

Server Side Scripting

Learning to love the Terminal

What is a server

A server is a computer that is tasked with serving files to users based on user actions and requests.

Types of Servers:

Static / Media Servers

Application servers



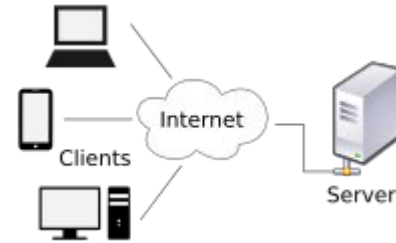
Media Vs Application Servers

Media servers serve static and media files.

Application servers run the code that makes the application work and handles the user requests, which communicates and serves files from both app and media servers.



Client Vs. Server



Client is the browser where the user requests your website or application to make some sort of actions - and displays the response

The Server side code receives requests, processes the request and other user data, and spits back the appropriate response.



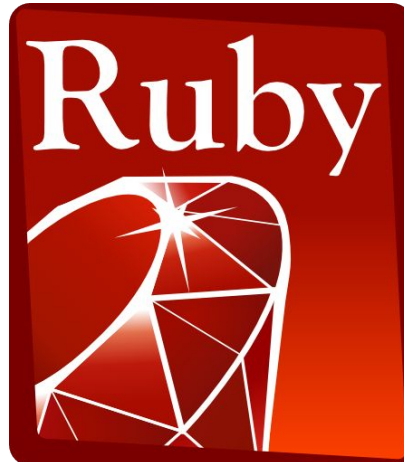
Application Server Tasks

Servers usually run code which does the following:

- Handle user GET requests and serves the proper html and media files.
- Handle user input or POST requests, and process user data handling all communication with the DataBase
- Processing Database data to be displayed on the client side.

Server side Code

Typically programming languages such as Python, Ruby, and Java were used to provide instruction to a server on how to perform the above tasks and alter the data stored in the application



Interacting with your server

Typically servers are setup in some sort of Linux based operating system, and must be navigated through the terminal.



Terminal Commands

`cd DIRECTORY_NAME` - Change Directory (move from one folder to another)

`cd ..` - Go back to an outer folder

`cd` - Go back to a your home folder



`mkdir DIRNAME` - create a new internal folder in whatever folder you are currently in

`rm FileName` - Delete a file in the folder

`cp -SOURCE - Destination` - Copy from a source to another destination on the computer

`mv source destination` - Move file from one place to another (cut and paste)

`sudo` - run as an administrator

Exercise

Cd one folder to another
create a file, move a file



make a directory, cd into the directory, create a text file in nano, open it in nano, and remove it in the end, cd back home



Setting up a Server ENV.

To develop for the server side you must install languages, and libraries for your useage, this is called setting up your environment or env.

Tools: Python PIP, Ruby - Gems, HomeBrew, easy-install, apt-get , npm install

Encapsulating Environments

Certain things can be installed universally on your terminal, but it is generally best practice to set up an encapsulated virtual Environments each specific project.



JS on the Server

For years JS was only for the client side, and thus a full stack dev needed to know JS and another scripting language.

Now with NodeJs and NPM, powerful JS server side scripting is a reality.

