Internship Report

# On

Java and Web development

# By

## DIPLOMA

**In**

COMPUTER SCIENCE

**MARCH– 2024**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTERS** | **PARTICULARS** | **PAGENO** |
| **CHAPTER-1** | **Company Profile** | **3-8** |
|  | * 1. About Organization | 3 |
|  | * 1. Vision and Mission of the Organization | 4 |
|  | * 1. Organization Structure | 5 |
|  | * 1. Roles and Responsibilities of personnel in the Organization | 6 |
|  | * 1. Products and Market Performance | 7 |
|  | * 1. Services | 7 |
| **CHAPTER -2** | **Programming Languages Used** | **9-33** |
|  | 2.1. HTML | 13 |
|  | 2.2. CSS | 15 |
|  | 2.3. JavaScript | 21 |
|  | 2.4. REACT JS | 22 |
|  | 2.5. Node JS | 23 |
|  | 2.6. SQL | 24 |
| **CHAPTER -3** | **Full Stack Web Development** | **26-31** |
|  | 3.1. Components of Full Stack Web Development | **26** |
|  | 3.2. Languages used by full stack developers | 28 |
|  | 3.3. Front end vs back-end vs full stack | 29 |
|  | 3.4. Advantages of Full stack development | 30 |
| **CHAPTER -4** | **PROJECT** | **32-35** |
|  | **6.1. Chat Application in Java** | 32 |
|  | 6.2. CONCLUSION | 35 |
| **CHAPTER -5** | **Conclusion** | **36** |
|  | REFERENCES | 37 |

**CHAPTER 1**

# COMPANY PROFILE

## About Organization

**EmbeddedFrU** is a group of Industry experienced individuals embedded with all the emerging technologies trying to bridge the gap between the professionals and the academics.

We are expertise in Software Development, Internship, Technical Workshops, Mock Tests and Placement training.

We have specialized teams and individuals working within a larger organization to provide support or expertise on specific projects or initiatives. These teams are typically focused on a particular function or area of expertise and are embedded within the larger organization to provide a more seamless and integrated approach to project delivery.

Our specialized teams and individuals work closely with the core teams of the organization, sharing knowledge, resources, and responsibilities to achieve a common goal. This can help to improve communication and collaboration between different parts of the organization, as well as enhance the overall efficiency and effectiveness of project delivery.

Our software development team provides specialized technical expertise to support the marketing or sales team in developing new products or services, or in optimizing existing ones.

Our organization structure is designed to promote greater collaboration and integration across different parts of an organization, enabling teams to work together more effectively to achieve common goals.

We aim at building strategic partnership with our customers and long-term relationship with our employees. We achieve this by

* Having a very strong focus on quality and productivity.
* Valuing time to market.
* Offering a very flexible operations model.
* Having a proactive work culture.
* Maintaining highest level of ethics.
* Being an equal-opportunity employer.

## 1.2. Vision and Mission of the Organization

**Vision:** Our vision is to help students upgrade their skills in the emerging fields and to develop innovative applications which make people lives better.

To empower individuals and organizations through high-quality training programs that enhance knowledge, skills, and performance, and ultimately drive success and growth.

A step towards making a skillful India.

**Mission:** Our mission is to provide comprehensive training programs that enable individuals and organizations to achieve their goals and objectives. We strive to deliver innovative, flexible, and customized training solutions that meet the unique needs of our clients. Our training programs are designed to enhance performance, increase efficiency, and foster a culture of continuous learning and development. We are committed to delivering exceptional customer service, utilizing the latest technologies and best practices, and maintaining the highest standards of quality and excellence in all that we do.

## 1.3. Organization Structure

Manager

Director

Manager

Developers

Design

Testers

Programmers

Testers

Trainees

Trainees

Finance

We participate in giving ideas for government startup contests to get sponsorships for the ideas of development.

## 1.4. Roles and Responsibilities of personnel in the Organization

1. Assess training needs: Identify training needs within the organization and develop training plans to address those needs.
2. Develop training materials: Create training materials, such as training manuals, job aids, and presentations, to support the learning process.
3. Deliver training: Deliver training to employees, either through in-person sessions, virtual training, or a combination of both.
4. Monitor training effectiveness: Measure the effectiveness of training programs by conducting evaluations and assessments to determine the effectiveness of the training and identify areas for improvement.
5. Collaborate with stakeholders: Work with key stakeholders to develop training programs that align with the organization's strategic goals and objectives.
6. Provide coaching and support: Provide coaching and support to employees to help them apply what they have learned in training to their job responsibilities.
7. Stay up-to-date on training trends: Keep up-to-date with the latest training trends, technologies, and best practices to ensure training programs are effective and relevant.
8. Document and report: Document all training activities and report on training outcomes and effectiveness to leadership and other stakeholders.
9. Design and develop applications: The primary responsibility of an application developer is to design, develop, test, and deploy software applications using programming languages, frameworks, and other software development tools.
10. Debug and troubleshoot issues: Debug and troubleshoot application issues and provide solutions to address them.

## 1.5. Products and Market Performance

* Application Software Development.
* Applications on IoT involving Arduino, Raspberry Pi , Intel's Edison and other hardware.
* Applications on IoT based on various platforms like Blynk, ThingSpeak, Google Cloud, Microsoft Azure, IBM Watson and many more.
* Advanced IoT applications involving automation, Smart cities, eHealth, Smart wearables, Remote monitoring, Agriculture, transportation.
* Applications involving various microcontrollers such as 8051, PIC, AVR, TI's MSP430, ARM and so on. Applications to support various interfaces of external modules like RF, Bluetooth, Wi-Fi, ZigBee, Ethernet, LoraWAN. Applications on Robotics technology.
* Game design and automation applications using Python.
* Web development involving technologies like HTML, CSS, JavaScript, Angular JS. App development using React Native and Ionic platforms.
* Applications based on deep learning which includes image recognition, medical diagnosis, training neural networks, Video Analysis and many more using MATLAB and Python.

Flexible, Agile and Efficient. That is our orientation and promise to the embedded marketplace. Over the years we have learned that alongside with the core embedded market demand of On Time Delivery, Quality and Cost efficiency, our customers expect their partners to be able to adjust into continuous change. That is how we develop and run our operations.

## 1.6. Services

###### IT Solutions and Services

EmbeddedFrU Technologies is a Bangalore based IT Training and Software Development center with an exclusive expertise in the area of IT Services and Solutions. EmbeddedFrU Technologies is also expertise in Web Designing and Consulting Services.

###### Embedded Design and Development

EmbeddedFrU Technologies has expertise in Design and development of embedded products and offers solutions and services in field of Electronics.

###### Academic Projects

EmbeddedFrU Technologies helps students in their academics by imparting industrial experience into projects to strive excellence of students. EmbeddedFrU Technologies encourages students to implement their own ideas to projects keeping in mind "A small seed sown upfront will be nourished to become a large tree one day”, thereby focusing the future entrepreneurs. They have a wide range of IEEE projects for B.E, MTech, MCA, BCA, DIPLOMA students for all branches in each and every domain.

###### Inplant Training

EmbeddedFrU Technologies provides Implant training for students according to the interest of students keeping in mind the current technology and academic benefit one obtains after completing the training. Students will be nourished and will be trained throughout with practical experience. Students will be exposed to industrial standards which boost their carrier. Students will become Acquaint to various structural partitions such as labs, workshops, assembly units, stores, and administrative unit and machinery units. They help students to understand their functions, applications and maintenance. Students will be trained from initial stage that is from collection of Project Requirements, Project Planning, Designing, implementation, testing, deployment and maintenance there by helping to understand the business model of the industry. Entire project life cycle will be demonstrated with hands on experience. Students will also be trained about management skills and team building activities. They assure that by end of implant training students will Enhance communication skills and acquire technical skills, employability skills, start-up skills, and will be aware of risks in industry, management skills and many other skills which are helpful to professional engagement.

###### Software Courses

EmbeddedFrU Technologies provides courses for students according to the interest of students keeping in mind the current technology and assist them for their further Employment. Company provides various courses such as C, C++, VB, DBMS, Dot Net, Core Java, J2EE, IoT, Machine Learning, Embedded Systems along with live projects.

# CHAPTER 2

# PROGRAMMING LANGUAGES FOR WEB DEVELOPMENT

## 2.1. HTML

HTML is an acronym which stands for Hyper Text Markup Language which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

Hyper Text:HyperText simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. HyperText is a way to link two or more web pages (HTML documents) with each other.

Markup language: A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

Web Page: A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. With the help of HTML only, we can create static web pages.

Hence, HTML is a markup language which is used for creating attractive web pages with the help of styling, and which looks in a nice format on a web browser. An HTML document is made of many HTML tags and each HTML tag contains different content.

**Let's see a simple example of HTML.**

1. <!DOCTYPE>

2. <html>

3. <head>

4. <title>Web page title</title>

5. </head>

6. <body>

7. <h1>Write Your First Heading</h1>

8. <p>Write Your First Paragraph.</p>

9. </body>

10. </html>

**Description of HTML**

**<!DOCTYPE>**: It defines the document type or it instruct the browser about the version of

HTML.

**<html >** :This tag informs the browser that it is an HTML document. Text between html tag

describes the web document. It is a container for all other elements of HTML except

<!DOCTYPE>

**<head>**: It should be the first element inside the <html> element, which contains the

metadata(information about the document). It must be closed before the body tag opens.

**<title>:** As its name suggested, it is used to add title of that HTML page which appears at the

top of the browser window. It must be placed inside the head tag and should close

immediately. (Optional)

**<body>:** Text between body tag describes the body content of the page that is visible to the

end user. This tag contains the main content of the HTML document.

**<h1> :** Text between <h1> tag describes the first level heading of the webpage.

**<p>:** Text between <p> tag describes the paragraph of the webpage.

**Features of HTML**

1) It is a very easy and simple language. It can be easily understood and modified.

2) It is very easy to make an effective presentation with HTML because it has a lot of

formatting tags.

3) It is a markup language, so it provides a flexible way to design web pages along with the

text.

4) It facilitates programmers to add a link on the web pages (by html anchor tag), so it

enhances the interest of browsing of the user.

5) It is platform-independent because it can be displayed on any platform like Windows,

Linux, and Macintosh, etc.

6) It facilitates the programmer to add Graphics, Videos, and Sound to the web pages which

makes it more attractive and interactive.

7) HTML is a case-insensitive language, which means we can use tags either in lower-case or

upper-case.

**HTML text Editors**

* An HTML file is a text file, so to create an HTML file we can use any text editors.
* Text editors are the programs which allow editing in a written text, hence to create a

web page we need to write our code in some text editor.

* There are various types of text editors available which you can directly download, but

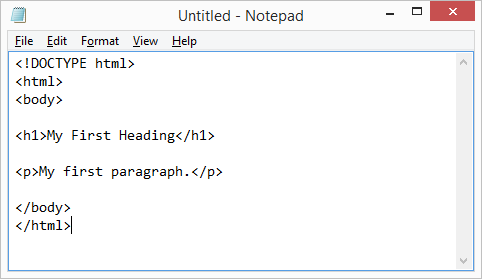
for a beginner, the best text editor is Notepad (Windows) or TextEdit (Mac).

* After learning the basics, you can easily use other professional text editors which are,

Notepad++, Sublime Text, Vim, etc.

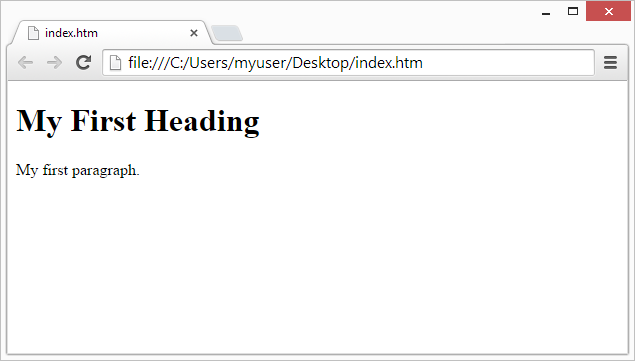
**Creating first Web page in Notepad(Windows)**

Below is the typing of HTML code into Notepad:

****

Open the saved HTML file in your favorite browser (double click on the file, or right-click - and choose "Open with").

The result will look much like this:



**Building blocks of HTML**

An HTML document consist of its basic building blocks which are:

Tags: An HTML tag surrounds the content and apply meaning to it. It is written

between < and > brackets.

Attribute: An attribute in HTML provides extra information about the element, and it

is applied within the start tag. An HTML attribute contains two fields: name & value.

**Syntax**

1. <tag name attribute\_name= " attr\_value"> content </ tag name>

Elements: An HTML element is an individual component of an HTML file. In an

HTML file, everything written within tags are termed as HTML elements.

## 2.2. CSS

**Designing Web Pages with CSS**

CSS is a style language for controlling how certain Hypertext Markup Language (HTML) elements of a web page are visualized in the frontend. For example, we can use CSS to specify the font in which a text should be displayed and its color. We can influence the appearance of lists, form elements, and tables, for example, setting the bullet points in a list or determining the background color of individual table cells. So, we have many options to make HTML, which looks rather dull by default more attractive and appealing.

**The Principle of CSS**

We can define how the content of certain HTML elements should be displayed using CSS rules. These rules basically consist of two parts, as shown in Figure 2.1.

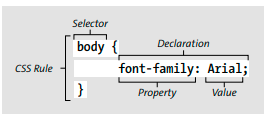


Figure 2.1. The Structure of CSS Rules

The CSS selector enables you to specify which HTML elements should be subjected to the respective CSS rule. We can use the CSS declaration written in curly brackets to specify how exactly these HTML elements should be displayed. Declarations, in turn, consist of a CSS property and a CSS value, both separated by a colon and ending together with a semicolon.

For example, properties can affect the color, font, dimensions, or border color of an element. Using the value of the property, you can then specify, for example, which color, which font, which width or height, or which frame color should be selected. Each property has certain predefined values that are valid.

First of all, for illustration purposes, let’s consider a simple CSS rule.

h1 {

font-family: Arial;

color: darkblue;

}

This CSS rule states that all first-level headings (defined by the h1 selector) should use

the “Arial” font (defined by the font-family property) and be displayed in dark blue

(defined by the color property).

**Including CSS in HTML**

In total, three different ways exist for defining CSS rules and including them in an HTML document:

* **External stylesheets (external CSS) :** In this case, you save the CSS instructions as a separate CSS file (with file extension .css) and include this file in the HTML document.
* **Internal stylesheets (internal CSS) :** In this case, you define the CSS instructions in the header of the HTML document within the <style> element.
* **Inline styles (inline CSS) :** In this case, you specify the CSS instructions directly in an HTML element.

**Including External CSS Files (External CSS)**

Let’s start with the variant that makes the most sense in most cases: the inclusion of separate CSS files (external CSS). The reason why this variant is often the most useful is you can cleanly separate the CSS code from the HTML code and thus easily include it in multiple HTML documents. Thus, web projects can have one central CSS file (or a few files) that can then be

included in all HTML documents in the project. This approach ensures that each HTML document uses the same CSS instructions and that the presentation of each web page is consistent. This structure also has a considerable advantage with regard to modifications: When we make style changes, you only need to make them once centrally, in the shared CSS files, which will affect all HTML documents, as shown in Figure 2.2.

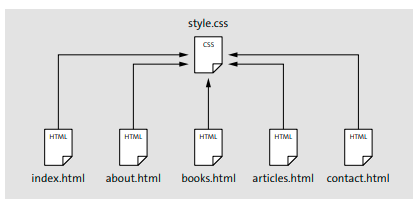


Figure 2.2. Reusing External CSS Files in Different HTML Files

CSS files are ordinary text documents that you can create and edit in any text editor with the .css file extension.

When we now load the HTML file, the browser finds the specified CSS file and applies the CSS rules to the corresponding HTML elements.

## 2.3. JavaScript

## JavaScript was designed with a purpose to make web pages dynamic and more interactive.

## JavaScript is one of the popular scripting languages having following features :

## (a) It can be used for client and server applications.

## (b) It is platform independent which means it can run on any operating systems (i.e. Linux, Microsoft Windows, Mac OS X etc.).

## (c) JavaScript codes are needed to be embedded or referenced into HTML documents then only it can run on a web browser.

## (d) It is an interpreted language.

## (e) It is a case-sensitive language and its keywords are in lowercase only

## JavaScript is not a full-fledged language and it needs to be embedded within a HTML document. Otherwise, or to specify an external file that contains the JavaScript code we can use word ‘script’ or ‘program’ for the JavaScript code.

## The most common way to set off a script is to use the HTML <script> and </script> tags in HTML document. We can place our JavaScript code in either the HEAD or BODY section of a HTML document.

## The Syntax (General format) is

## <SCRIPT [Attributes = [“Value”] …. ]> Indicates starting of

## JavaScript Code

## … JavaScript statement(s);

## </SCRIPT> Indicates ending of

## JavaScript Code

## Placing the Javascript Code

## There are two ways to place the JavaScript code :

## 1. Embedded/Inline JavaScript : JavaScript code can be placed either in the HEAD or in the BODY section of a HTML document.

## a. It is advised to place JavaScript code in HEAD section when it is required to be used more than once.

## b. If the JavaScript code is small in size and used only once, it is advisable to put it in the BODY section of the HTML document.

## 2. External JavaScript : In case, same JavaScript code needs to be used in multiple documents then it is the best approach to place JavaScript code in external files having extension as “ .js”. To do

## so, we will use src attribute in <SCRIPT> tag to indicate the link for the source JavaScript file.

### Statements In Javascript

Statements are the commands or instructions given to the JavaScript interpreter to take some actions as directed. A JavaScript interpreter resides within almost Internet browsers. A collection of statements to accomplish a job, is called as a script or program. The JavaScript statements will be as follows :

a = 100; // stores value 100 in variable a

b = 200; // stores value 200 in variable b

c = a + b; // stores the sum of a and b in variable c

document.write

(“Sum of A and B : “); // displays the string

document.write(c); // displays the value of c

In JavaScript, semicolon(;) is used to end a statement but if two statements are written in separate lines then semicolon can be omitted.

**Variables**

A variable is an identifier that can store values. These values can be changed during the execution of script. Once a value is stored in a variable it can be accessed using the variable name. Variable

declaration is not compulsory, though it is a good practice to use variable declaration. Keyword var is used to declare a variable.

**Syntax**

var var-name [= value] [..., var-name [= value] ]

**Data Types**

JavaScript supports three basic data types – number, string, boolean

and two composite data types – arrays and objects.

1. **Number**

The number variable holds any type of number, either an integer or a real number. Some examples of numbers are:

29, -43, 3.40, 3.4323

1. **String**

A string is a collection of letters, digits, punctuation characters, and so on. A string literal is enclosed within single quotes or double quotes (‘or “). Examples of string literals are:

‘welcome’, “7.86” , “wouldn’t you exit now”, ‘ country=”India” ’

1. **Boolean Values**

A boolean variable can store only two possible values, either true or false. Internally it is stored as 1 for true and 0 for false. It is used to get the output of conditions, whether a condition results in true or false.

Example

x == 100; // results true if x=100 otherwise false.

1. **Arrays**

An array is a collection of data values of same types having a common name. Each data element in array is referenced by its position in the array also called its index number. Individual array elements can be referenced by the array name followed by the pair of square

brackets having its index number. The index number starts with zero in JavaScript i.e. the first element in JavaScript has it index value as 0, second has its index value as 1 and so on. An array can be declared in any of the following ways :

var a = new a( );

**Objects**

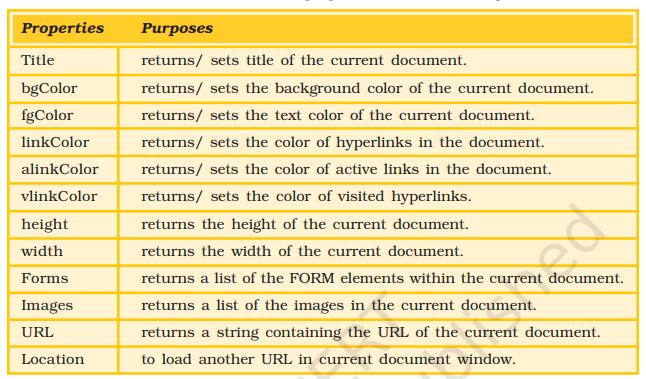
JavaScript is an object based scripting language. It allows us to define our own objects and make our own variable types. It also offers a set of predefined objects. The tables, forms, buttons, images, or links on our web page are examples of objects. The values associated with object are properties and the actions that can performon objects are methods or behaviour. Property associated to an object can be accessed as follows:

ObjectName.PropertyName

## Document Object

## The Document object is one of the parts of the Window object. It can be accessed through the window.document property. The document object represents a HTML document and it allows one to access all the elements in a HTML document.

## Some of the common properties of document object are:



**Date Object**

This object is used to set and manipulate date and time. JavaScript dates are stored as the number of milliseconds since midnight, January 1, 1970. This date is called the epoch. Dates before 1970 are represented by negative numbers. A date object can be created by using the new

keyword with Date().

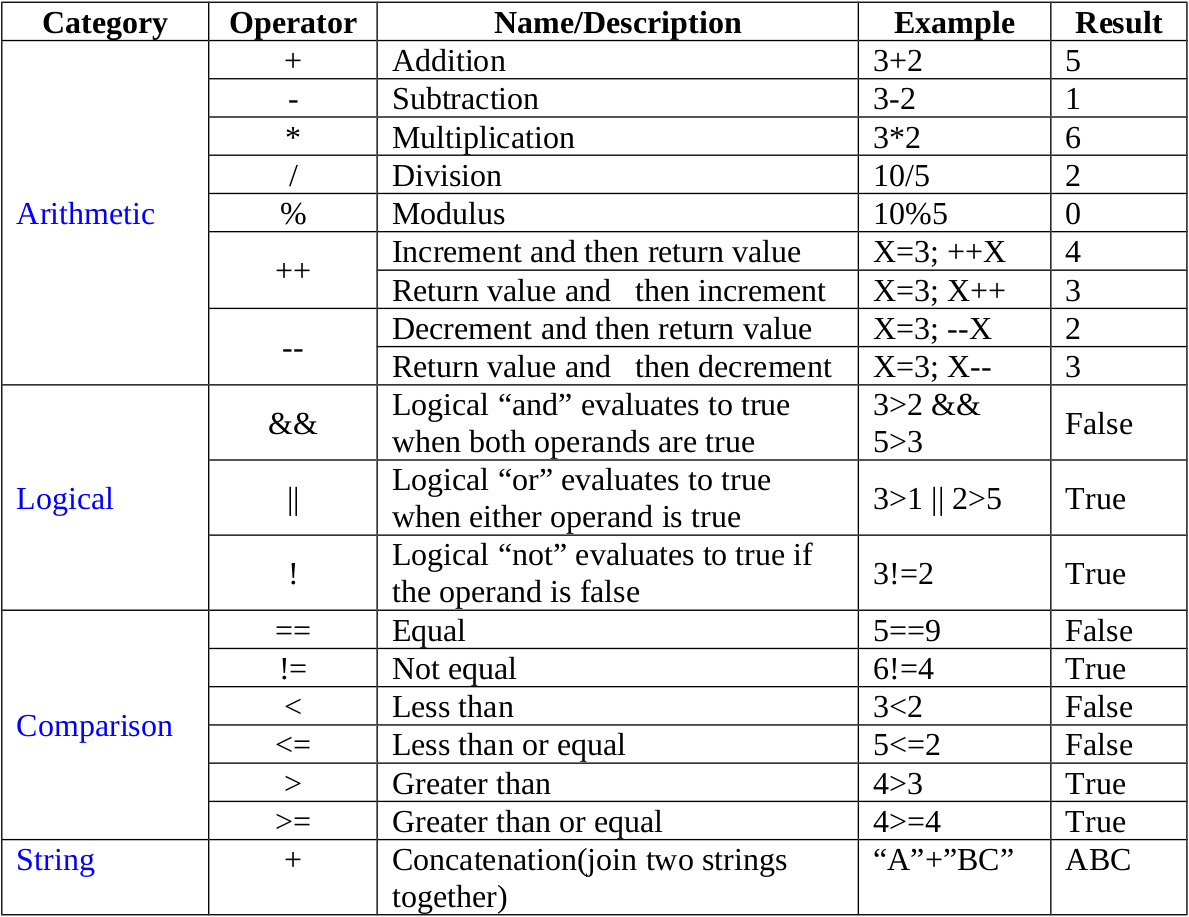
**Math Object**

This object contains methods and constants to carry more complex mathematical operations. This object cannot be instantiated like other objects. All properties and methods of Math are static. We can refer to the constant p as Math.PI and the sine function as Math.sin(x), where x is the method’s argument.

**Expressions And Operators**

An expression is a combination of operators operands that can be evaluated. It may also include function calls which return values.

Below table shows the different operators used in JavaScript:



## JavaScript Popup Boxes (Dialog Boxes)

## Alert Box

## Alert( ) method of window object creates a small dialog box with a short text message and “OK” command button called alert box. Alert box contains an icon indicating a warning.

## Syntax:

## [window].alert(“Text to be displayed on the popup box”);

## The word window is optional.

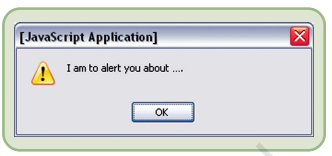
**Example:**

window.alert(“I am to alert you about ….”);

or

alert(“I am to alert you about ….”);

**Output:**



**Confirm Box**

Confirm box is used if we want the user to verify and confirm the information. The user will have to click either “OK” or “Cancel” buttons.

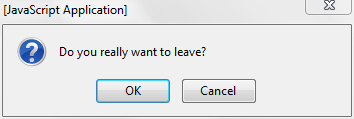
**Syntax:**

[window].confirm(“Text to be confirmed”);

**Example:**

confirm(“Do you really want to leave?”);

**Output:**



**Prompt Box**

Prompt box allows getting input from the user. We can specify the default text for the text field. The information submitted by the user from prompt( ) can be stored in a variable.

**Syntax:**

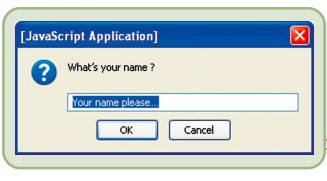
prompt(“Message” [, “default value in the text field”]);

**Example:**

var name = prompt(“What’s your ↵

name? “, “Your name please…”);

**Output:**



## 2.4. REACT JS

## ReactJS is a declarative, efficient, and flexible JavaScript library for building reusable UI components. It is an open-source, component-based front end library responsible only for the view layer of the application. It was created by Jordan Walke, who was a software engineer at Facebook. It was initially developed and maintained by Facebook and was later used in its products like WhatsApp & Instagram. Facebook developed ReactJS in 2011 in its newsfeed section, but it was released to the public in the month of May 2013.

## A ReactJS application is made up of multiple components, each component responsible for outputting a small, reusable piece of HTML code. The components are the heart of all React applications. These Components can be nested with other components to allow complex applications to be built of simple building blocks. ReactJS uses virtual DOM based mechanism to fill data in HTML DOM. The virtual DOM works fast as it only changes individual DOM elements instead of reloading complete DOM every time.

## To create React app, we write React components that correspond to various elements. We organize these components inside higher level components which define the application structure. For example, we take a form that consists of many elements like input fields, labels, or buttons. We can write each element of the form as React components, and then we combine it into a higher-level component, i.e., the form component itself. The form components would specify the structure of the form along with elements inside of it.

## 2.5. Node JS

Node.js”is an open-source, cross-stage, JavaScript runtime condition that executes JavaScript code

outside of an internet browser. Node.js lets engineers use JavaScript to compose order line devices

and for server-side scripting—running contents server-side to deliver dynamic website page

content before the page is sent to the client's internet browser. Subsequently, Node.js speaks to a

"JavaScript all over the place" worldview,” bringing together web-application improvement around a solitary programming language, as opposed to various dialects for server-and customer side contents.

“However .js is the standard filename expansion for JavaScript code, the name "Node.js" doesn't

allude to a specific document in this unique circumstance and is only the name of the item. Node.js

has an occasion driven design equipped for offbeat I/O. These plan decisions mean to upgrade

throughput and adaptability in web applications with many”info/yield tasks, just as for continuous

Web applications (e.g., ongoing correspondence projects and program games).

The Node.js disseminated advancement venture was recently represented by the Node.js

Foundation, and has now converged with the JS Foundation to frame the OpenJS Foundation,

which is encouraged by the Linux Foundation's Collaborative Projects program

## 2.6. SQL

## SQL (Structured Query Language) is used to perform operations on the records stored in the database, such as updating records, inserting records, deleting records, creating and modifying database tables, views, etc.

## SQL is not a database system, but it is a query language.

## Suppose you want to perform the queries of SQL language on the stored data in the database. You are required to install any database management system in your systems, for example, Oracle, MySQL, MongoDB, PostgreSQL, SQL Server, DB2, etc.

SQL is a short-form of the structured query language, and it is pronounced as S-Q-L or

sometimes as See-Quell.

This database language is mainly designed for maintaining the data in relational database

management systems. It is a special tool used by data professionals for handling structured

data (data which is stored in the form of tables). It is also designed for stream processing in

RDSMS.

You can easily create and manipulate the database, access and modify the table rows and

columns, etc. This query language became the standard of ANSI in the year of 1986 and ISO

in the year of 1987.

If you want to get a job in the field of data science, then it is the most important query

language to learn. Big enterprises like Facebook, Instagram, and LinkedIn, use SQL for

storing the data in the back-end.

**Features of SQL**

* The basic use of SQL for data professionals and SQL users is to insert, update, and delete the

data from the relational database.

* SQL allows the data professionals and users to retrieve the data from the relational database

management systems.

* It also helps them to describe the structured data.
* It allows SQL users to create, drop, and manipulate the database and its tables.
* It also helps in creating the view, stored procedure, and functions in the relational database.
* It allows you to define the data and modify that stored data in the relational database.
* It also allows SQL users to set the permissions or constraints on table columns, views, and

stored procedures.

**Process of SQL**

When we are executing the command of SQL on any Relational database management

system, then the system automatically finds the best routine to carry out our request, and the

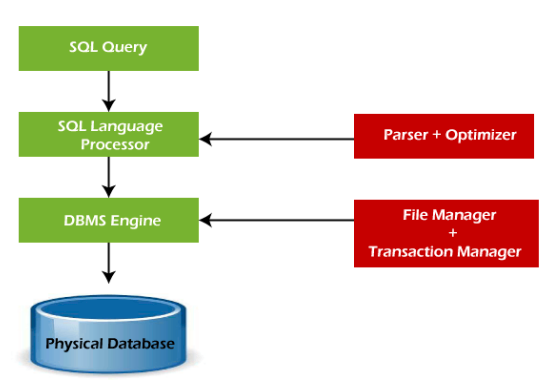
SQL engine determines how to interpret that particular command.

Structured Query Language contains the following four components in its process:

* Query Dispatcher
* Optimization Engines
* Classic Query Engine
* SQL Query Engine, etc.

A classic query engine allows data professionals and users to maintain non-SQL queries. The

architecture of SQL is shown in the following diagram:



**Some SQL Commands**

The SQL commands help in creating and managing the database. The most common SQL

commands which are highly used are mentioned below:

1. CREATE command

2. UPDATE command

3. DELETE command

4. SELECT command

5. DROP command

6. INSERT command

**CREATE Command**

This command helps in creating the new database, new table, table view, and other objects of

the database.

**UPDATE Command**

This command helps in updating or changing the stored data in the database.

**DELETE Command**

This command helps in removing or erasing the saved records from the database tables. It

erases single or multiple tuples from the tables of the database.

**SELECT Command**

This command helps in accessing the single or multiple rows from one or multiple tables of

the database. We can also use this command with the WHERE clause.

**DROP Command**

This command helps in deleting the entire table, table view, and other objects from the

database.

**INSERT Command**

This command helps in inserting the data or records into the database tables. We can easily

insert the records in single as well as multiple rows of the table.

**Advantages of SQL**

SQL provides various advantages which make it more popular in the field of data science. It

is a perfect query language which allows data professionals and users to communicate with

the database. Following are the best advantages or benefits of Structured Query Language:

**1. No programming needed**

SQL does not require a large number of coding lines for managing the database systems. We

can easily access and maintain the database by using simple SQL syntactical rules. These

simple rules make the SQL user-friendly.

**2. High-Speed Query Processing**

A large amount of data is accessed quickly and efficiently from the database by using SQL

queries. Insertion, deletion, and updation operations on data are also performed in less time.

**3. Standardized Language**

SQL follows the long-established standards of ISO and ANSI, which offer a uniform

platform across the globe to all its users.

**4. Portability**

The structured query language can be easily used in desktop computers, laptops, tablets, and

even smartphones. It can also be used with other applications according to the user's

requirements.

**5. Interactive language**

We can easily learn and understand the SQL language. We can also use this language for

communicating with the database because it is a simple query language. This language is also

used for receiving the answers to complex queries in a few seconds.

**6. More than one Data View**

The SQL language also helps in making the multiple views of the database structure for the

different database users.

**Disadvantages of SQL**

With the advantages of SQL, it also has some disadvantages, which are as follows:

**1. Cost**

The operation cost of some SQL versions is high. That's why some programmers cannot use

the Structured Query Language.

**2. Interface is Complex**

Another big disadvantage is that the interface of Structured query language is difficult, which

makes it difficult for SQL users to use and manage it.

**3. Partial Database control**

The business rules are hidden. So, the data professionals and users who are using this query

## language cannot have full database control.

# CHAPTER 3

# Full Stack Web Development

## 3.1. Components of Full Stack Web Development

Full stack development refers to the end-to-end application software development, including the front end and back end. The front end consists of the user interface, and the back end takes care of the business logic and application workflows.Full Stack Web Development alludes to the development of both front end(client side) and back end(server side) portion of web application.

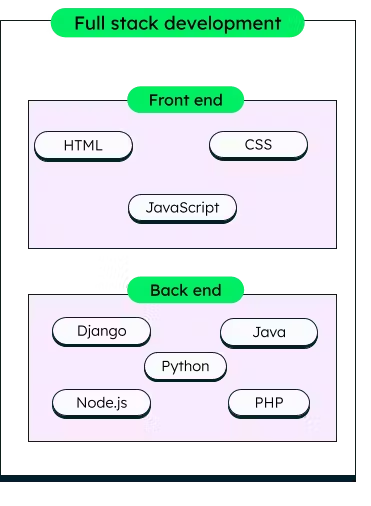
” Full stack”web engineers can configuration complete web application and sites. They take a shot at the frontend, backend,”database and maintenance of web application or sites.

“Front end is the noticeable piece of site or web application which is liable for client experience. The client straightforwardly interfaces with the front end segment of the web application or site.

” Back end alludes“to the server-side portion of web application or site with an essential spotlight on how the site functions. It is answerable for dealing with the database through questions” and APIs by customer side orders. This sort of site primarily comprises of three sections front end, back end, and database.

“Database is the assortment of between related information which helps in productive recovery, addition and erasure of information from database and sorts out the information as tables, sees, patterns, reports and so on.”

The main components of a full stack development are the front-end, back-end and database

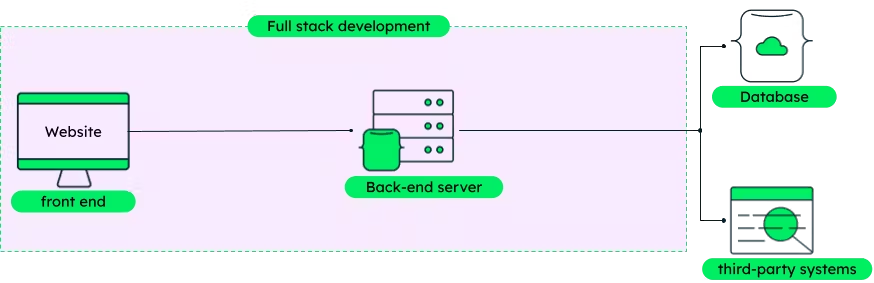
****

**Figure 3.1. Main components of full stack Development**

Consider a retail website. Users can browse or purchase specific items, delete or add items in cart, change their profile, and do many other things. All these actions require a front-end user interface (UI), as well as some business logic, written in the back-end.

* The website UI can be built using various, front-end technologies like HTML, CSS, Javascript.
* The back end is written in programming languages like Java or Python. Further, a good web application would need scalability, event handling, and routing, which are usually handled by libraries and frameworks like SpringBoot or Django.
* The back end also consists of logic that can connect the application to other services and databases. For example, all the user and transaction data is stored in a database through specific drivers handled on the back end.

A full stack developer is one who can single-handedly implement both the front-end and back-end workflows, like placing the order or changing the user profile.



**Figure 3.2. Demonstration of full stack development in an end-to-end workflow**

Full stack developers must have knowledge of an entire [technology stack](https://www.mongodb.com/basics/technology-stack), i.e., the set of technologies that are used to build an end-to-end application quickly and efficiently. For example, if they want to build an application using the [MEAN stack](https://www.mongodb.com/mean-stack), they should know how to work with [MongoDB](https://www.mongodb.com/atlas/database), Express, Angular and Node.

Full stack developers should be able to judge whether the selected technologies are the right choice for their project during the early phases. Some responsibilities of a full stack developer are to:

* Help in choosing the right technologies for the project development and testing both on the front end and the back end.
* Write clean code across the stack by following the best practices of the tools used.
* Be up to date with the latest technologies and tools to make the best technology usage decisions.

## 3.2. Languages used by full stack developers

Full stack developers are free to use any set of languages that are compatible with each other and the overall application framework. JavaScript is a popular language often used by full-stack developers as it’s one of the very few languages that can be used both on the front end and back end. Companies will most likely hire a full stack developer for smaller or medium-size projects. Some popular languages are:

* Front end: HTML, CSS, JavaScript.
* Back end: Python, Java, R, Ruby, Node.js, PHP.

It’s also a popular and convenient practice to use full technology stacks like [MEAN stack](https://www.mongodb.com/mean-stack), [MERN stack](https://www.mongodb.com/mern-stack), Ruby on Rails, and LAMP for faster and more efficient development, and an easier learning curve.

## 3.3. Front end vs back end vs full stack

Applications that require higher scalability and more complex workflows require broader skill sets and collaboration across teams. For example, the front end may be handled by the UI team, and the back end by another team. In some organizations, individuals will be required to work on both the front-end and back-end implementation of a feature. This is where full stack developers would come into play.

### Front-end developers

These developers handle the UI of a web application (or website)—for example, visual effects, frames, navigation, and forms. They focus mainly on user experience and use HTML, CSS, and JavaScript as programming languages.

### Back-end developers

They deal with the business logic, security, performance, scalability, and handling request-response of the application. They create or use frameworks to design the core application workflows and use technologies like JavaScript, Python, Java, and .NET.

### Full stack developers

They are responsible for coding end-to-end workflows by using both front-end and back-end technologies. [MERN stack](https://www.mongodb.com/languages/mern-stack-tutorial) and [MEAN stack](https://www.mongodb.com/languages/mean-stack-tutorial) are examples of JavaScript-based [technology stacks](https://www.mongodb.com/basics/technology-stack) that full stack developers can use to build end-to-end applications.

## 3.4. Advantages of Full stack development

There are many advantages of hiring full stack developers for web application development:

* Complete ownership and understanding of the project
* Saves both project time and cost, and enhances productivity
* Faster bug fixing due to knowledge of complete system
* Easy knowledge transfer to other team members
* Better division of work amongst team members

# CHAPTER 4 PROJECT

# 4.1.1 Group Chat Application in Java

This group chat application is developed using Java Programming Language.

Chat applications allow users to communicate with each other that should be in real-time. Where the messages are getting exchanged in real-time. It can run multiple client-server chats easily. Chat applications can be used for various purposes such as connecting with friends and using the application for business inquiries.

In this group chat application, we are using MulticastSocket class. A MulticastSocket is a (UDP) Datagram Socket, with additional capabilities for joining “groups” of other multicast hosts on the internet.

The multicast datagram socket class is useful for sending and receiving IP multicast packets. A MulticastSocket is a (UDP) DatagramSocket, with additional capabilities for joining "groups" of other multicast hosts on the internet. This class is used for sending and receiving multicast IP packets. It extends DatagramSocket class and provides additional functionality for joining groups. A message sent to the group IP address will be received by all the clients who have joined the group. This must be kept in mind that for sending packets to the group, datagram socket doesn’t have to join the group but for receiving the packets addressed to the group, it must join the group. This class provides various methods for controlling the flow of multicast packets like setting the ttl, network interface to use etc, along with the major functions of joining and leaving a group.

A multicast group is specified by a class D IP address and by a standard UDP port number. Class D IP addresses are in the range 224.0.0.0 to 239.255.255.255, inclusive. The address 224.0.0.0 is reserved and should not be used.

One would join a multicast group by first creating a MulticastSocket with the desired port, then invoking the joinGroup(InetAddress groupAddr) method.

**SNAPSHOTS OR OUTPUT**

**Group Chat Application**

1. The java file is compiled using javac GroupChat.java
2. Then the java file is executed using java GroupChat 239.0.0.0 1234

239.0.0.0 is the IP address and the port number is 1234

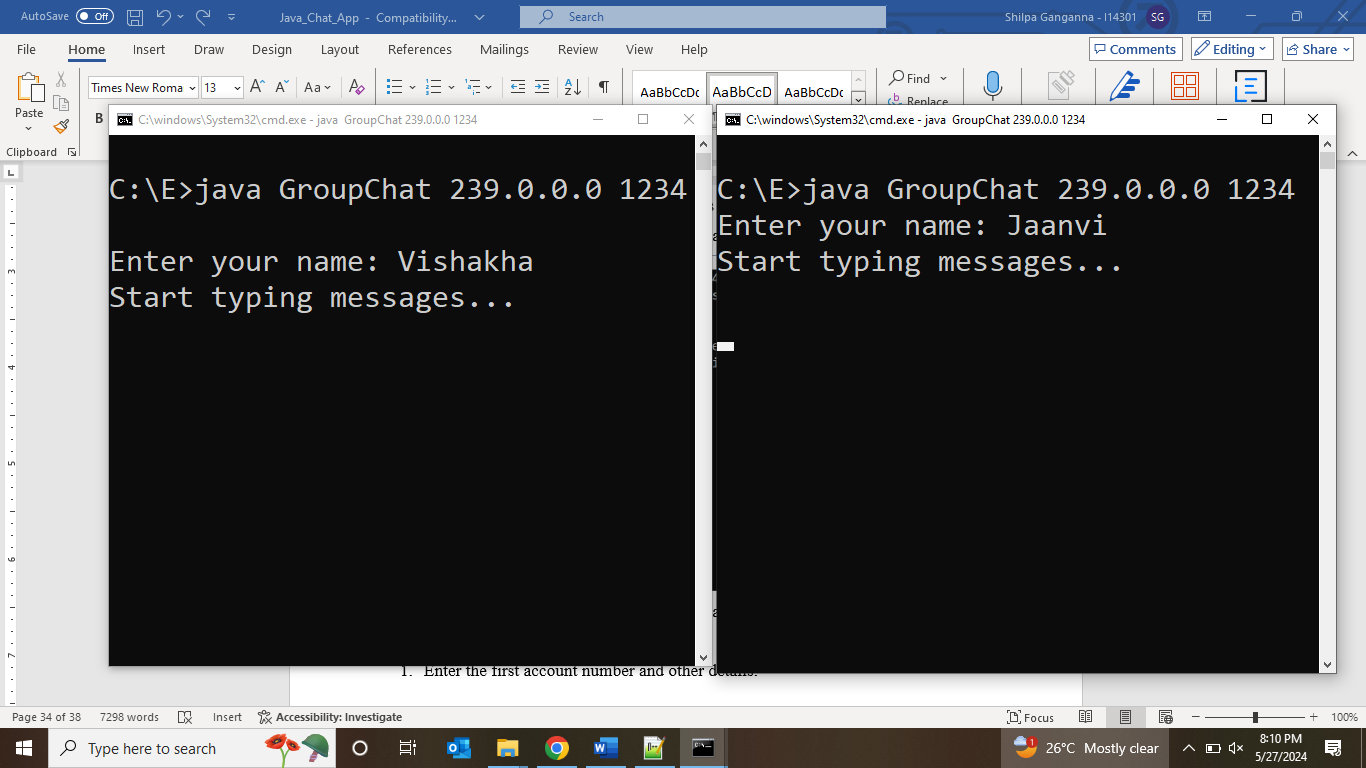
We can start more than one chat windows using this command.

1. The output window looks like below and asks to enter your name.

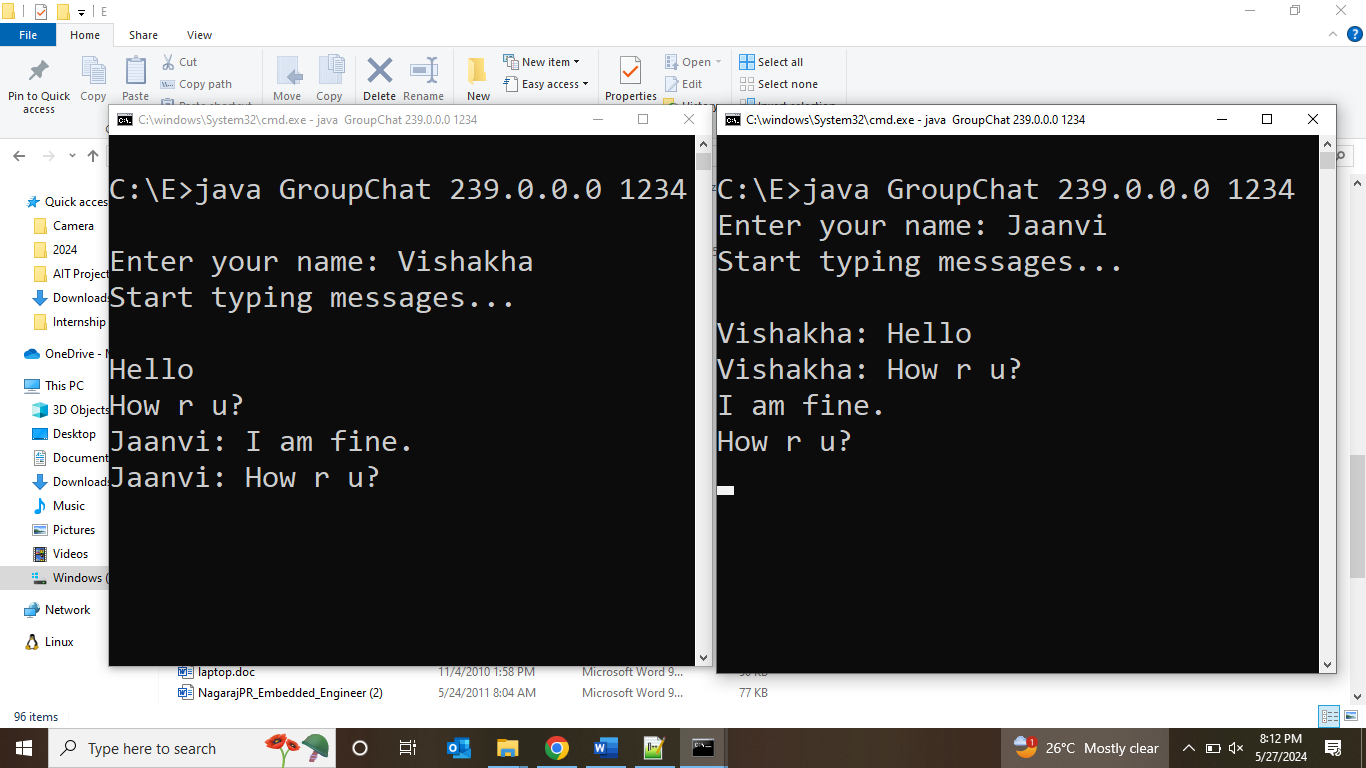
A screenshot of a computer

Description automatically generated

1. We will open another window using the same command and give a different name.



1. Now we will start sending messages from one window to another and it looks like below:



1. We can add as many people as possible for this group chat using the same command. Whenever the message is sent, it will be broadcasted to all the people.

## 6.1.2. Conclusion

The above program is executed on a single machine. Socket programming is meant for distributed programming. The same piece of code snippet when present on different machines which have Java installed can satisfy that requirement. This is just the bare bones service logic. The project would be even more fascinating if the front-end is developed. We can use Java’s AWT (Abstract Window Toolkit) or its advanced counterpart, Java Swing to develop the front end.

We can incorporate network security feature by performing encryption before sending the message over the network.

Primitive techniques such as Caesar cipher or advanced methods such as RSA can be used to perform encryption-decryption. If the primary objective is efficiency, then Socket programming is the best choice. Since it doesn’t require any run time support, it is a bit faster compared to RMI(Remote Method Invocation)

**CHAPTER 5**

## Conclusion

Full stack development provides a faster and more efficient development experience to build web applications, as team members are equipped with the knowledge of various technologies and tools and can use their diverse knowledge for development as well as future research.

“The main objective of the internship is to provide an opportunity to undergraduates to identify, observe and practice how engineering is applicable in the real industry. It is not only to get experience on technical practices but also to observe management practices and to interact with fellow workers.

“The internship that I received at EmbeddedFrU was great experience for me not only on technical terms but also in “terms of interaction with other employees. I learnt a great deal on applying the knowledge I have gained at the College. Learning something from books and lectures Is nothing ”like having firsthand experience. I got to apply my Programming knowledge. The company gives the full freedom to the interns to get a sufficient exposure. The interns are allowed to perform tasks relevant to the internship. Engineers are always very enthusiastic on giving us the best training Page experience. The ”employees do not think of their designation but only on the work to be done. Teamwork has been excellent. Everyone was friendly from top to bottom. Employees are busy with their work but they always help us to improve our knowledge.”

# REFERENCES

[1] "Head First Java" by Kathy Sierra and Bert Bates: A beginner-friendly book that covers Java fundamentals in an engaging and interactive manner.

[2] "Effective Java" by Joshua Bloch: Provides best practices and advanced techniques for writing efficient and maintainable Java code.

[3] "Java: The Complete Reference" by Herbert Schildt: Comprehensive guide covering all aspects of Java programming language.

[4] “Java Programming and Software Engineering Fundamentals" by Duke University and "Object Oriented Programming in Java" by University of California, San Diego.

[5] Apache Tomcat: An open-source web server and servlet container for deploying Java web applications.

[6] IntelliJ IDEA or Eclipse: Popular Integrated Development Environments (IDEs) for Java development with support for web development plugins.

[7] Community-driven platform for asking and answering programming-related questions, including Java and web development topics on StackOverflow.