**Differentiation of Client-side and Server-side scripting:**

|  |  |
| --- | --- |
| **Client-side** | **Sever-side** |
| The primary function of client-side scripting is to provide the requested output to the end-user | The primary function of server-side scripting is to manipulate and give access to the required database as per request. |
| The client side is used as the front end, where the user gets to see what we have browsed. | The server side is used as a back end where data is processed and is not visible to the client user. |
| On the client side, the user is allowed to access the code written after verifying the user’s need. | Server-side scripting allows the back-end developer to hide the source code from the user. |
| The client side does not need any interaction with the server. | Server-side scripting on the other hand is all about communicating with the servers. |
| Used for the visibility and getting out the required data from servers’ database | Used for the customization or modification of the data to change the website dynamically. |
| Client-side scripting depends upon the user’s browser version. | Serve-side does not depend on the client. |
| This way of scripting is less secure than Server-side scripting because of the accessibility of code provided to the client. | Server-side scripting is considered a more secure way while working on a web application |
| The client side does not connect to the database at the webserver | The server side helps connect with the database, which is already stored in the server database. |
| It occurs when the browser process all the codes, and then it reacts according to the client’s query | It only acts after the client initiates the browsing request. |
| It runs on the end-users system. | It runs on the web server. |
| HTML, JavaScript, and CSS are used to display the request | PHP, Python, Ruby, nodejs are some of the programming languages used on server-side |