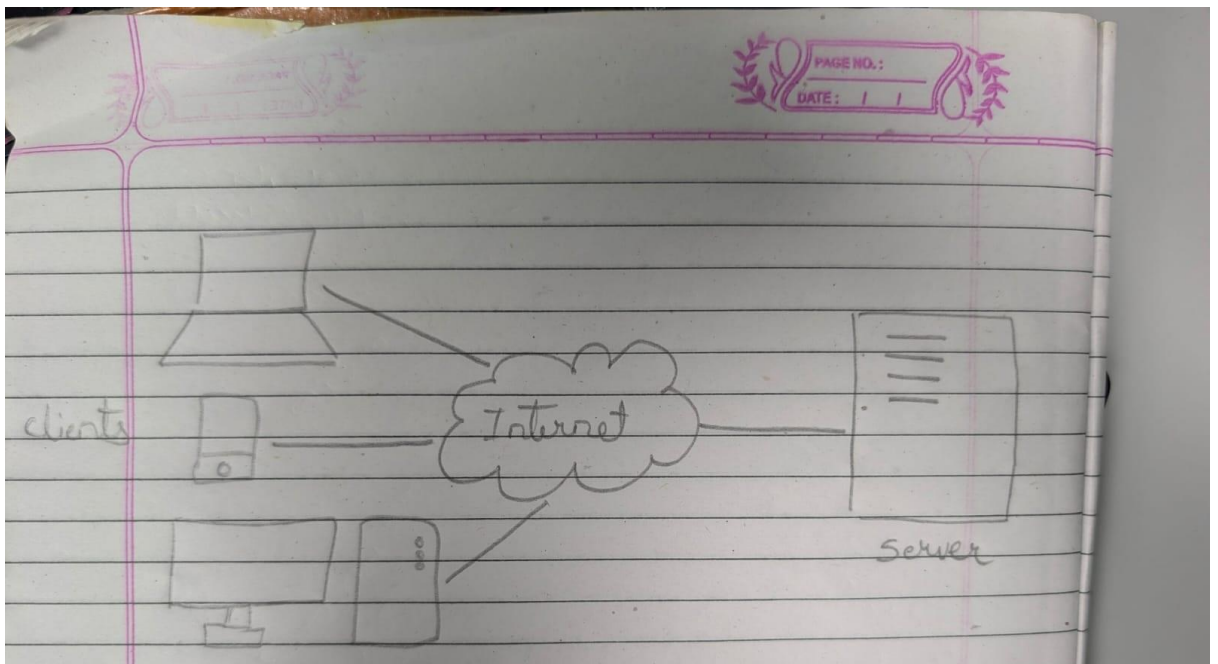


## Assignment - 1

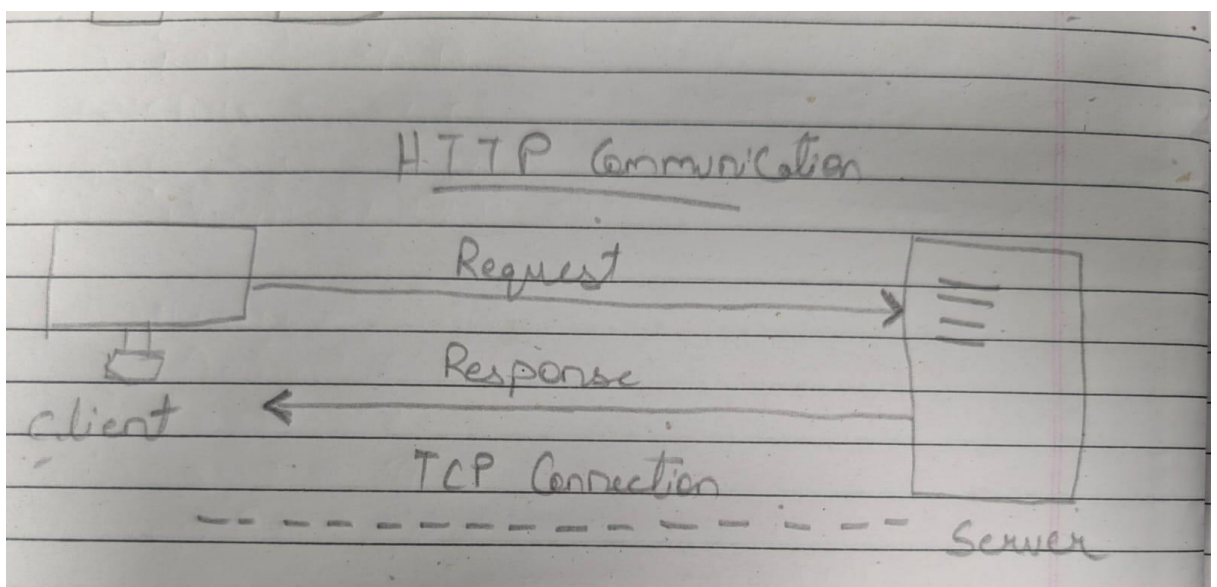
1. Write a simple "Hello World" program in two different programming languages of your choice. Compare the structure and syntax.

Uploaded on GitHub.

2. Research and create a diagram of how data is transmitted from a client to a server over the internet.



3. Design a simple HTTP client-server communication in any language.



## **Assignment - 1**

### **4. Research different types of internet connections (e.g., broadband, fibre, satellite) and list their pros and cons.**

#### **Broadband:**

##### **Pros:**

- High-speed internet access.
- Reliable and consistent connection.
- Can support multiple devices simultaneously.
- Often more affordable than satellite internet.

##### **Cons:**

- May not be available in all remote or rural areas.
- Can be more expensive than dial-up.
- Potential security risks.

#### **Fibre Optic:**

##### **Pros:**

- Fastest connection speeds available.
- Extremely reliable and consistent performance.
- Symmetrical upload and download speeds (upload speeds are as fast as download speeds).
- Less susceptible to interference than other connection types.

##### **Cons:**

- Limited availability, often only in urban and suburban areas.
- Higher installation costs: compared to other broadband options.
- May require professional installation.

#### **Satellite:**

##### **Pros:**

- Widest availability, especially in remote and rural areas.
- Can be a good option when other types of internet are not available.

##### **Cons:**

- Slower speeds compared to broadband and fibre.
- Higher latency: (delay in data transmission).
- Susceptible to weather interference.
- More expensive: than most broadband options.

## Assignment - 1

### 5. Simulate HTTP and FTP requests using command line tools (e.g., curl).

HTTP requests using Curl –

```
curl -u user:password http://example.org/
```

```
curl https://curl.se
```

FTP requests using Curl –

```
curl ftp://ftp.slackware.com
```

```
curl ftp://ftp.slackware.com/welcome.msg
```

```
curl -u "anonymous:password" ftp://ftp.slackware.com/
```

### 6. Identify and explain three common application security vulnerabilities. Suggest possible solutions.

**Three common application security vulnerabilities are:**

SQL Injection

Cross-Site Scripting

Broken Access Control

**Solution to these vulnerabilities are as follow:**

Implement prepared statements to prevent injection of malicious SQL queries.

Implement output encoding or escaping to ensure that user-supplied data is treated as text and not as executable code.

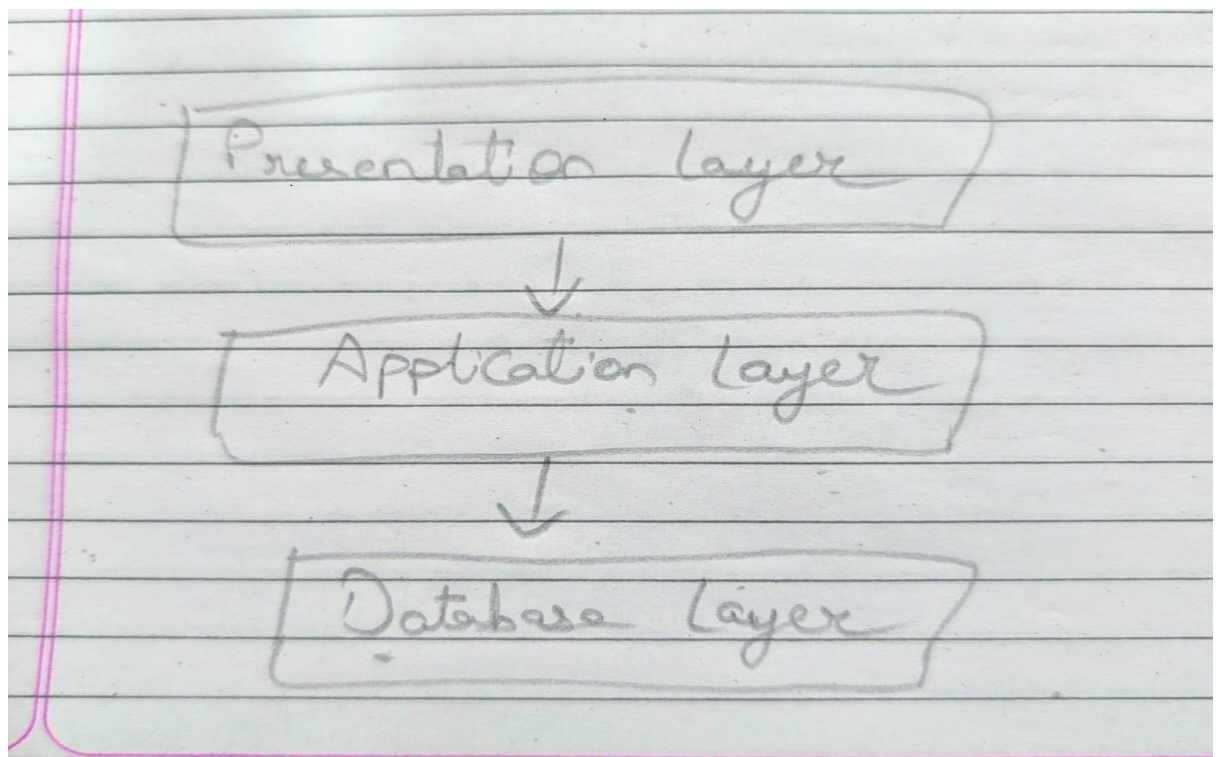
Implement robust access control mechanisms, such as role-based access control (RBAC) or attribute-based access control (ABAC), to ensure that users only have access to the resources and functionality they are authorized to use.

## Assignment - 1

7. Identify and classify 5 applications you use daily as either system software or application software

- Visual Studio Code
- Google Chrome
- File Manager
- Sublime Text
- VLC Media Player

8. Design a basic three-tier software architecture diagram for a web application.



## Assignment - 1

### 9. Create a case study on the functionality of the presentation, business logic, and data access layers of a given software system.

#### Presentation Layer:

**User Interface:** The presentation layer is the user interface, including web pages, mobile apps, or desktop applications.

**Interactions:** It handles user input and displays data to the user.

**Communication:** It communicates with the business logic layer to initiate actions and receives processed data for display.

#### Business Logic Layer:

**Application Rules:** This layer contains the core business logic, including validation, calculations, and decision-making processes.

**Data Processing:** It processes data received from the presentation layer and interacts with the data access layer to retrieve or modify data.

**Reusability:** Business logic is designed to be reusable across multiple presentation layers or even other applications.

#### Data Access Layer:

**Database Interaction:** The data access layer handles all interactions with the database or other data sources.

**Data Retrieval:** It retrieves data from the database based on requests from the business logic layer.

**Data Persistence:** It also handles data persistence, including inserting, updating, and deleting data in the database.

### 10. Explore different types of software environments (development, testing, Production). Set up a basic environment in a virtual machine.

#### Development Environment:

This is where developers write, test, and debug code.

It's a workspace for creating and refining the software.

Integrated Development Environments (IDEs) like VS Code, Eclipse, or IntelliJ IDEA are often used in development environments.

# Assignment - 1

## Testing Environment:

This environment is dedicated to thorough testing of the application, including functional, performance, and security tests.

It helps identify and fix bugs before the software is released to users.

## Staging Environment:

A staging environment is a near-production replica used for final testing and validation before deployment.

It allows teams to simulate the live environment and ensure a smooth transition to production.

## Production Environment:

This is the live environment where the application is accessible to end-users.

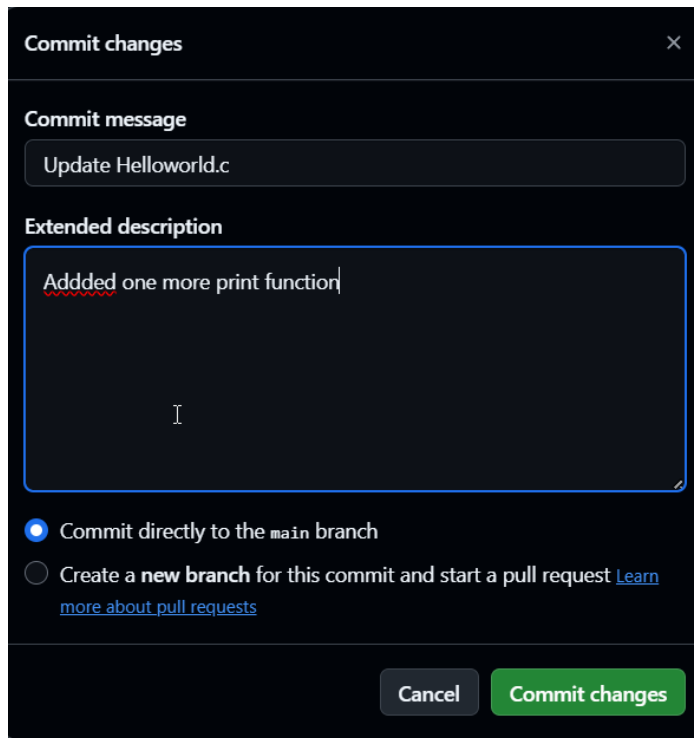
## 11. Write and upload your first source code file to GitHub.

- Go to <https://github.com/dashboard> and log in.
- Click the + icon--->new repository.
- Enter a repository name.
- Choose: public and private.
- Click create repository.

## 12. Create a GitHub repository and document how to commit and push code changes.

- Open the file you to make changes for.
- Make the changes.
- Click On: Commit Changes button.
- Add description for more understanding of the commit changes.
- Now click on: Commit Changes button to make the changes.

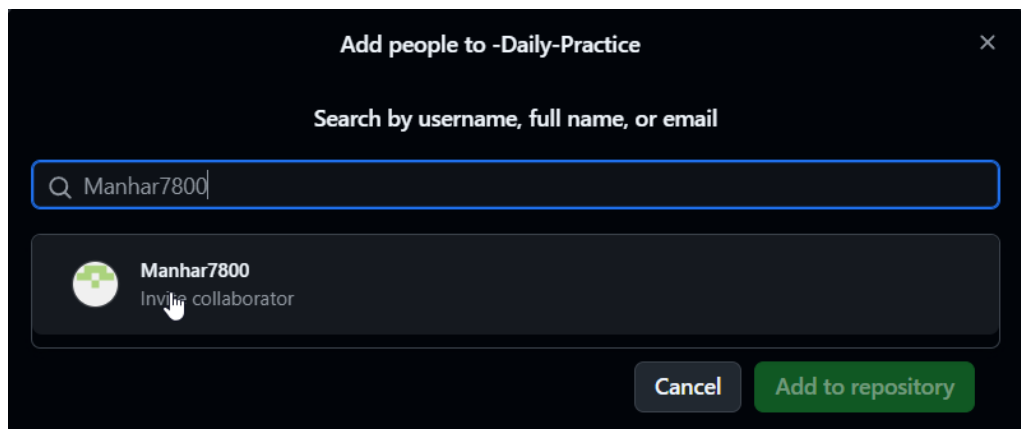
## Assignment - 1



The screenshot shows a 'Commit changes' dialog box with a dark theme. At the top, there's a title bar with a close button. Below it, the 'Commit message' section has a text input field containing 'Update Helloworld.c'. The 'Extended description' section has a larger text area with the text 'Added one more print function|'. Below the text area, there are two radio button options: 'Commit directly to the main branch' (which is selected) and 'Create a new branch for this commit and start a pull request' (with a link to 'Learn more about pull requests'). At the bottom right, there are two buttons: 'Cancel' and 'Commit changes'.

### 13. Create a student account on GitHub and collaborate on a small project with a classmate.

Already have an Account and collaboration screenshot is down below:



The screenshot shows a dialog box titled 'Add people to -Daily-Practice'. It has a search bar with the placeholder text 'Search by username, full name, or email' and the search term 'Manhar7800'. Below the search bar, there is a list of search results. The first result is a user profile for 'Manhar7800' with a green and white avatar and the text 'Invite collaborator'. At the bottom right, there are two buttons: 'Cancel' and 'Add to repository'.

## Assignment - 1

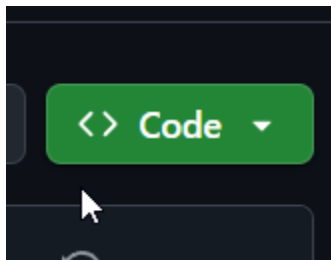
**14. Create a list of software you use regularly and classify them into the following categories: system, application, and utility software**

- System Software – Windows, Linux, MacOS, Device drivers, Antivirus Software.
- Application Software - Microsoft Word, Spreadsheet, Chrome, Firefox, VLC Media Player.
- Utility Software – File Compression tools, Microsoft PC manager, Backup and recovery Tools.

**15. Follow a GIT tutorial to practice cloning, branching, and merging repositories.**

**Cloning:**

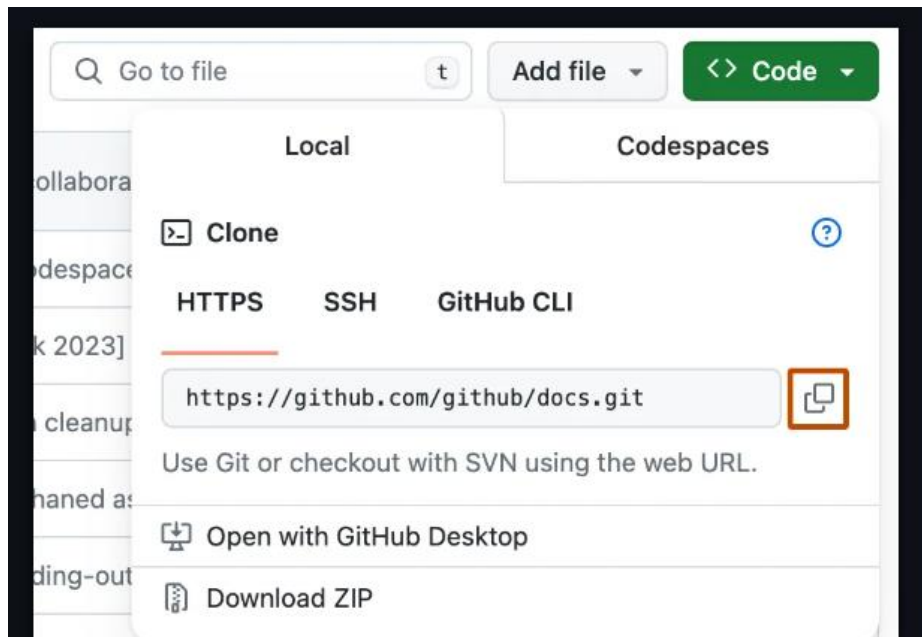
1. On GitHub, navigate to the main page of the repository.
2. Above the list of files, click Code.



3. Copy the URL for the repository.
  - To clone the repository using HTTPS, under "HTTPS", click.
  - To clone the repository using an SSH key, including a certificate issued by your organization's SSH certificate authority, click SSH, then click.
  - To clone repository using GitHub CLI, click GitHub CLI, then click.



## Assignment - 1



4. Open Git Bash.
5. Change the current working directory to the location where you want the cloned directory.
6. Type git clone, and then paste the URL you copied earlier.

git clone <https://github.com/shiv09-dev/Assignment-1>

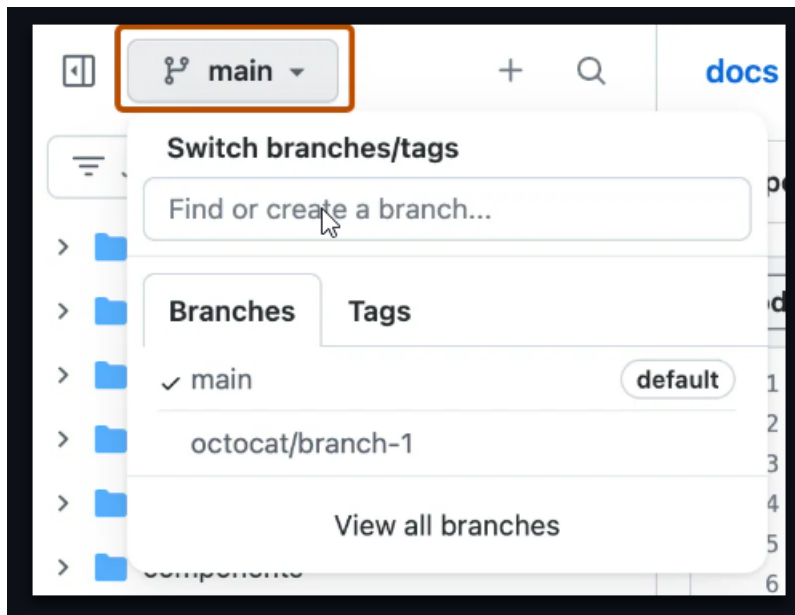
7. Press Enter to create your local clone.

```
> Cloning into `Spoon-Knife`...
> remote: Counting objects: 10, done.
> remote: Compressing objects: 100% (8/8), done.
> remove: Total 10 (delta 1), reused 10 (delta 1)
> Unpacking objects: 100% (10/10), done.
```

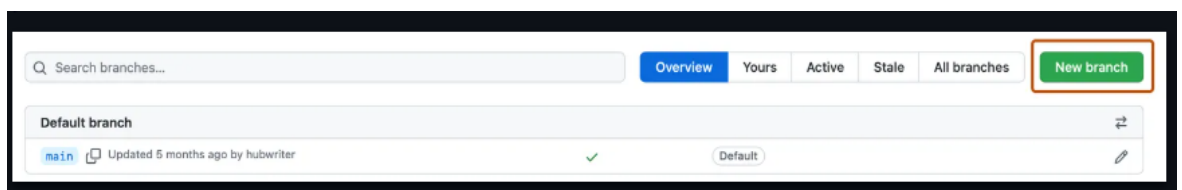
### Branching:

1. On GitHub, navigate to the main page of the repository.
2. From the file tree view on the left, select the branch dropdown menu, then click View all branches. You can also find the branch dropdown menu at the top of the integrated file editor.

## Assignment - 1



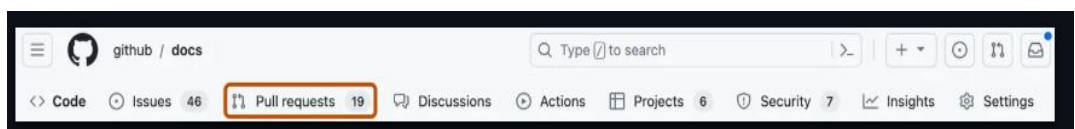
3. Click New branch.



4. Under "Branch name", type a name for the branch.
5. Under "Branch source", choose a source for your branch.
  - If your repository is a fork, select the repository dropdown menu and click your fork or the upstream repository.
  - Select the branch dropdown menu and click a branch.
6. Click Create branch.

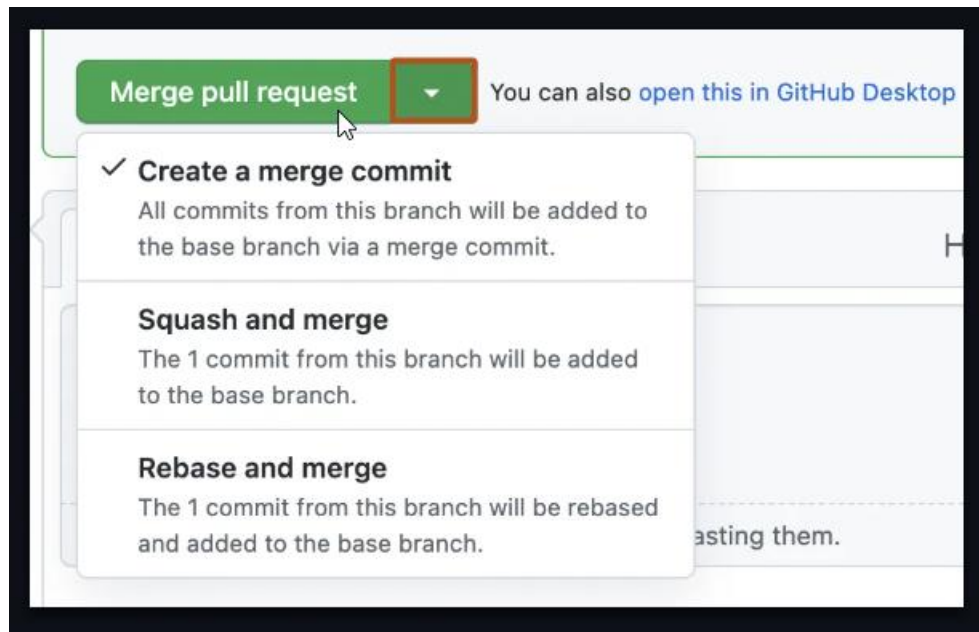
### Merging:

1. Under your repository name, click Pull requests.

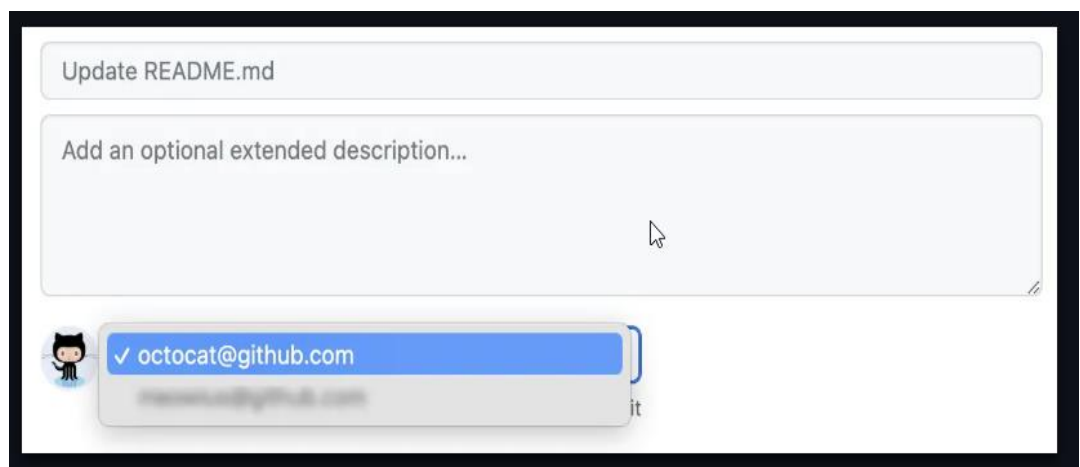


2. In the "Pull Requests" list, click the pull request you'd like to merge.
3. Scroll down to the bottom of the pull request. Depending on the merge options enabled for your repository, you can:

## Assignment - 1



4. If prompted, type a commit message, or accept the default message.
5. If you have more than one email address associated with your account on GitHub, click the email address drop-down menu and select the email address to use as the Git author email address. Only verified email addresses appear in this drop-down menu. If you enabled email address privacy, then a no-reply will be the default commit author email address. For more information about the exact form the no-reply email address can take, see [Setting your commit email address](#).



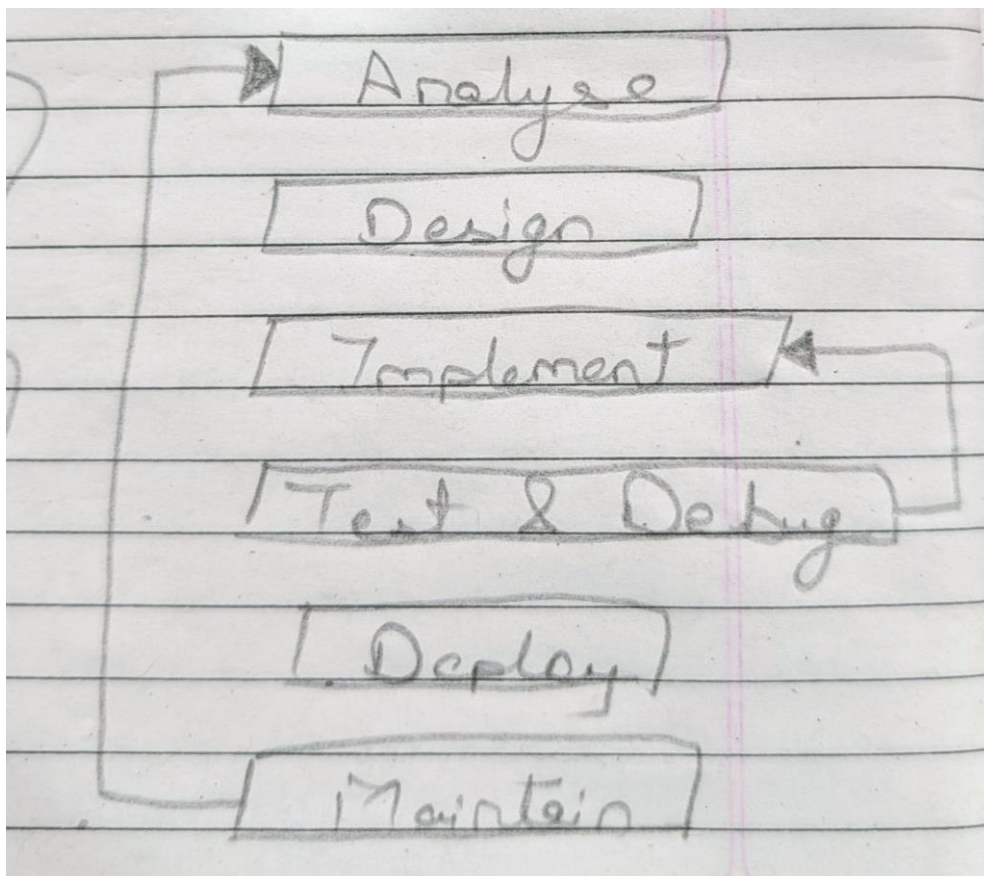
6. Click Confirm merge, Confirm squash and merge, or Confirm rebase and merge.
7. Optionally, delete the branch. This keeps the list of branches in your repository tidy.

## Assignment - 1

16. Write a report on the various types of application software and how they improve productivity.

- **Google Docs ,Microsoft Word -**
  - Enhances document creation with formatting tools and templates
  - Allows real-time collaboration and cloud saving
- **Microsoft Excel, Google Sheets**
  - Automates calculations and data analysis
  - Provides visual tools like charts for decision-making
- **Microsoft PowerPoint, Google Slides**
  - Helps convey ideas visually through slides and graphics
  - Offers design templates to save time and improve consistency
- **Oracle, MySQL, Microsoft Access**
  - Organizes and stores large amounts of data efficiently
  - Enables quick data retrieval through search and queries

17. Create a flowchart representing the Software Development Life Cycle (SDLC).



# Assignment - 1

## 18. Write a requirement specification for a simple library management system.

### 1. Purpose

To develop a simple system for managing books, members, and borrowing activities in a small library.

### 2. Functional Requirements

- Book Management: Add/update/delete books, check availability, search by title/author.
- Member Management: Add/update/delete members, view borrowing history.
- Issue/Return: Issue and return books, track due dates, block overdue borrowers.
- Fine Handling: Calculate/display late return fines.
- Reports/Search: Generate reports on issued/returned/overdue books.

### 3. Non-Functional Requirements

- Usability: Easy-to-use interface.
- Performance: Fast search and response time.
- Security: Login system and user roles.
- Reliability: Daily data backup.

### 4. System Requirements

- Hardware: Basic PC with 4GB RAM, 100GB storage.
- Software: Windows/Linux, MySQL/SQLite, Python/Java/C#.

### 5. Optional Features

- Online reservations
- Email/SMS alerts
- Barcode support

# Assignment - 1

## 19. Perform a functional analysis for an online shopping system

### 1. Customer Functions

**Register/Login:** Create account, login, reset password

**Browse/Search:** View products by category, search with filters

**Product View:** See details (image, price, description, stock)

**Shopping Cart:** Add/remove/update items

**Checkout/Payment:** Choose address, pay securely

**Order Tracking:** Track status and view order history

**Reviews:** Rate and review purchased items

### 2. Admin Functions

**User Management:** View, block, or delete accounts

**Product Management:** Add/edit/delete products, manage stock

**Order Handling:** View, cancel, or update orders

**Reports:** View sales and activity reports

### 3. System Functions

**Notifications:** Email/SMS alerts for orders/offers

**Security:** Secure login and payment

**Responsive Design:** Works on mobile and desktop

## 20. Design a basic system architecture for a food delivery app.

### 1. User Interfaces

Customer App – Browse food, place orders, track delivery

- Delivery App – Accept deliveries, see routes, mark orders as delivered
- Restaurant App/Dashboard – Receive and prepare orders
- Admin Panel – Manage users, restaurants, and reports
- Core Features (App Logic)

### 2. Login/Signup

- Browse Menu & Add to Cart
- Place Order & Pay
- Track Delivery in Real Time Notifications (SMS, push)

# Assignment - 1

## 3. Backend

- User Data – Info about customers, delivery drivers, restaurants
- Order Data – Orders, status, payment, delivery info
- Menu Data – Restaurant menus and food details

## 21. Develop test cases for a simple calculator program

All Test cases are uploaded in a text file on GitHub.

## 22. Document a real-world case where a software application required critical maintenance.

On October 4, 2021, Facebook and its associated services — Instagram, WhatsApp, and Messenger — experienced a global outage lasting nearly 6 hours, affecting billions of users worldwide. This incident is one of the most prominent examples where critical maintenance was required to restore services.

### Root Cause

A faulty configuration change during routine maintenance of Facebook's backbone routers caused:

A disruption in network traffic between data centres.

A complete disconnection of Facebook's DNS (Domain Name System) from the internet.

As a result, not only did the platforms go down, but internal tools used by Facebook employees were also rendered inaccessible.

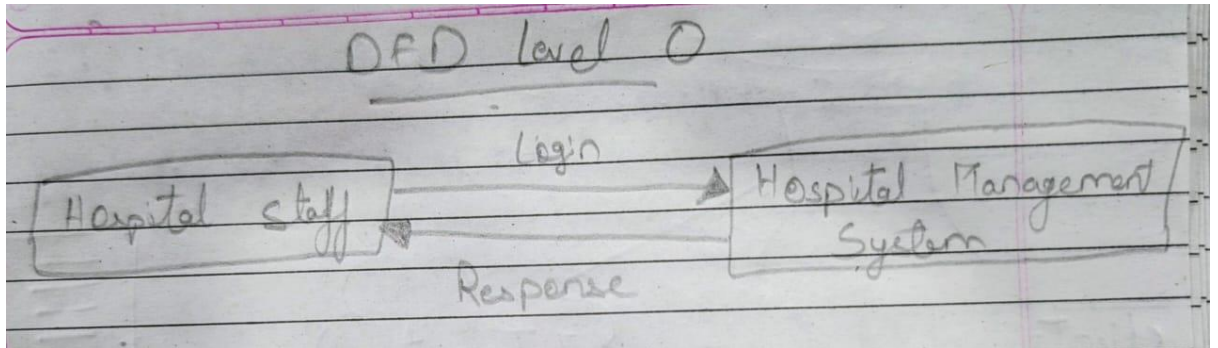
### Critical Maintenance Actions Taken

- On-Site Data Centre Access
- Engineers had to physically go to the data centres to manually restart and reconfigure servers
- This process was delayed due to security and access restrictions.
- Network Configuration Rollback
- The erroneous BGP (Border Gateway Protocol) routing information was corrected.
- DNS servers were gradually brought back online.
- Infrastructure Recovery
- Services were carefully brought back in stages to avoid further overload or instability

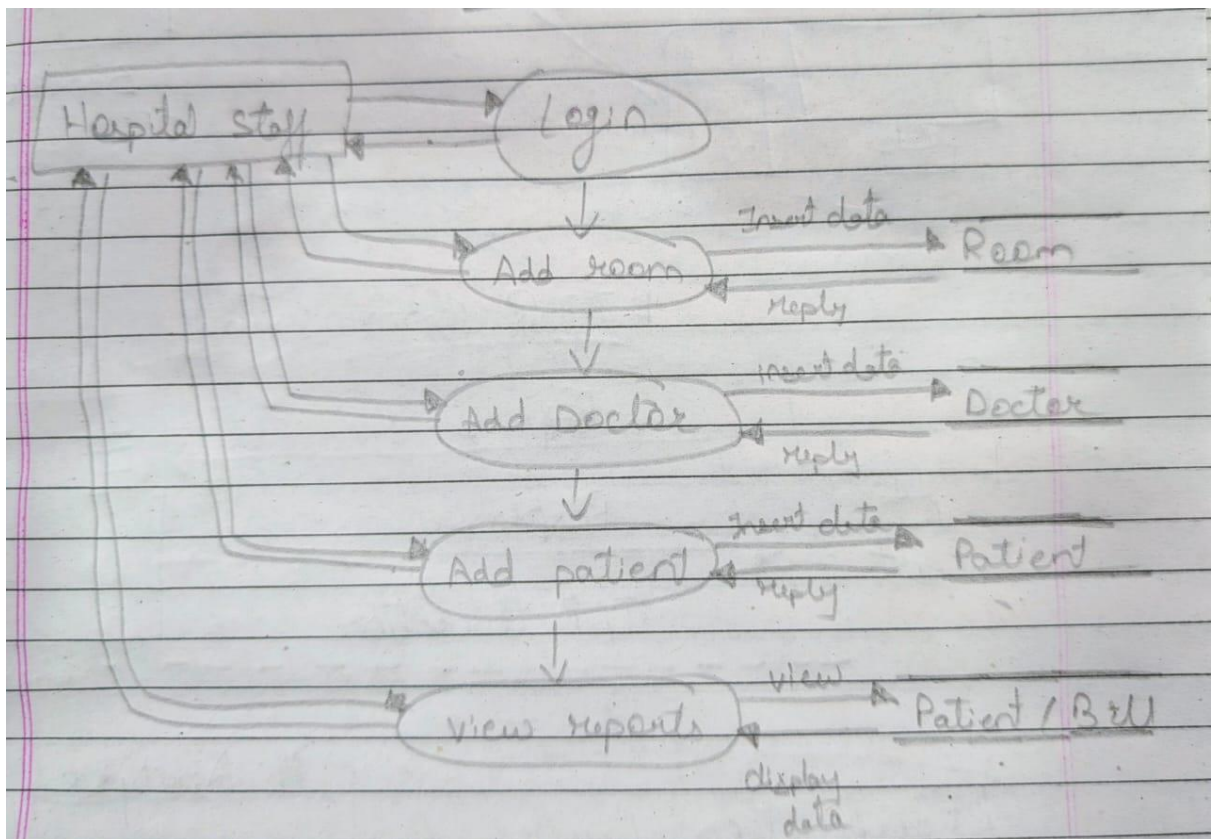
## Assignment - 1

23. Create a DFD for a hospital management system.

DFD Level 0



DFD Level 1





## Assignment - 1

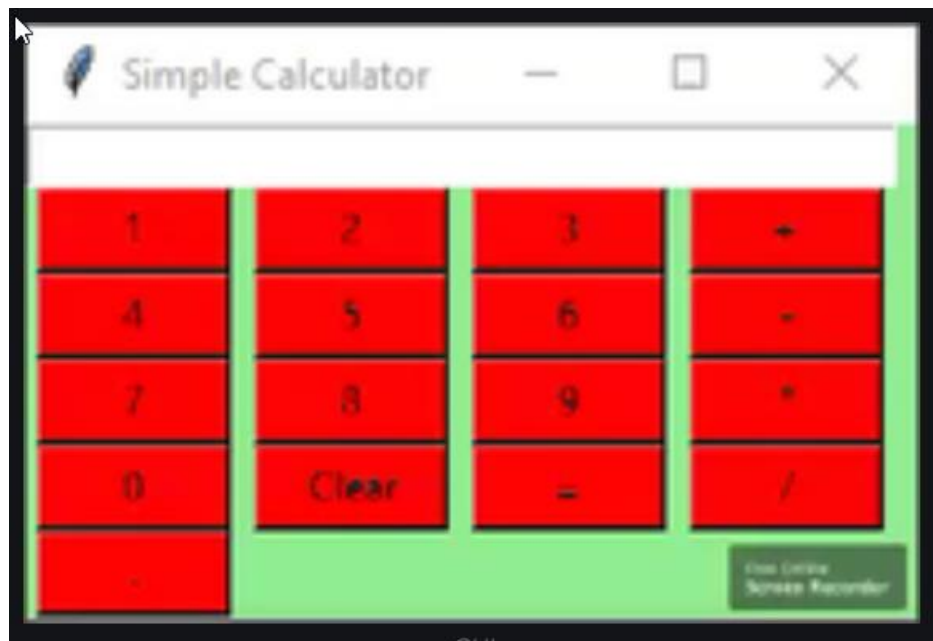
### 24. Build a simple desktop calculator application using a GUI library.

Python offers multiple options for developing a GUI (Graphical User Interface). Out of all the GUI methods, Tkinter is the most commonly used method. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with Tkinter outputs the fastest and easiest way to create GUI applications. Creating a GUI using Tkinter is an easy task.

To create a Tkinter:

- Importing the module – tkinter
- Create the main window (container)
- Add any number of widgets to the main window
- Apply the event Trigger on the widgets.

Implementation file is attached on git.



## Assignment - 1

25. Draw a flowchart representing the logic of an basic online registration system.

