FULL STACK WEB DESIGN AND DEVELOPMENT

FREQUENTLY ASKED QUESTIONS –FAQS

Table of Contents

What are grid lines in CSS?

In CSS, grid lines refer to the lines that form the grid tracks in a CSS grid. A grid is a layout system in CSS that allows you to create complex, two-dimensional layouts with rows and columns. The grid lines divide the grid into a series of tracks, which can be either rows or columns.

Grid lines are numbered starting from 1 and increase towards the right for columns and towards the bottom for rows. You can reference specific grid lines using their line number or by using a grid line name that you define using the grid-template-rows and grid-template-columns properties.

For example, if you have a grid with two rows and three columns, the horizontal grid lines would be numbered 1, 2, and 3, while the vertical grid lines would be numbered 1 and 2. You could reference the first row by using grid-row: 1, and the second column by using grid-column: 2.

You can also use grid lines to specify the placement of grid items using the grid-row-start, grid-row-end, grid-column-start, and grid-column-end properties, which allow you to specify the start and end positions of an item in terms of grid lines.

What is a Framework?

In computer programming, a framework is a set of pre-written code that provides a structure for building software applications. A framework typically includes a set of libraries, tools, and best practices that help developers to build, test, and deploy their applications quickly and efficiently.

Frameworks are designed to solve common problems that developers face when building applications, such as handling database connections, processing user input, and managing application logic. By providing pre-written code for these tasks, frameworks can save developers a significant amount of time and effort, allowing them to focus on building the unique features of their application.

Frameworks can be written for a specific programming language, such as Ruby on Rails for Ruby, or they can be language-agnostic, like Angular or React, which are web development frameworks for building dynamic, single-page applications using JavaScript. Some popular frameworks used in web development include Django and Flask for Python, Laravel for PHP, and .NET for C#.

Using a framework can also have other benefits, such as standardizing the application architecture across multiple developers and making it easier to maintain and update the application over time. However, using a framework can also have downsides, such as a learning curve to master the framework and potential performance overhead due to the added layers of abstraction.

Who was the inventor of Bootstrap?

Bootstrap was invented by Mark Otto and Jacob Thornton, who were both working at Twitter at the time. The framework was originally designed as an internal tool to help developers at Twitter to create consistent, responsive web interfaces across a wide range of devices and screen sizes.

The first version of Bootstrap, which was called Twitter Blueprint, was released in August 2011. It was later renamed Bootstrap and released as an open-source project in August 2012 under the MIT License, which allows developers to use, modify, and distribute the code freely.

Since its release, Bootstrap has become one of the most popular front-end development frameworks in the world, used by developers to create websites, web applications, and mobile applications. The framework provides a wide range of pre-built components, such as buttons, forms, and navigation menus, as well as a responsive grid system that makes it easy to create layouts that adapt to different screen sizes. It also includes a range of utilities and plugins that make it easy to add advanced functionality to your web projects.

What is a Bootstrap?

In computing, a bootstrap refers to the process of loading and initializing a computer system or software application. The term is derived from the phrase "pulling oneself up by one's bootstraps", which refers to a self-sustaining process that requires no external help.

In the context of web development, Bootstrap is also the name of a popular front-end development framework. Bootstrap is a set of pre-written HTML, CSS, and JavaScript code that makes it easy to create responsive, mobile-first web interfaces. The framework includes a range of UI components, such as navigation menus, forms, buttons, and alerts, as well as a responsive grid system that makes it easy to create layouts that adapt to different screen sizes.

Bootstrap was originally created by Mark Otto and Jacob Thornton at Twitter to help developers create consistent, responsive web interfaces for the wide range of devices and screen sizes used by Twitter's users. Since then, Bootstrap has become one of the most widely used front-end development frameworks in the world, used by developers to create websites, web applications, and mobile applications.

What do you know panels?

In the context of user interfaces, a panel is a container that holds a group of related user interface components or controls. Panels are often used to organize and group related content or functionality within an application or webpage.

Panels are commonly used in desktop applications, where they can be displayed as separate windows or embedded within a larger application window. In web development, panels are often implemented using HTML and CSS, and can be displayed as a separate section of a web page, or as an overlay on top of other content.

Panels can contain a wide range of user interface components, such as buttons, text inputs, checkboxes, and radio buttons. They can also contain more complex components, such as charts, tables, and multimedia content. Panels are often designed to be collapsible or expandable, allowing users to show or hide content as needed.

Panels can be used in a wide range of applications and contexts, such as configuring settings, displaying help or documentation, or presenting data or statistics. They are an important tool for creating clean, organized, and user-friendly interfaces that help users to understand and interact with an application more easily.

Describe the purpose of models?

In computer programming, a model is a representation of a real-world object, system, or process that is used to support software development. The purpose of a model is to provide a simplified, abstract view of a complex system or process, allowing developers to work with it in a more manageable way.

Models are used in a wide range of software development contexts, such as:

- **Database design:** In database design, models are used to define the structure and relationships between data elements. This allows developers to create a blueprint for a database that can be used to generate code or scripts to create the actual database.
- **Software design:** In software design, models are used to describe the architecture and behavior of a software system. This can include models of the user interface, data flow, algorithms, and business logic.
- **Machine learning:** In machine learning, models are used to train and test algorithms that can make predictions or decisions based on data. Machine learning models can be used in a wide range of applications, such as natural language processing, image recognition, and fraud detection.

Overall, the purpose of models is to provide a simplified, abstract view of a complex system or process, allowing developers to reason about it in a more manageable way. By creating models, developers can better understand the requirements of a system or process, design software that meets those requirements, and test and refine their designs.

Who is a developer?

In the context of software development, a developer is a person who writes code and creates software programs and applications. Developers use programming languages, software development tools, and other technologies to design, build, test, and deploy software systems.

Developers can work on a wide range of projects, such as creating web applications, developing mobile apps, building desktop software, designing video games, or working on large-scale enterprise systems. They may work independently or as part of a larger team, collaborating with other developers, designers, project managers, and stakeholders.

Developers may specialize in different areas of software development, such as front-end development (creating user interfaces), back-end development (building server-side systems), full-stack development (working on both front-end and back-end), or in a particular programming language or technology.

To be a successful developer, one needs to have strong problem-solving skills, attention to detail, and the ability to work independently or as part of a team. Developers also need to keep up-to-date with the latest trends and technologies in their field to ensure that they are using the most efficient and effective tools and techniques to build high-quality software systems.

What do you know about cards?

In web development, a card is a UI element that represents a container for content, typically used to display information in a visually appealing and organized way. Cards can contain a variety of content, such as text, images, buttons, and other UI elements, and can be used to present information in a concise and easy-to-digest format.

Cards are often used in web and mobile design to display content in a way that is flexible and responsive, meaning that it can be easily adapted to fit different screen sizes and resolutions. This makes them particularly useful for displaying content in a grid layout, where multiple cards can be arranged in rows and columns.

Cards are also a popular design pattern in e-commerce and social media websites, where they can be used to display products, articles, photos, and other types of content. They are designed to be visually appealing, with a focus on using images, icons, and typography to create a pleasing and cohesive aesthetic.

Cards can be customized in a variety of ways, including changing the background color, adding borders, adjusting the size and position of images, and modifying the layout of the content. They are a versatile and useful UI element that can be used to display a wide range of content in an attractive and organized way.

What is Tailwind CSS?

Tailwind CSS is a utility-first CSS framework that provides a set of pre-defined classes for building custom user interfaces. It is designed to be highly customizable, with a focus on providing a flexible and efficient workflow for web developers.

The framework includes a large number of CSS utility classes, which can be used to add custom styles to HTML elements. These classes are designed to be compostable, meaning that they can be combined to create more complex styles and layouts.

Tailwind CSS is designed to be highly customizable, with a configuration file that allows developers to specify custom colors, fonts, spacing, and other design properties. This makes it easy to create a consistent and cohesive design across an entire web application or website.

One of the key benefits of Tailwind CSS is its ability to help developers work more efficiently by providing a set of pre-defined classes that can be used to quickly create custom styles. This can help to reduce the amount of custom CSS that needs to be written, as well as speed up the development process.

Tailwind CSS is a popular choice among web developers who want to create custom user interfaces quickly and efficiently, while still maintaining a high level of design control and flexibility.

What is meant by default configuration?

In software development, default configuration refers to the settings and parameters that are used by a software system when it is first installed or initialized. These settings are often pre-defined by the software developer and are intended to provide a reasonable starting point for most users.

Default configuration typically includes settings such as language, time zone, display resolution, and other system settings. For software applications, default configuration may also include settings such as the default font, color scheme, and layout.

Default configuration is often customizable, allowing users to modify settings to better fit their needs. In some cases, users may need to modify default configuration in order to properly configure the software to work with their system or environment.

Default configuration is an important aspect of software development because it can greatly impact the user experience. By providing reasonable default settings, software developers can make it easier for users to get started with the software and reduce the amount of configuration that needs to be done before the software can be used effectively. At the same time, allowing users to customize default configuration ensures that the software can be tailored to meet individual needs and preferences.

What is the other name of CSS?

CSS stands for Cascading Style Sheets, and it is sometimes also referred to as simply "Stylesheets". However, there is not another common name for CSS besides these two terms.

What is meant by CSS frameworks?

In web development, a CSS framework is a pre-written set of CSS styles, rules, and components that can be used to create a website or web application more efficiently. CSS frameworks are designed to provide a set of common design patterns and components, such as grids, forms, buttons, and navigation menus, which can be used to build web pages quickly and easily.

CSS frameworks are typically created with the goal of making web development faster and easier, while also providing a consistent and cohesive design across an entire website or application. Frameworks can be customized to match the design of a specific website or application, and they are often designed to be responsive, meaning that they can adapt to different screen sizes and device types.

Some popular CSS frameworks include Bootstrap, Foundation, Bulma, and Materialize. These frameworks provide a range of features and components, and they are widely used by web developers to create responsive, visually appealing websites and applications more efficiently.

What is the use of stylesheet?

In web development, a stylesheet is used to define the visual presentation and layout of a web page or website. Specifically, a stylesheet is a file containing a set of rules and instructions written in CSS (Cascading Style Sheets) that define the appearance of HTML elements on a web page.

By using a stylesheet, web developers can separate the visual presentation of a web page from the underlying HTML structure and content. This allows them to modify the visual design of a website or web page without having to make changes to the underlying HTML markup.

Stylesheets are used to define a wide range of visual properties, including fonts, colors, sizes, borders, backgrounds, positioning, and layout. They can be applied to individual elements on a web page, such as text, images, and buttons, or to groups of elements, such as lists, tables, or sections.

Overall, the use of stylesheets allows web developers to create visually appealing and professional-looking web pages and websites, while also providing flexibility and maintainability by separating the content from the presentation layer.

What is meant by breadcrumbs?

In web design, breadcrumbs refer to a navigation aid that allows users to keep track of their location within a website. Breadcrumbs typically appear at the top of a page and show the user's current location in the website's hierarchy, usually in the form of a series of clickable links.

Breadcrumbs are typically organized hierarchically, with the top-level category appearing first, followed by any subcategories and sub-pages. For example, a breadcrumb trail for a product page on an e-commerce site might look like this:

Home > Clothing > Men's Clothing > Shirts > Casual Shirts > Product Name

Breadcrumbs provide users with a quick and easy way to navigate back to higher-level pages, without having to use the back button or retrace their steps through the website. They are particularly useful for websites with complex navigation structures or a large number of pages.

In addition to helping users navigate a website, breadcrumbs can also have SEO benefits, as they can provide search engines with additional information about the website's structure and hierarchy.

Describe the term cascading?

In the context of CSS (Cascading Style Sheets), the term "cascading" refers to the process of combining multiple styles and rules that apply to the same element on a web page.

When a web page is loaded, the browser applies the styles defined in the CSS file to the various HTML elements on the page. If multiple styles are defined for the same element, the browser uses a set of rules to determine which style takes precedence.

The "cascading" part of CSS refers to the way in which the browser determines which style to apply. The browser considers a number of factors when deciding which style takes precedence, including the order in which the styles are defined, the specificity of the selector used to define the style, and the importance of the style (as defined by the! important keyword).

The cascading nature of CSS allows web developers to define a set of general styles that apply to all elements on a web page, while also providing the ability to override or modify those styles for specific elements or sections of the page. This flexibility allows for greater control over the appearance and layout of a web page, while also making it easier to maintain a consistent design across the site.

What is mean by a device responsive design?

A device responsive design is a type of web design that allows a website to adapt to different screen sizes and device types, such as desktop computers, laptops, tablets, and smartphones. With a device responsive design, a website's layout, content, and navigation adjust dynamically based on the size and orientation of the device's screen.

The goal of a device responsive design is to ensure that a website provides a good user experience on any device, regardless of the screen size or device type. This can involve adjusting the size and layout of text and images, repositioning elements on the page, and altering the navigation menu to make it easier to use on a smaller screen.

A device responsive design typically uses a combination of HTML, CSS, and JavaScript to detect the screen size and adjust the layout and content accordingly. Responsive design techniques can include using fluid grid layouts, flexible images, and media queries, which allow the web page to respond to the characteristics of the device being used.

Overall, a device responsive design is important for ensuring that a website is accessible and usable on any device, and it can help to improve user engagement, reduce bounce rates, and increase conversions.

What is meant by backend programming?

Backend programming refers to the development of the server-side of a web application or software. It involves writing code that interacts with the database, manages user authentication, handles HTTP requests, and processes data to produce the output that the user sees on the frontend. In other words, the backend is the part of the application that is responsible for the "behind-the-scenes" functionality, such as retrieving and storing data, managing user accounts, and processing business logic. The frontend, on the other hand, is the part of the application that users interact with directly, such as the user interface and graphics.

Some examples of programming languages and frameworks commonly used for backend development include PHP, Python, Ruby on Rails, Node.js, and Java. These tools allow developers to build web applications that are efficient, secure, and scalable, while also providing a great user experience on the frontend.

What do you know about Laravel?

Laravel is a free, open-source PHP web application framework designed to make it easier for developers to build scalable and secure web applications. It was created by Taylor Otwell in 2011 and has become one of the most popular PHP frameworks in recent years.

Some of the key features of Laravel include:

- **Routing:** Laravel makes it easy to define and handle routes for your application, allowing you to map URLs to specific actions or controller methods.
- **Middleware:** Middleware in Laravel provides a way to filter incoming HTTP requests and take specific actions before or after they are handled by your application.
- **Eloquent ORM:** Laravel's Eloquent ORM makes it easy to work with databases by providing a simple and intuitive interface for defining database tables and relationships.
- **Blade Templating Engine:** Laravel's Blade templating engine provides an easy and efficient way to separate your application's presentation logic from its business logic.

- **Artisan Command-Line Interface:** Laravel's Artisan CLI provides a way to automate common development tasks, such as generating code, running database migrations, and testing your application.
- **Security:** Laravel provides built-in features for protecting your application from common web application vulnerabilities, such as cross-site scripting (XSS) and SQL injection attacks.

Overall, Laravel is a powerful and flexible framework that provides developers with a wide range of tools and features to build robust, scalable, and maintainable web applications.

What are various advantages of Bootstrap?

Bootstrap is a popular front-end development framework that offers a variety of advantages for developers. Here are some of the main benefits of using Bootstrap:

- **Responsiveness:** Bootstrap offers a responsive grid system and a collection of pre-built components that allow developers to easily create responsive web pages that look great on any device, from desktops to mobile phones.
- **Consistency:** Bootstrap provides a consistent and unified design language across all its components, making it easy to create a cohesive look and feel for your application.
- **Customization:** Bootstrap is highly customizable, allowing developers to modify and extend its CSS and JavaScript components to fit the needs of their project.
- **Rapid development:** Bootstrap offers a large set of pre-built components and plugins that can be easily incorporated into your application, saving developers time and effort in building custom components from scratch.
- **Cross-browser compatibility:** Bootstrap is tested and supported across all major browsers, ensuring that your application works consistently across different browsers and platforms.
- **Accessibility:** Bootstrap is designed with accessibility in mind, ensuring that your application is usable by people with disabilities.
- **Community support:** Bootstrap is widely used and has a large and active community of developers who contribute to its development, offer support and advice, and provide plugins and extensions.

Overall, Bootstrap is a powerful and flexible framework that offers a variety of benefits for developers looking to build responsive, consistent, and customizable web applications quickly and efficiently.

What is meant by break points?

In the context of web development, "breakpoints" refer to specific points or ranges of screen sizes at which a website's layout or design changes. When a website is viewed on a smaller device, such as a smartphone or tablet, the content may need to be rearranged or resized to fit the smaller screen size.

Breakpoints allow developers to define specific screen sizes at which these layout changes occur, ensuring that the website looks good and remains usable on any device. For example, a website may be designed with a desktop layout that includes multiple columns and a full-width banner, but when the screen size is reduced below a certain breakpoint, the layout may switch to a single column with a smaller banner.

Developers can use CSS media queries to define breakpoints and adjust the styles of various elements on the page at different screen sizes. By setting breakpoints and adjusting styles accordingly, developers can ensure that the website remains responsive and easy to use across a wide range of devices and screen sizes

What do you know about prototyping?

Prototyping is the process of creating an early version or model of a product, service, or system that is used to test and validate its design, functionality, and usability. In the context of software development, prototyping typically involves creating a working model of a website, application, or other digital product to explore its features and test its user experience.

Prototyping is an important step in the design process because it allows designers and developers to identify and resolve issues early in the development cycle, when changes are easier and less expensive to make. By creating a prototype, designers can test the product's usability, get feedback from users, and refine the design before moving on to the development phase.

There are several different types of prototypes that can be created, including low-fidelity paper prototypes, wireframes, interactive prototypes, and high-fidelity prototypes. Each type of prototype has its own advantages and disadvantages, depending on the goals of the design process and the resources available.

Overall, prototyping is an essential part of the design process, as it allows designers and developers to iterate on their ideas, identify and resolve issues early in the development cycle, and create products that are more usable, functional, and user-friendly.

What is a view or User Interface(UI)?

A view, or User Interface (UI), in the context of software development, refers to the part of a program that is visible to and interacts with the user. The view is the layer of the application that is responsible for rendering the graphical user interface (GUI) and providing a way for users to interact with the system.

In web development, views are typically created using HTML, CSS, and JavaScript, and are displayed in a user's web browser. The view is responsible for presenting data to the user, allowing them to interact with the system, and providing feedback on their actions.

Examples of UI elements that might be included in a view include buttons, text fields, menus, and other graphical elements that allow users to navigate the application and perform actions.

The design of the view is an important part of the user experience, and designers must consider factors such as usability, accessibility, and visual design when creating the UI. Good UI design can make an application more intuitive and user-friendly, while poor UI design can lead to frustration and confusion for users.

Overall, the view or UI is a critical component of any software application, as it provides the means for users to interact with the system and achieve their goals.

What is meant by components?

In software development, a component is a self-contained, reusable module that performs a specific function within a larger software system. Components can be thought of as building blocks that can be combined to create more complex software applications.

Components typically have well-defined interfaces and can communicate with other components through standardized methods, such as application programming interfaces (APIs). This allows components to be developed independently and used across multiple applications or systems.

Components can take many different forms, depending on the needs of the system. For example, a component might be a software library that provides a set of functions or classes that can be used by other parts of the application. Alternatively, a component might be a graphical user interface (GUI) widget, such as a button or menu, that can be reused across multiple screens or applications.

The use of components in software development has a number of benefits, including:

- **Reusability:** Components can be reused across multiple applications or systems, reducing development time and cost.
- **Modularity:** Components can be developed independently and tested in isolation, making it easier to maintain and update the system over time.
- **Scalability:** Components can be added or removed from the system as needed, making it easier to scale the system to meet changing demands.
- **Interoperability:** Components with well-defined interfaces can communicate with other components from different vendors or systems, making it easier to integrate different technologies and services.

Overall, the use of components is a common practice in software development, as it allows developers to build more complex systems more efficiently and with greater flexibility.

Write the role of JavaScript?

JavaScript is a programming language that is primarily used to add dynamic, interactive behavior to websites and web applications. Some of the key roles of JavaScript in web development include:

Client-side validation: JavaScript can be used to validate user inputs on the client-side, before the data is submitted to the server. This can help to improve the user experience by providing instant feedback and reducing the need for server round-trips.

Dynamic content and interaction: JavaScript can be used to create interactive user interfaces that respond to user input and change dynamically based on the state of the application. This can include features such as dropdown menus, tooltips, and modal windows.

Browser manipulation: JavaScript can be used to manipulate the DOM (Document Object Model) of a web page, allowing developers to dynamically change the content and structure of the page without needing to reload the entire page.

Asynchronous communication: JavaScript can be used to make asynchronous requests to a server, allowing web applications to update content without requiring a page reload. This can be used to build real-time web applications such as chat apps and social media platforms.

Animation and visual effects: JavaScript can be used to create animations and visual effects on a web page, allowing developers to add a professional touch to their websites.

Overall, JavaScript plays a critical role in modern web development, enabling developers to create rich, interactive user experiences that run entirely within the user's web browser.

What do you know about term Cascading?

In web development, cascading refers to the way that styles are applied to HTML elements using Cascading Style Sheets (CSS). The term "cascading" comes from the fact that multiple styles can be applied to the same element, and these styles can "cascade" down to the element from multiple sources, such as user-defined styles, styles defined in external CSS files, and default browser styles.

The cascading behavior of CSS is determined by a set of rules that define how conflicting styles are resolved. These rules are based on the specificity of the styles, with more specific styles taking precedence over less specific styles. The rules also take into account the order in which the styles are defined, with styles defined later in the CSS file taking precedence over styles defined earlier.

The cascading behavior of CSS is what allows developers to create consistent, reusable styles across multiple web pages and applications. By defining styles once and applying them to multiple elements, developers can save time and ensure a consistent look and feel across their websites. However, the cascading behavior can also lead to unintended consequences if styles conflict or if the rules for resolving conflicts are not well understood.

Overall, the cascading behavior of CSS is a fundamental aspect of web development, and understanding how it works is essential for creating effective, well-designed websites.

What are different ways to apply cascading stylesheet?

There are several ways to apply cascading style sheets (CSS) to HTML documents:

Inline Style: You can apply styles directly to an HTML element by using the "style" attribute. For example, <p style="color: red;">This is a red paragraph. </p> will create a paragraph with red text.

Internal Style Sheet: You can define styles within the head section of an HTML document using the <style> tag. For example:

<head> <style> p { color: red; } </style> </head>

This will set the color of all paragraph elements in the document to red.

External Style Sheet: You can define styles in a separate CSS file and link to it from your HTML document using the <link> tag. For example:

<head> <link rel="stylesheet" href="styles.css"> </head>

This will link to an external CSS file named "styles.css" and apply the styles defined in that file to the HTML document.

@import Rule: You can use the @import rule to import an external CSS file into another CSS file. For example:

@import url("styles.css");

This will import an external CSS file named "styles.css" into the current CSS file.

CSS in JavaScript: You can use JavaScript to apply CSS styles to HTML elements dynamically. For example:

document.getElementById("myElement").style.color = "red";

This will set the color of an element with the ID "myElement" to red.

What is general idea behind CSS?

The general idea behind CSS (Cascading Style Sheets) is to separate the presentation or visual design of a webpage from its content and structure. CSS is used to style HTML elements, and it allows web developers to control the layout, typography, color, and other visual aspects of a webpage.

By separating the presentation from the content, CSS enables web developers to make changes to the visual design of a webpage without having to modify the underlying HTML code. This makes it easier to maintain and update a website, as changes to the visual design can be made through the CSS file, rather than having to make changes to each individual HTML element.

CSS works by applying rules or styles to specific HTML elements, either by targeting individual elements or by applying styles to groups of elements. These styles can be defined in a separate CSS file or embedded within the HTML code itself.

Overall, the general idea behind CSS is to provide a flexible and efficient way to style and design webpages, while keeping the content and structure separate from the visual presentation.

Describe actual role of stylesheets?

The actual role of stylesheets in web development is to define the visual presentation or styling of a webpage, separate from its content and structure. Stylesheets are written in CSS (Cascading Style Sheets) and are used to apply a set of rules to specific HTML elements to define their appearance, such as color, font, size, position, and layout.

Stylesheets allow web developers to:

Keep the content and structure of a webpage separate from its visual design, making it easier to maintain and update the site.

Apply consistent styling across all pages of a website, ensuring a cohesive and professional look.

Create responsive designs that adapt to different screen sizes and devices, by using media queries and other techniques to adjust the layout and styling.

Improve accessibility by making it possible to adjust the font size and other visual aspects of a page for users with different needs.

Improve performance by reducing the amount of code that needs to be downloaded and processed by the browser, since stylesheets can be cached and reused across multiple pages.

Overall, the role of stylesheets is to define the visual design of a website, and to provide a flexible, efficient, and scalable way to create beautiful, responsive, and accessible webpages.

Instructor:

ABUBAKAR SIDDIQUE

Full-Stack Developer

MS in Computer Science (MDS, WSN, MCS, BED), LCC (AMAP, Online), BTI (CIT, Australia), LDT (open SAP, Online), CLC (Peking University, China), CAC (TEVTA, PAK), ICSC (The Open University, UK), Data Science (SAP, Online), RPJ (VU, PAK)

dr.sagher@gmail.com, +923017362696