**Difference Between Static and Dynamic Websites**

Static and dynamic websites differ in how they are built, how they function, and how they interact with users. Below is a detailed comparison, including **client-server architecture** and **real-world examples** like an **organization's static website** and a **banking app as a dynamic website**.

**1. Static Website**

A **static website** consists of fixed content that does not change unless manually updated by a developer. These websites are built using **HTML, CSS, and JavaScript**, and they do not require a database or server-side processing.

**Characteristics of Static Websites:**

* Displays **the same content** to all users.
* Pages are pre-built and stored on the server.
* Faster loading speed since there is no server-side processing.
* Less expensive to host and maintain.
* No real-time updates or user interactivity.

**Client-Server Architecture for Static Website:**

1. The **client (browser)** requests a webpage from the server.
2. The **server** retrieves and sends the HTML, CSS, and JavaScript files to the client.
3. The client displays the webpage exactly as it was stored on the server.

**Example: Organization’s Static Website**

Many organizations have **static websites** for their **company profile, services, and contact information**. For example:

* A **university website** displaying course details.
* A **corporate website** showing company history and leadership.
* A **portfolio website** for showcasing work.

These sites do not require frequent updates or user interaction.

**2. Dynamic Website**

A **dynamic website** generates content in real time based on user interaction and database queries. These websites use **server-side technologies** such as **PHP, Python, Node.js, and databases (MySQL, MongoDB, etc.)**.

**Characteristics of Dynamic Websites:**

* Content changes dynamically based on user input, preferences, and interactions.
* Requires server-side processing and a database.
* Allows users to log in, make transactions, and personalize experiences.
* More resource-intensive than static websites.

**Client-Server Architecture for Dynamic Website:**

1. The **client (browser)** requests a webpage.
2. The **server processes the request**, interacts with the database, and generates a response.
3. The generated page is sent back to the client and displayed dynamically.

**Example: Banking App as a Dynamic Website**

A **banking application** is a **dynamic web-based system** because:

* Users log in and see **personalized account details**.
* Transactions are processed in real time (e.g., balance updates, fund transfers).
* Security mechanisms like authentication and authorization are required.
* The database stores and retrieves customer data dynamically.

**Example Platforms:** Online banking systems like **HDFC NetBanking, SBI Online, PayPal, and Revolut**.

**Key Differences Between Static and Dynamic Websites**

| **Feature** | **Static Website** | **Dynamic Website** |
| --- | --- | --- |
| **Content** | Fixed, does not change unless manually updated | Changes based on user interaction |
| **Technology** | HTML, CSS, JavaScript | Uses server-side languages (PHP, Python, etc.), databases |
| **Database** | Not required | Required for storing data |
| **User Interaction** | Minimal or none | High interactivity (logins, transactions, etc.) |
| **Speed** | Fast | Slower due to server processing |
| **Example** | University website, portfolio | Banking app, social media |

**Conclusion**

* A **static website** is ideal for showcasing fixed information with minimal updates.
* A **dynamic website** is necessary for applications that require real-time data processing and user interaction.