docker-frontend', 'neeraj2005/docker-backend', 'neeraj2005/springboot-app', and 'neeraj2005/my-reactapp'. Each repository entry includes the name, last pushed time, visibility (Public), and status (Inactive). A 'Create a repository' button is visible at the top right of the list. The bottom of the screen shows a taskbar with various icons and system status.

->in docker-compose.yml make sure username is correct for frontend image name

The screenshot shows the VS Code editor with the file 'docker-compose.yml' open. The code is as follows:

```
volumes:
  - mysql_data:/var/lib/mysql
networks:
  - springboot-mysql-net

backend:
  image: neeraj2005/docker-backend:latest
  container_name: springboot
  ports:
    - "2025:2000"
  depends_on:
    - mysqldb
  restart: on-failure
  networks:
    - springboot-mysql-net

frontend:
  image: suryakiran1993/docker-frontend:latest
  container_name: react
  ports:
    - "3000:80"
  depends_on:
    - backend
```

The code editor interface is visible, including the Explorer sidebar on the left and the bottom navigation bar.

->in docker-compose.yml make sure username is correct for backend image name

```

version: '3.8'
services:
  backend:
    image: neeraj2005/docker-backend:latest
    container_name: springboot
    ports:
      - "2025:2000"
    depends_on:
      - mysql
    restart: on-failure
  frontend:
    image: neeraj2005/docker-frontend:latest
    container_name: react
    ports:
      - "3000:80"
    depends_on:
      - backend
    restart: on-failure
  mysql:
    image: mysql:8.0
    networks:
      - springboot-mysql-net
    environment:
      MYSQL_ROOT_PASSWORD: password
      MYSQL_DATABASE: db
      MYSQL_USER: user
      MYSQL_PASSWORD: password
    volumes:
      - mysql_data:/var/lib/mysql
    restart: always
volumes:
  mysql_data:

```

->As Configured image.yml in git repo we need to git pull first in local repo to push updated code as it create merge conflict .

->And then push the updated code

```

PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACCTIONS-DOCKER-FULLSTACK> git merge origin/main
.gitHub/workflows/docker-image.yml | 40 ++++++
1 file changed, 40 insertions(+)
create mode 100644 .github/workflows/docker-image.yml
● PS D: [Follow link (ctrl + click)] CI CD DEVEOPS\DOCKER\GITACCTIONS-DOCKER-FULLSTACK> git pull origin main
From https://github.com/Neeraj2005/github/GITACCTIONS-DOCKER-FULLSTACK-105
 * branch            main       -> FETCH_HEAD
Already up to date.

● PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACCTIONS-DOCKER-FULLSTACK> git push origin main
Enumerating objects: 9, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 12 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 573 bytes | 286.00 KiB/s, done.
Total 5 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/Neeraj2005/github/GITACCTIONS-DOCKER-FULLSTACK-105.git
  a3125e4..37523a2 main -> main

```

->we can see the updated code started actions after push without any manual deployment

The screenshot shows a web browser window with multiple tabs open. The active tab is 'github.com/Neeraj2005-github/GITACTONS-DOCKER-FULLSTACK-105/actions'. The page displays the 'Actions' section for the repository 'GITACTONS-DOCKER-FULLSTACK-105'. On the left, there's a sidebar for 'Docker Fullstack CI' with sections for Management, Caches, Attestations, Runners, Usage metrics, and Performance metrics. The main content area shows 'All workflows' with a search bar and a filter for 'Workflow runs'. There are two workflow runs listed:

Workflow	Branch	Event	Status	Time Ago
Merge remote-tracking branch 'origin/main'	main	Pushed	Success	2 minutes ago 1m 12s
Refactor Docker CI workflow for fullstack build	main	Pushed	Success	24 minutes ago 1m 31s

The browser's status bar at the bottom shows the date as 18-09-2025 and the time as 12:41.

The screenshot shows two instances of a Java development environment, likely VS Code, demonstrating a CI/CD pipeline for a Spring Boot application using Docker and Git Actions.

Top Window (Light Theme):

- Explorer:** Shows the project structure under "GITATIONS-DOCKER-FULLSTACK".
- Terminal:** Displays the following command and its output:

```
PS D:\3-1 CI CD DEVEOPS\DOCKER\GITATIONS-DOCKER-FULLSTACK> PS D:\3-1 CI CD DEVEOPS\DOCKER\GITATIONS-DOCKER-FULLSTACK>
```
- Status Bar:** Shows the current weather (30°C, Sunny), date (20-09-2025), and time (09:46).

Bottom Window (Dark Theme):

- Explorer:** Shows the same project structure.
- Terminal:** Displays the command and its output, followed by a commit message:

```
PS D:\3-1 CI CD DEVEOPS\DOCKER\GITATIONS-DOCKER-FULLSTACK> git commit -m "Project v2.0"
[main ee8cc22] Project v2.0
 1 file changed, 6 insertions(+), 2 deletions(-)
PS D:\3-1 CI CD DEVEOPS\DOCKER\GITATIONS-DOCKER-FULLSTACK> PS D:\3-1 CI CD DEVEOPS\DOCKER\GITATIONS-DOCKER-FULLSTACK>
```
- Status Bar:** Shows the current weather (30°C, Sunny), date (20-09-2025), and time (09:48).

The code in both windows is identical:

```
17 public class StudentController {  
18     @Autowired  
19     private StudentService studentService;  
20     @GetMapping("/docker")  
21     public String docker()  
22     {  
23         return "Docker Full Stack Deployment Demo with Git Actions";  
24     }  
25     @GetMapping("/")  
26     public String home()  
27     {  
28         return "Docker-Backend is Running";  
29     }  
30     @PostMapping("/add")  
31     public ResponseEntity<Student> addStudent(@RequestBody Student student) {  
32         Student savedStudent = studentService.addStudent(student);  
33     }  
34 }
```

The screenshot displays a Windows desktop environment with two main windows open:

- Code Editor (VS Code):** The title bar reads "GITACTIONS-DOCKER-FULLSTACK". The left sidebar shows a project structure under "GITACTIONS-DOCKER-FULLSTACK" with "STUDENTAPI-SPRINGBOOT" expanded, showing "src", ".mvn", and "controller". The "controller" folder contains "StudentController.java", which is currently selected and highlighted in orange. The code editor shows the following Java code:

```
17 public class StudentController {  
18     @Autowired  
19     private StudentService studentService;  
20     @GetMapping("/docker")  
21     public String docker()  
22     {  
23         return "Docker Full Stack Deployment Demo with Git Actions";  
24     }  
25     @GetMapping("/")  
26 }
```

- Browser (Microsoft Edge):** The address bar shows "github.com/Neeraj2005/github/GITACTIONS-DOCKER-FULLSTACK-105". The page content includes:
 - A terminal window showing a successful git push command to origin/main.
 - A GitHub repository page for "GITACCTIONS-DOCKER-FULLSTACK-105". The repository has 1 branch and 0 tags. It lists 15 commits from "Neeraj2005-github".
 - A "README" section with a "Add a README" button.
 - Project statistics: Docker Deployment of Full Stack Project using Actions.
 - Activity, stargazing, and forks counts.
 - Releases, Packages, and Languages sections.

The taskbar at the bottom shows various pinned icons, including Microsoft Office applications like Word, Excel, and Powerpoint, and other tools like Docker and Git. The system tray indicates it's 09:49 on 20-09-2025, the weather is 30°C and sunny, and the language is set to ENG IN.

The screenshot shows a Windows desktop environment with two main windows open:

- Top Window (GitHub Actions):** A browser window displaying the GitHub Actions page for the repository "Neeraj2005-github/GITACtions-Docker-Fullstack-105". The "Actions" tab is selected, showing three workflow runs:
 - Project v2.0**: Docker Fullstack CI #3: Commit ee8cc22 pushed by Neeraj2005-github. Status: main, 2 minutes ago, 1m 4s.
 - Merge remote-tracking branch 'origin/main'**: Docker Fullstack CI #2: Commit 37523a2 pushed by Neeraj2005-github. Status: main, Sep 18, 12:38 PM GMT+5:30, 1m 12s.
 - Refactor Docker CI workflow for fullstack build**: Docker Fullstack CI #1: Commit a3125e4 pushed by Neeraj2005-github. Status: main, Sep 18, 12:16 PM GMT+5:30, 1m 31s.
- Bottom Window (VS Code):** An IDE window showing the "StudentController.java" file from the "GITACTIONS-DOCKER-FULLSTACK" project. The code editor displays:

```
STUDENTAPI-SPRINGBOOT > src > main > java > com > klef > dev > controller > StudentController.java
17 public class StudentController {
18     @Autowired
```

The terminal tab shows the output of a Docker compose command:

```
PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACTIONS-DOCKER-FULLSTACK> docker compose up -d
[+] Running 5/5
✓ frontend Pulled
✓ 7d82f3ad3b43 Pull complete
✓ backend Pulled
✓ 7a4391ba01b6 Pull complete
✓ e15ae6e6fc8a Pull complete
[+] Running 5/5
✓ Network gitactions-docker-fullstack.springboot-mysql-net Created
✓ Volume "gitactions-docker-fullstack_mysql_data" Created
✓ Container mysql-db Started
✓ Container springboot Started
✓ Container react Started
```

The screenshot shows a Visual Studio Code (VS Code) interface with the following details:

- File Explorer (Left):** Shows the project structure under "GITACCTIONS-DOCKER-FULLSTACK". The "controller" folder contains "StudentController.java". Other visible folders include ".mvn", "src", "main", "java\com\klef\dev", and "test\java\com\klef\dev".
- Terminal (Bottom):** Displays the output of a Docker command:

```
PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACCTIONS-DOCKER-FULLSTACK> docker compose up -d
[+] Running 5/5
  ✓ frontend Pulled
    ✓ 7d82f3ad3b43 Pull complete
  ✓ backend Pulled
    ✓ 7a4391ba01b6 Pull complete
    ✓ e15ae6e6fc8a Pull complete
[+] Running 5/5
  ✓ Network gitactions-docker-fullstack.springboot-mysql-net Created
  ✓ Volume "gitactions-docker-fullstack_mysql_data" Created
  ✓ Container mysql-db Started
  ✓ Container springboot Started
  ✓ Container react Started
PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACCTIONS-DOCKER-FULLSTACK> docker-compose down -v
[+] Running 5/5
  ✓ Container react Removed
  ✓ Container springboot Removed
  ✓ Container mysql-db Removed
  ✓ Network gitactions-docker-fullstack.springboot-mysql-net Removed
  ✓ Volume gitactions-docker-fullstack_mysql_data Removed
PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACCTIONS-DOCKER-FULLSTACK>
```
- Status Bar (Bottom):** Shows the current file is "StudentController.java", it was last saved 4 days ago, and the cursor is at Ln 26, Col 21.
- Bottom Icons:** Includes icons for file operations like Open, Save, Find, and Replace, as well as links to GitHub, Go Live, and Prettier.

The screenshot shows a Visual Studio Code (VS Code) interface with a Docker CI pipeline running in the background. The terminal output indicates the following steps:

- `PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACtIONS-DOCKER-FULLSTACK> docker compose up -d`
- Execution results:
 - ✓ Network gitactions-docker-fullstack.springboot-mysql-net Created 0.1s
 - ✓ Volume "gitactions-docker-fullstack_mysql_data" Created 0.0s
 - ✓ Container mysql-db Started 1.1s
 - ✓ Container springboot Started 1.3s
 - ✓ Container react Started 1.6s
- `PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACtIONS-DOCKER-FULLSTACK> docker-compose down -v`
- Execution results:
 - [+] Running 5/5
 - ✓ Container react Removed 0.5s
 - ✓ Container springboot Removed 0.5s
 - ✓ Container mysql-db Removed 1.7s
 - ✓ Network gitactions-docker-fullstack.springboot-mysql-net Removed 0.3s
 - ✓ Volume gitactions-docker-fullstack_mysql_data Removed 0.0s
- `PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACtIONS-DOCKER-FULLSTACK> docker-compose up -d`
- Execution results:
 - [+] Running 5/5
 - ✓ Network gitactions-docker-fullstack.springboot-mysql-net Created 0.1s
 - ✓ Volume "gitactions-docker-fullstack_mysql_data" Created 0.0s
 - ✓ Container mysql-db Started 0.9s
 - ✓ Container springboot Started 1.2s
 - ✓ Container react Started 1.5s
- `PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACtIONS-DOCKER-FULLSTACK>`

The screenshot shows two instances of a Java IDE (likely IntelliJ IDEA) displaying the same project structure and code.

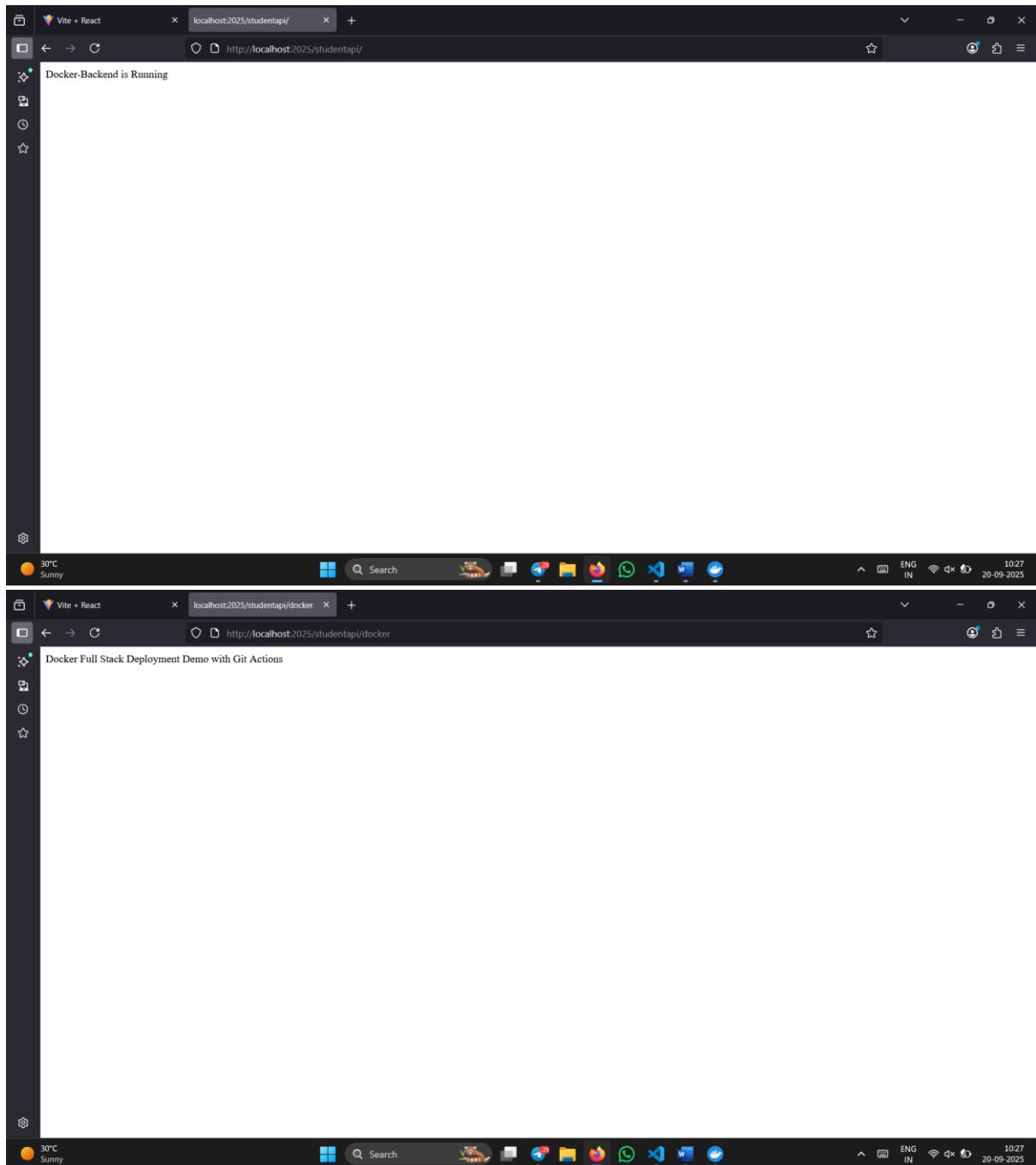
Project Structure:

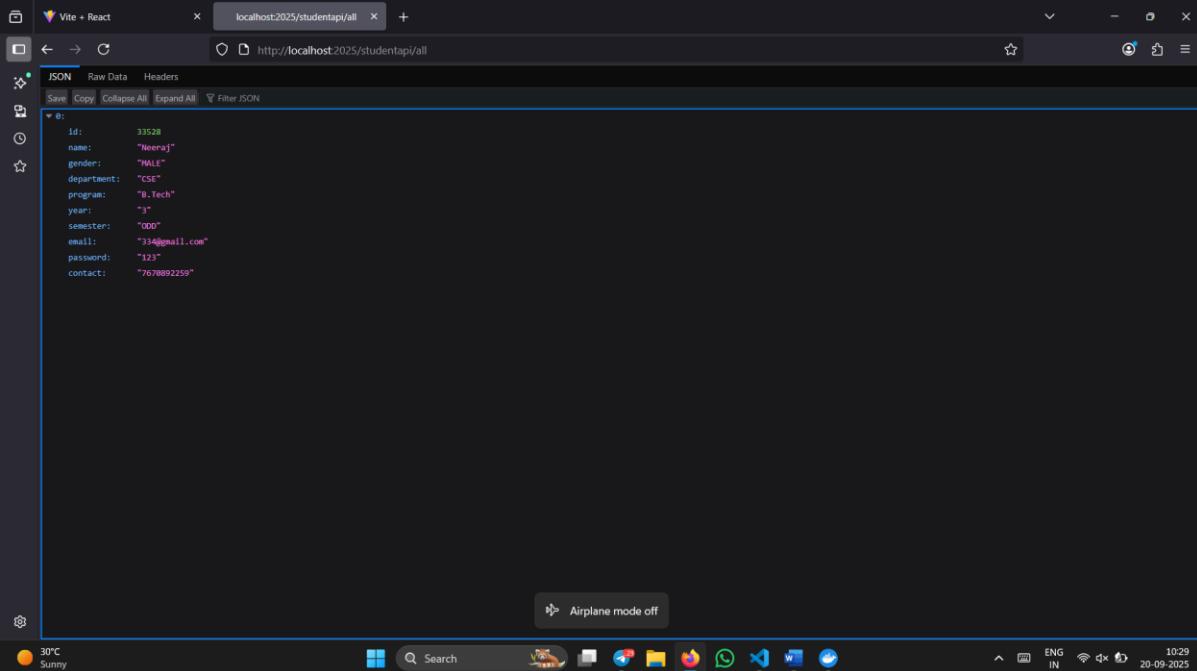
- EXPLORER** pane on the left shows the project tree:
 - GITACCTIONS-DOCKER-FULLSTACK** (selected)
 - STUDENTAPI-SPRINGBOOT**
 - .mvn
 - src
 - main
 - java\com\klef\dev
 - controller
 - StudentController.java (selected)
 - entity
 - repository
 - service
 - SpringBootProjectAppl...
 - resources
 - test\java\com\klef\dev
 - OUTLINE, TIMELINE, DOCKER CONTAINERS, DOCKER IMAGES, AZURE CONTAINER REGISTRY, DOCKER HUB, SUGGESTED DOCKER HUB IMAGES, JAVA PROJECTS

The screenshot shows a developer's environment with multiple windows open:

- Code Editor:** An IDE window displays `StudentController.java` with annotations like `@Autowired`. The terminal shows logs from a Docker container named `springboot`, indicating successful database schema creation and a Tomcat server startup.
- Terminal:** Shows the command `docker logs -f springboot` running in a Docker container.
- Browser:** A browser window shows a Spring Boot application at `http://localhost:3000`. It has two main sections:
 - Add Student:** A form with a "Fetch" button and a success message "Student updated successfully."
 - Get Student By ID:** A search bar with value "33528" and a "Fetch" button. Below it, a JSON response is shown under "Student Found":

```
{  "id": 33528,  "name": "Neeraj",  "gender": "MALE",  "department": "CSE",  "program": "B.Tech",  "year": "3",  "semester": "ODD",  "email": "334@gmail.com",  "password": "123",  "contact": "7670892259"} 
```
 - All Students:** A table listing one student record:
- File Explorer:** Shows the project structure for `STUDENTAPI-SPRINGBOOT`, including `src/main/java/com/klef/dev/controller` and `SpringBootProjectApplication.java`.
- System Tray:** Shows weather (30°C, Sunny), battery status, and system icons.





Vite + React

localhost:2025/studentapi/all

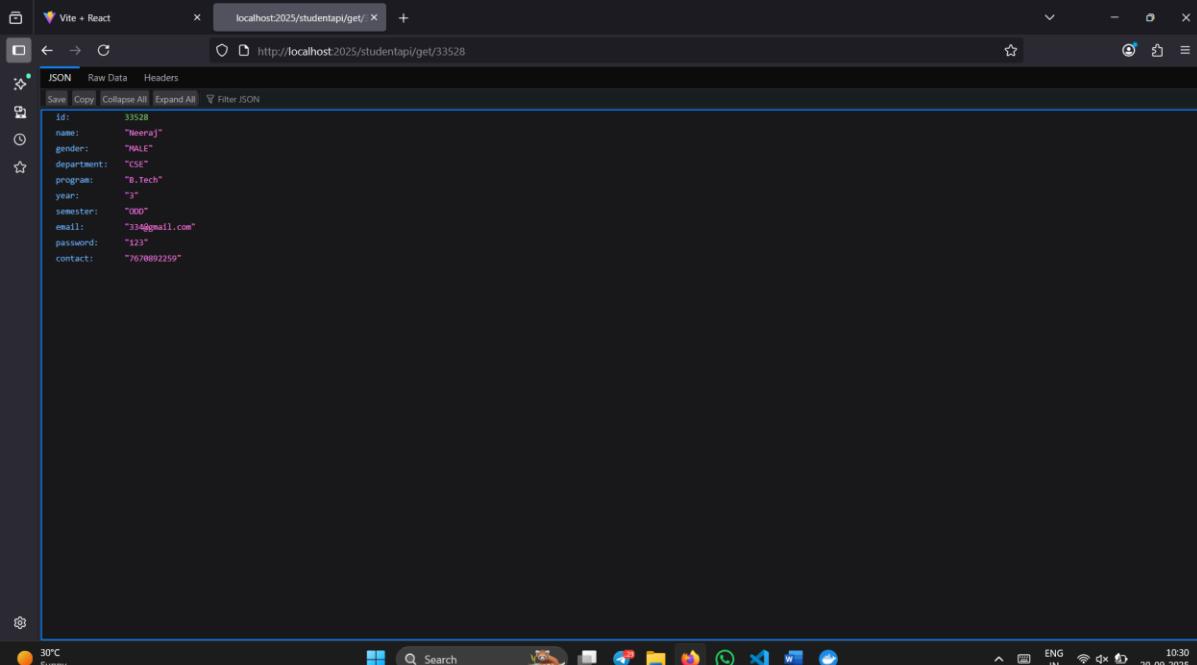
JSON Raw Data Headers

Save Copy Collapse All Expand All Filter JSON

```
* 0:
  id: "33528"
  name: "Neeraj"
  gender: "MALE"
  department: "CSE"
  program: "B.Tech"
  year: "3"
  semester: "000"
  email: "334@gmail.com"
  password: "123"
  contact: "7678892259"
```

30°C Sunny

Airplane mode off

ENG IN 10:29 20-09-2025


Vite + React

localhost:2025/studentapi/get/33528

JSON Raw Data Headers

Save Copy Collapse All Expand All Filter JSON

```
* 0:
  id: "33528"
  name: "Neeraj"
  gender: "MALE"
  department: "CSE"
  program: "B.Tech"
  year: "3"
  semester: "000"
  email: "334@gmail.com"
  password: "123"
  contact: "7678892259"
```

30°C Sunny

ENG IN 10:30 20-09-2025

The screenshot shows two instances of the Visual Studio Code (VS Code) interface. Both instances have the title bar "GITACTIONS-DOCKER-FULLSTACK".

Left Instance (Top):

- Terminal Tab:** Displays a MySQL shell session connected to a MySQL database named "mysql". The session includes standard MySQL welcome messages, copyright information, and a prompt for help or clearing input.
- Right Side Bar:** Shows icons for "powershell" and "docker".
- Bottom Status Bar:** Shows weather (1 cm of rain), date (Monday), and system status (ENG IN).

Right Instance (Bottom):

- Terminal Tab:** Displays a MySQL shell session connected to a MySQL database named "klustudentdb". It shows table completion information and a note about turning off the feature.
- Bottom Status Bar:** Shows weather (Finance headline: Trump Imposes...), date (20-09-2025), and system status (ENG IN).

In both instances, the left sidebar shows a project structure for "GITACTIONS-DOCKER-FULLSTACK" with "StudentController.java" selected in the Explorer tab. The right sidebar contains various development tools like Docker and MySQL.

The screenshot shows a Java development environment with the following details:

- File Bar:** File, Edit, Selection, View, ...
- Search Bar:** GITACCTIONS-DOCKER-FULLSTACK
- Terminal:** TERMINAL tab is selected, showing MySQL command-line interface output:

```
PS D:\3-1 CI CD DEVEOPS\DOCKER\GITACCTIONS-DOCKER-FULLSTACK> docker exec -it mysql-db mysql
1 -uroot -pdocker
mysql> show tables;
+-----+
| Tables_in_klustudentdb |
+-----+
| student_table |
+-----+
1 row in set (0.01 sec)

mysql> select * from student_table;
+-----+-----+-----+-----+-----+-----+-----+-----+
| student_id | student_contact | student_department | student_email | student_gender | student_name | student_password | student_program | student_semester | student_year |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 33528 | 7670892259 | CSE | 334@gmail.com | MALE | Neeraj | B.Tech | ODD | 3 |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```
- SIDE BAR:** Explorer, Outline, Timeline, Docker Containers, Docker Images, Azure Container Registry, Docker Hub, Suggested Docker Hub Images, Java Projects.
- Bottom Bar:** Finance headline, Search, Taskbar icons (File Explorer, Task View, Task Manager, Task Scheduler, Task History, Task Scheduler Log, Task Scheduler Library, Task Scheduler Object, Task Scheduler Object Log, Task Scheduler Object Library), Go Live, Prettier, Date/Time (20-09-2025).

docker-compose up -d

docker-compose down -v

docker logs -f containernname