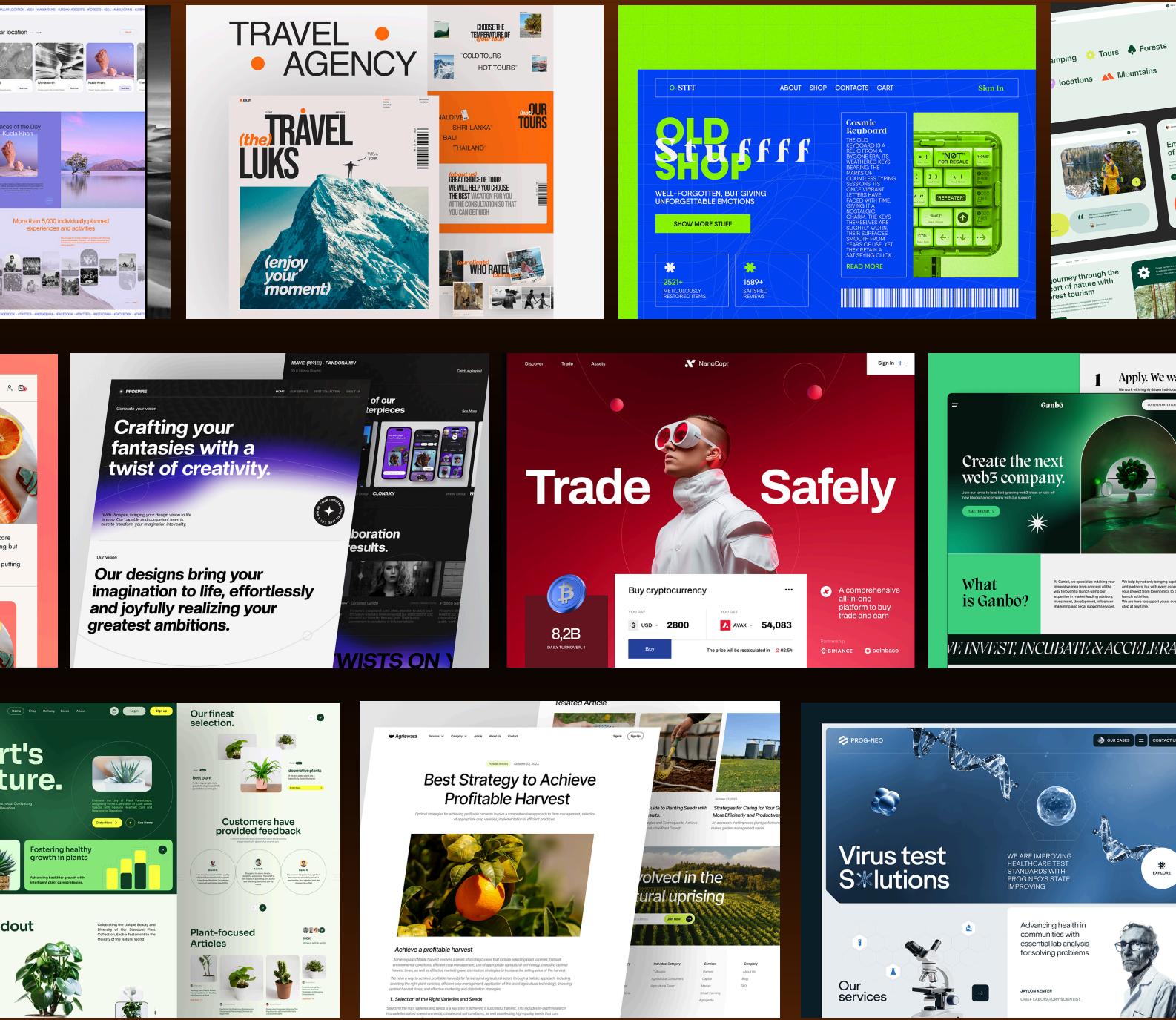


Introduction to

WEB DESIGN



INTRODUCTION TO INTERNET

The internet is a global network that connects millions of computers and other devices worldwide. It allows these devices to communicate with each other through various means such as wired and wireless connections.

The internet functions on a protocol called TCP/IP (Transmission Control Protocol/Internet Protocol), which enables data to be transmitted reliably between devices connected to the network.

Originally developed in the late 1960s by the United States Department of Defense, the internet was initially known as ARPANET. It was designed to facilitate communication and data exchange between research institutions and government agencies.

Over the decades, the internet has evolved into a vast network that touches almost every aspect of modern life. It serves as a platform for communication, information exchange, entertainment, commerce, and much more.

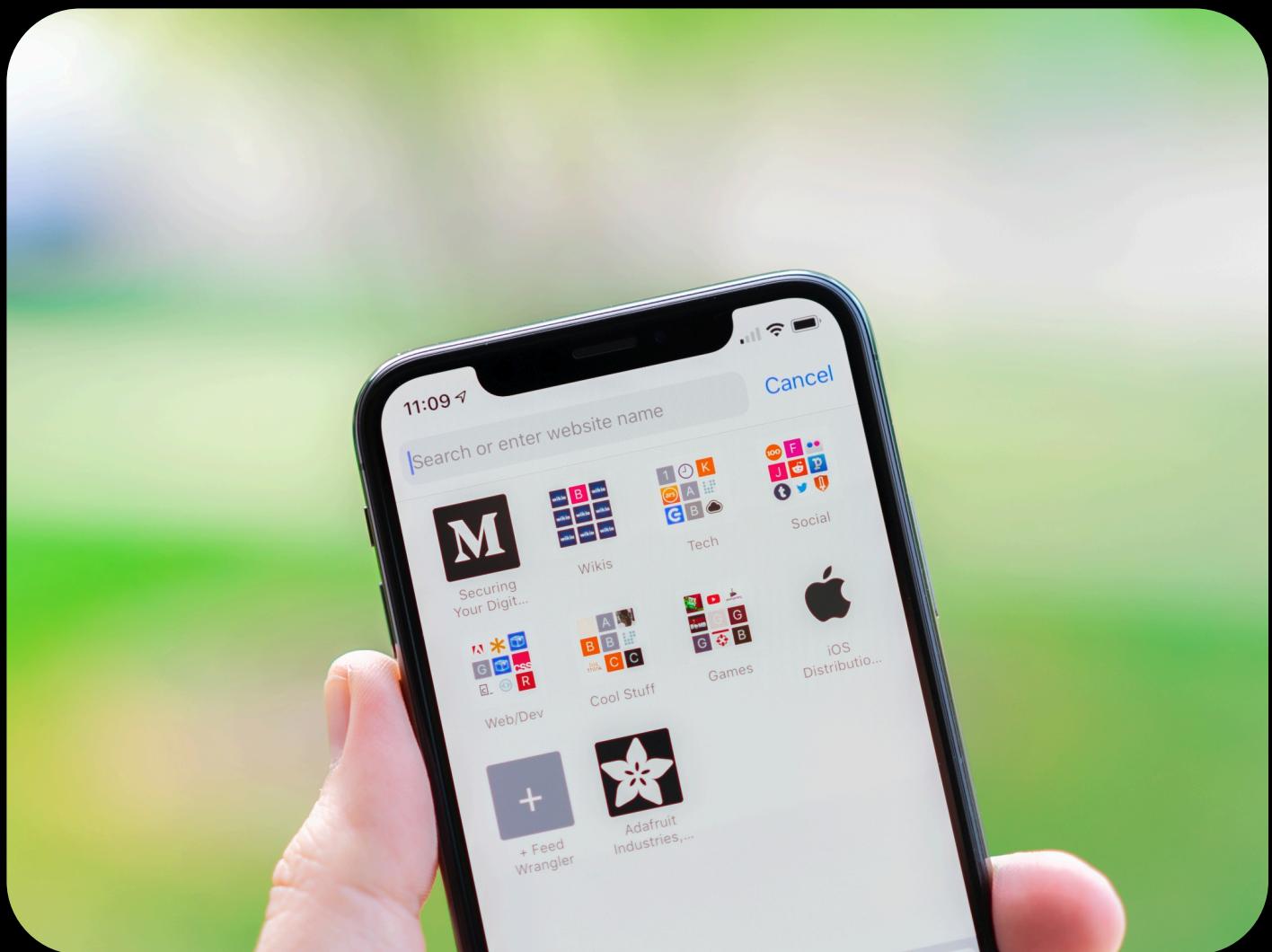
Some key components of the internet include:

- World Wide Web (WWW): Often used interchangeably with the internet, the World Wide Web is actually a system of interlinked hypertext documents accessed via the internet. It was invented by Tim Berners-Lee in 1989.
- Web Browsers: Software applications that allow users to access and navigate the World Wide Web. Examples include Google Chrome, Mozilla Firefox, and Safari.
- Websites: Collections of web pages that contain various types of content, such as text, images, videos, and interactive elements. Websites are hosted on web servers and accessed using web browsers.
- Search Engines: Tools that help users find specific information on the internet by searching through vast amounts of web pages. Examples include Google, Bing, and Yahoo.
- Email: Electronic mail allows users to send and receive messages over the internet. It is one of the earliest and most widely used internet services.

INTRODUCTION TO INTERNET

- **Social Media:** Platforms that enable users to create and share content, connect with others, and participate in online communities. Examples include Facebook, Twitter, and Instagram.
- **Online Services:** Various online services such as online banking, e-commerce, streaming media, and cloud computing have become integral parts of the internet.

Overall, the internet has revolutionized the way we communicate, work, learn, and entertain ourselves, making it one of the most significant technological innovations of the modern era.



W O R L D W I D E W E B

WWW

The World Wide Web (WWW), often referred to simply as the web, is an information system that allows documents and other resources to be accessed over the Internet. It was invented by Sir Tim Berners-Lee, a British computer scientist, in 1989, and its development was facilitated by the creation of the Hypertext Transfer Protocol (HTTP) and the Hypertext Markup Language (HTML).

The web operates through a network of interconnected documents and resources, which are linked together using hyperlinks. Users can navigate through these resources by clicking on hyperlinks or by using web browsers to access specific URLs (Uniform Resource Locators).

The web has become an essential part of modern life, enabling access to vast amounts of information, services, and entertainment. It encompasses everything from static web pages to dynamic web applications, allowing for communication, collaboration, commerce, and much more.



W H A T I S

WEBSITE?

A website is a digital space hosted on the internet that houses various web pages containing information, media, or interactive elements. It serves as a platform for individuals, businesses, organizations, and institutions to showcase their content, products, or services to a global audience. Websites are accessed through web browsers like Chrome, Firefox, or Safari, enabling users to navigate through different pages using hyperlinks, menus, or search functionalities.

Each website is uniquely designed and structured, often featuring a homepage as the main entry point that provides an overview of the site's content and purpose. Websites can range from simple static pages with basic text and images to complex dynamic platforms with interactive features such as forms, e-commerce capabilities, or multimedia elements like videos and animations. They can serve various purposes, including informational, educational, entertainment, social networking, or commercial activities.

Behind the scenes, websites are built using various technologies such as HTML, CSS, and JavaScript for front-end development, while server-side scripting languages like PHP, Python, or Ruby are used for back-end functionality. Additionally, databases are often employed to store and retrieve dynamic content. Websites are continually evolving entities, subject to updates, redesigns, and optimizations to enhance user experience, accessibility, and functionality in line with changing technological advancements and user expectations.

FRONT-END

The "front end" of a website or application refers to everything that users interact with directly. It encompasses the design, layout, and user interface elements that users see and interact with in their browsers or on their devices. Front-end development involves the use of languages like HTML, CSS, and JavaScript to create the visual and interactive aspects of a website or application.

Front-end developers focus on creating a seamless user experience by ensuring that the interface is intuitive, responsive, and visually appealing. They work closely with designers to implement the visual elements of a project and with back-end developers to integrate the front-end with the underlying server infrastructure.

In addition to creating the user interface, front-end developers are also responsible for optimizing the performance of the website or application. This includes tasks like minimizing page load times, optimizing images and other media, and ensuring compatibility across different browsers and devices. Ultimately, the front end plays a crucial role in shaping the overall user experience and determining the success of a digital product.

BACK-END

The "back end" of a website or application refers to the server-side components and logic that operate behind the scenes to power the front-end interface. It includes the server, database, and application logic responsible for processing requests, managing data, and delivering content to users. Unlike the front end, which users interact with directly, the back end is typically invisible to users but essential for the functionality of the application.

Back-end development involves working with server-side programming languages like Python, Ruby, Java, or Node.js to build the logic and functionality of the application. This includes tasks such as handling user authentication, processing form submissions, accessing and manipulating data in databases, and integrating with external services and APIs.

Back-end developers focus on creating robust, scalable, and secure systems that can handle large volumes of traffic and data. They often work closely with front-end developers to ensure that the front-end interface communicates effectively with the back end and that data is transferred securely between the two.

In addition to building and maintaining the core functionality of the application, back-end developers are also responsible for ensuring the security and performance of the system. This includes implementing measures to protect against security threats such as SQL injection and cross-site scripting, as well as optimizing the performance of the server and database to minimize response times and maximize scalability. Overall, the back end is essential for powering the functionality and performance of modern web applications.

CLIENT-SIDE

Client-side refers to the part of software or a web application that runs on the user's device, typically a web browser, rather than on the server. It's responsible for the user interface and client-side interactions. Client-side languages are programming languages that are primarily used to develop functionalities that run on the client side.

The main client-side languages used in web development are:

- **JavaScript:** JavaScript is the most common client-side language used for web development. It allows developers to add interactivity, animations, and dynamic content to web pages. JavaScript frameworks/libraries like React, Angular, and Vue.js are widely used to build complex web applications.
- **HTML (Hypertext Markup Language):** Although HTML is not a programming language per se, it's the standard markup language used to create the structure of web pages. It defines the layout and content of the web page elements.
- **CSS (Cascading Style Sheets):** CSS is used to style HTML elements, controlling the layout, appearance, and presentation of web pages. It defines how HTML elements should be displayed on the screen, including colors, fonts, spacing, and more.

These three languages, JavaScript, HTML, and CSS, form the backbone of client-side web development. They work together to create interactive and visually appealing web applications that run smoothly in web browsers. Additionally, there are other languages and technologies like TypeScript (a superset of JavaScript), Sass (a CSS preprocessor), and various frameworks and libraries that complement these core technologies in client-side development.

SERVER-SIDE

Server-side refers to the part of software or a web application that runs on the server rather than on the client's device. It handles the logic, data processing, and database operations of an application, providing responses to client requests. Server-side languages are used to develop these functionalities that execute on the server.

The main server-side languages used in web development are:

- **PHP:** PHP (Hypertext Preprocessor) is a server-side scripting language widely used for web development. It is commonly used to generate dynamic web pages, interact with databases, handle form data, manage sessions, and perform various server-side tasks.
- **Python:** Python is a versatile programming language used for server-side development, among other purposes. It offers web frameworks like Django and Flask, which provide tools and libraries for building web applications with ease.
- **Ruby:** Ruby is another programming language commonly used for server-side development. The Ruby on Rails framework is particularly popular for developing web applications quickly and efficiently.
- **Java:** Java is a robust and widely used programming language for server-side development. Java-based frameworks like Spring and Jakarta EE provide extensive libraries and tools for building enterprise-grade web applications.
- **Node.js (JavaScript):** While JavaScript is primarily a client-side language, Node.js allows developers to run JavaScript code on the server. It is built on the V8 JavaScript engine and is particularly well-suited for building scalable and high-performance web applications.

These server-side languages and frameworks handle tasks such as processing user requests, accessing databases, authentication, authorization, business logic implementation, and generating dynamic content that is then sent to the client's browser for display. Server-side languages work in conjunction with client-side technologies like HTML, CSS, and JavaScript to create full-fledged web applications.

RESPONSIVE

W E B D E S I G N I N G

Responsive web design is an approach to web design aimed at crafting sites to provide an optimal viewing and interaction experience across a wide range of devices, from desktop computers to mobile phones. It involves using flexible grids and layouts, images, and CSS media queries to adapt the design to the viewing environment. By employing responsive techniques, websites can automatically adjust their layout and content based on the screen size and orientation of the device being used, ensuring a seamless and user-friendly experience for visitors regardless of the device they're using.

One of the key principles of responsive web design is fluidity, which means that elements on the webpage are sized in relative units like percentages rather than fixed units like pixels. This allows them to resize and reposition themselves based on the size of the viewport. Another important aspect is flexibility, which involves using CSS media queries to apply different styles based on the characteristics of the device, such as its width, height, resolution, and orientation. These queries enable designers to create layouts that are optimized for different screen sizes and ensure that content remains readable and accessible across devices.

Overall, responsive web design is essential in today's digital landscape where users access websites through a variety of devices with different screen sizes and capabilities. By adopting responsive techniques, businesses can enhance their online presence, improve user experience, and reach a wider audience, ultimately driving engagement and conversions. Moreover, responsive design aligns with modern web standards and best practices, ensuring that websites remain relevant and accessible in an ever-evolving technological landscape.

TYPES OF

W E B S I T E S

Static and dynamic websites are two common types of websites, each with distinct characteristics and purposes:

- **Static Websites:**

- Content: Static websites contain fixed content that remains the same for every user who visits the site. This content typically includes text, images, videos, and other multimedia elements.
- Design: The design of a static website is usually created using HTML and CSS. Each web page is a separate file stored on the server.
- Interactivity: Static websites are limited in interactivity. They can include simple forms and navigation menus but lack dynamic features like user logins, personalized content, or real-time updates.
- Advantages: Static websites are easy to create and host since they require minimal server-side processing. They are also faster to load since there's no need to generate content dynamically.

- **Dynamic Websites:**

- Content: Dynamic websites generate content dynamically in response to user actions or requests. Content can be personalized based on user preferences, login credentials, or other variables.
- Design: Dynamic websites often use server-side scripting languages like PHP, Python, or Ruby, along with databases like MySQL or PostgreSQL. These technologies allow the website to retrieve and manipulate data to generate dynamic content.
- Interactivity: Dynamic websites offer a high level of interactivity. They can support user logins, content management systems, e-commerce platforms, forums, social media integration, and other advanced features.
- Advantages: Dynamic websites are more flexible and scalable since they can adapt to user input and preferences. They are suitable for websites that require frequent updates, user engagement, or complex functionality.

In summary, while static websites are simpler and more straightforward, dynamic websites offer greater interactivity and flexibility, making them ideal for applications that require user engagement and real-time content updates.



THANK YOU

CONTACT US
FOR FURTHER
INQUIRIES

+0191-2585266
doeaccaksharcomputercentre.com

AKSHAR
COMPUTER
CENTRE