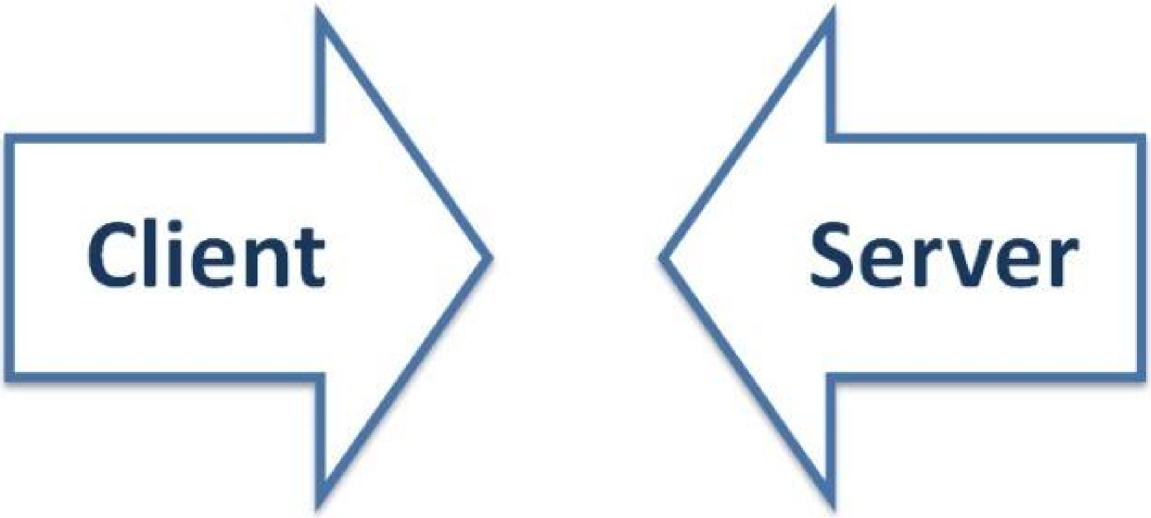
Introduction

In this article, we will learn about Client-side vs Server-side Programming Languages.

Basic Background

Web development is all about communication and data exchange. This communication takes place via two parties over the HTTP protocol.

These parties are:



Server

The Server is responsible for serving the web pages depending on the client/end-user requirement. It can be either static or dynamic.

Client

A client is a party that requests pages from the server and displays them to the end-user. In general a client program is a web browser.

Example | Working

We can explain this entire mechanism using the following:

- The user opens his web browser (client)
- The user starts browsing

(for example http://c-sharpcorner.com)

- The client forwards this request to the server, for accessing their web page.
- The server then acknowledges the request and replies back to the client program.

  (An access link to that web page)

- The client then receives the page source and renders it.

  (Into a viewable/under a stable website)

- Now the user types into the search bar
- The client then submits data to the server
- The server processes the data and replies back with a related search result
- The client again renders it back for the user's view
- The user gets access to the requested link.

Server-side Programming

Server-side programming can be explained as:

It is the general name for the kind of program that runs directly on the server.

Or we can say that server-side programming must deal with dynamic content. It runs on the server. Most web pages are not static since they deal with searching databases.

Server-side Uses

- It processes the user input
- Displays the requested pages
- Structure of web applications
- Interaction with servers/storages
- Interaction with databases
- Querying the database
- Encoding of data into HTML
- Operations over databases like delete, update.

Server-side Languages Example

There are several languages that can be used for server-side programming:

- PHP

- ASP.NET (C# OR Visual Basic)
- C++
- Java and JSP
- Python
- Ruby on Rails and so on.

Server-side Example

```
// This is a sample C# code.
using System;
// namespace
class ServerSide
{
    public static void Main()
    {
        System.Console.WriteLine(“Hello C# Corner”);
        // printing a line
    }
}
```

Client-side Programming

Similarly to server-side programming, client-side programming is also the name of the entire program that runs on the client.

Or we can say that client-side programming mostly deals with the user interface with which the user interacts in the web. It is mostly a browser, in the user's machine, that runs the code and is mainly done in any scripting language like JavaScript (or we can use Flash instead of JavaScript or VNScript).

Client-side Uses

- Makes interactive web pages
- Make stuff work dynamically
- Interact with temporary storage
- Works as an interface between user and server
- Sends requests to the server
- Retrieval of data from Server
- Interact with local storage
- Provides remote access for client-server program

Client-side Languages Example

There are many client-side scripting languages too.

- JavaScript
- VBScript
- HTML (Structure)
- CSS (Designing)
- AJAX
- jQuery etc.

(Some other languages also can be used on the basis of the modeling/designing /graphics/animations and for extra functionalities.)

Client-side Example

```
// sample HTML code
<html>
<head>
    <title>Client Side </title>
</head>
<body>
    <h1>
        Hello C# Corner
    </h1>
</body>
</html>
```

Summary

In this article, we learned about Client-side vs Server-side Programming Languages.