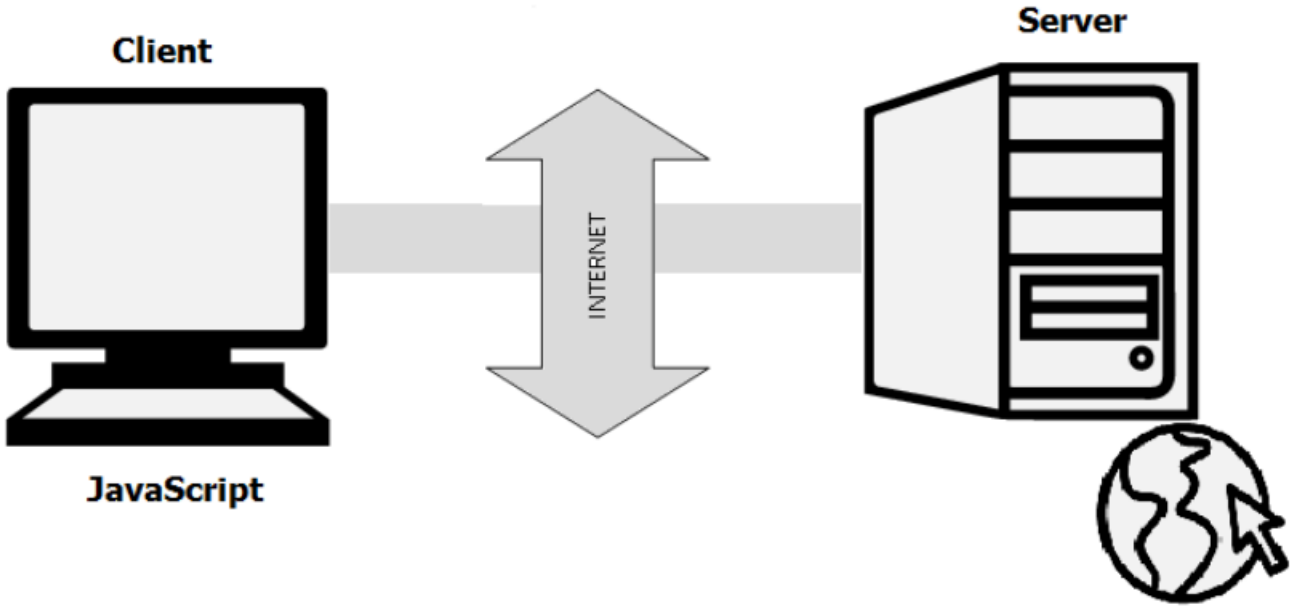
The Server Side Javascript

Prof. Rachana V. Modi

JavaScript

- Powerful client-side scripting language
- Lightweight and Cross-platform
- Open source scripting language supported by all browsers
- Translated language



Application of JavaScript

- JavaScript is used to create interactive websites.
  - Client-side validation
  - Enhancing the interaction of a user with the webpage
  - Dynamic drop-down menus
  - Displaying date and time
  - Displaying pop-up windows and dialog boxes
  - Perform calculations

CLIENT SIDE VS SERVER SIDE SCRIPTING

CLIENT SIDE SCRIPTING	SERVER SIDE SCRIPTING
Runs on client computer's browser	Runs on web server
Functionality: 1) Interact with temporary storage 2) Make interactive web pages 3) Sending request for data to server 4) Interface between server and user	Functionality: 1) Querying the database 2) Operations over databases 3) Access/Write a file on server 4) Interact with other servers 5) Structure web applications 6) Process user input
Example: HTML, CSS, Javascript	Example: PHP, Java, Python, ASP.NET
Not secure	Secure

What is Node.js?

- Node.js is a JavaScript runtime environment
- Useful for server-side development
- Open source and platform independent
- It provides a rich library of modules that simplify the development process.

Node.js = Runtime Environment + JavaScript Library

- It was created by Ryan Dahl in 2009.
- Node.js runs on V8.
- Node.js uses an event-driven and non-blocking I/O model that makes it lightweight and efficient.

How node.js differ with other languages?

- File request handled by PHP or ASP:
  1. Sends the task to the computer's file system.
  2. Waits while the file system opens and reads the file.
  3. Returns the content to the client.
  4. Ready to handle the next request.

- File request handled by Node.js:
  1. Sends the task to the computer's file system.
  2. Ready to handle the next request.
  3. When the file system has opened and read the file, the server returns the content to the client.

- Node.js runs single-threaded, non-blocking, asynchronously programming, which is very memory efficient.

What Can Node.js Do?

- Node.js creates different types of applications
- Node.js can generate dynamic page content
- Node.js can create, open, read, write, delete, and close files on the server
- Node.js can collect form data
- Node.js can add, delete, modify data in your database

Advantages of Node.js

- Extremely fast
- I/O is Asynchronous and Event Driven
- Single threaded
- Highly Scalable
- No buffering
- Open source
- Library

Why Node.js is so popular?

- Full stack developer – Javascript
- Development speed and productivity increases
- Minimum hardware configuration required
- Most suitable for real-time applications with intense I/O

Application of node.js

- Online payment system: Paypal
- Social platform: LinkedIn - Professional networking
- Streaming service: Netflix - Video on Demand
- E-Commerce platform: eBay
- Online publishing platform: Medium
- E-learning platform: Quizelet
- Project management: Trello

Any query??