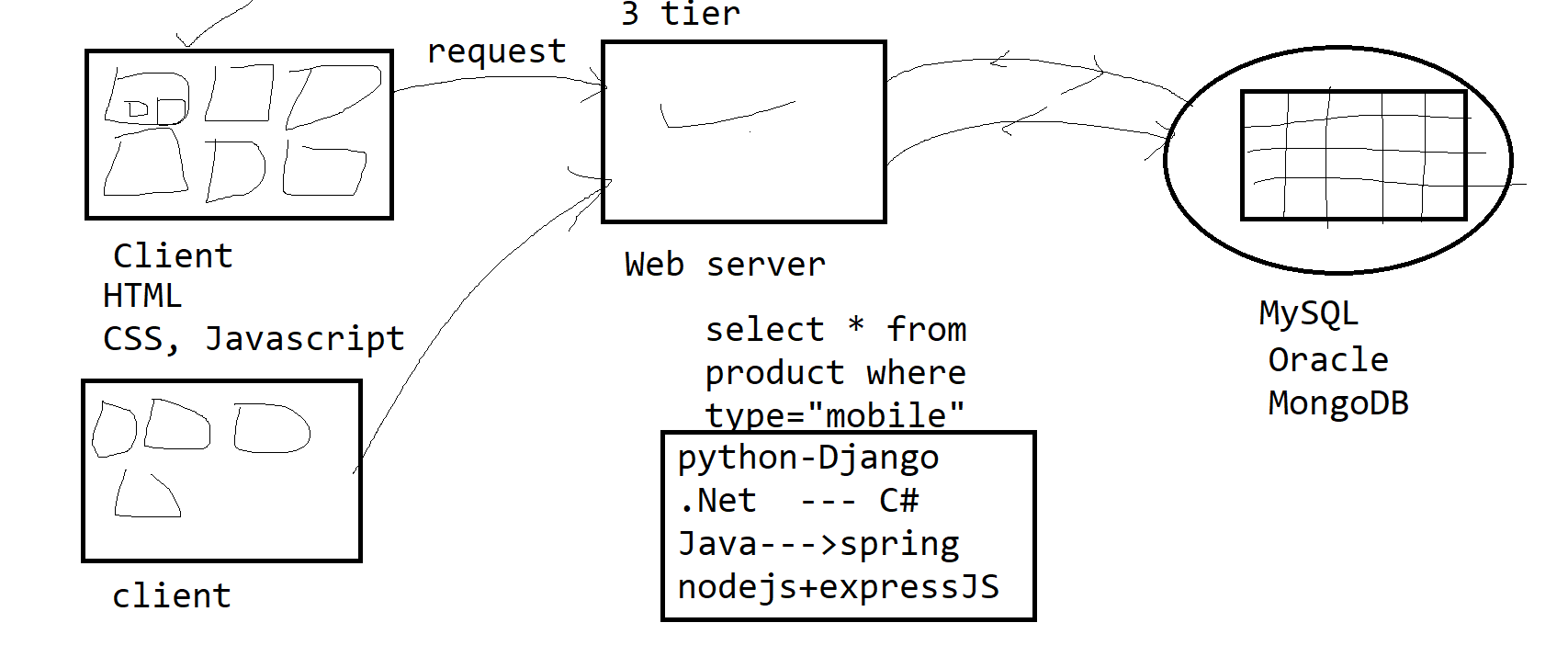
Types of applications

1. Single user
2. Web application
3. Mobile Application
4. Web Services

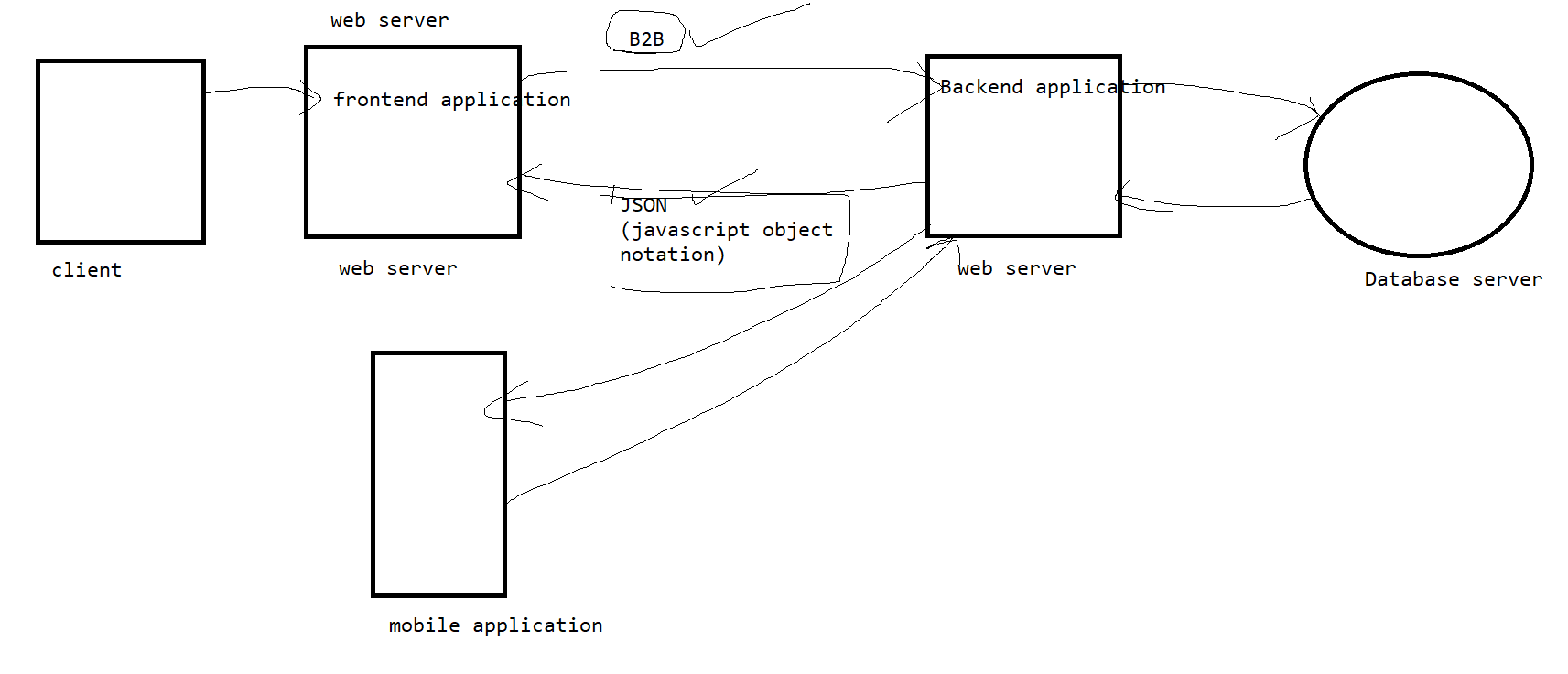
Web Application:

Any application which accepts data from database /web service and converts the data into customer understandable format, by adding presentation logic. (means converts it in to HTML page)



Web service:

The application which gets the data from database and transfers the data to frontend application in either JSON format, XML format, text format.



HTML (Hypertext markup language)

Pages are of two types

1. Static web page --- the page looks same on multiple client machines

Example --- blogs, tutorial site, news paper

Technology --🡪 html, CSS

1. Dynamic web page--- if the page look and feel, or contents are different for different users based on user’s i/p

Examples: flipcart.com, bookmyshow.com

Technology-🡪 html,css,javascript

HTML

1. Every begin tag should have end tag.
2. Tags are predefined tags, and not case sensitive.
3. The file extension has to either html or htm

Editor

Visual studio code

1. To download visual studio code

<https://code.visualstudio.com/download>

download user installer

1. Using exe/msi file install it
2. Got to windows start button--🡪 open visual studio code
3. File🡪open folder--🡪select the folder
4. View>extension> search for live server> install the extension
5. To run the html file -🡪 use go live option which is at the bottom of the screen

Or right click in editor window > open with liver server

|  |  |  |
| --- | --- | --- |
| Header tags | 6 header tags are there, these tags are called as semantic tags | h1, h2, h3, h4, h5, h6 |
| Lists | There are 3 types of lists, in every list there are many list items <li> | Ol --- > type , start  Ul -🡪 type  dl |
| Anchor tag | It is used to display links on the page.  Using this we can open a new file, or a url or section of the same page, or section of the other page, or blank email with email address written in To | A--🡪Href, name, target |
| Body | The contents in this tag will appear in browser window | Body-🡪 bgcolor, background, alink, vlink, link, text |
| table | To display data in tabular format, then use table tag | Table--🡪border, width, cellspacing, cellpadding, bgcolor  Thead  tbody  Tr--🡪bgcolor, align  Th🡪 colspan, rowspan, bgcolor, align, valign  Td -🡪colspan, rowspan, bgcolor, align, valign |
| Br | To add a new line character, use br tag,  It is empty tag |  |
| Hr | To draw a horizontal line | Hr-align, width, size, color |
| Img | To display image on the screen | Img--🡪 border, vspace, hspace, width, height,alt |

|  |  |  |  |
| --- | --- | --- | --- |
| Product id | Product name | quantity | price |
| 1 | lays | 40 | 40 |
| 2 | nachos | 50 | 150 |
| <th colspan=”4”> Product details <th> | | | |
| Product id | Product name | quantity | price |
| <td rowspan=2> 1</td> | lays | 40 | 40 |
|  |  |  |
| nachos | | 50 | |

Types of tags

|  |  |
| --- | --- |
| Empty tag | The tag which never contains any data between begin and end tag  Examples: <br/> , <hr/> <input/> |
| Semantic tags | The tags which gives extra information about the contents, are called as semantic tag  Examples:  H1, h2,…h6, header, footer, nav, section, article,main. |
| tag | Example  P, div, table |
| Auto closable tags | The begin is closed automatically when the next occurrence of the same tag is written.  Examples -🡪 tr, li |

Form tag attribute

|  |  |
| --- | --- |
| method | Get-> sends the data to server via query string, the data will be transferred via header of request object. It is not secure data transfer because the data will be visible in the browser url bar  (Querystring-🡪 it is is string which is concatenated to the url, separated by ?, it contains key ,value pairs separated by &)  Post-🡪 to send data via body part of request object, it is secure way of data transfer because data will not be visible on the screen  When the data you want to transfer to the server is sensitive or if size of data is very big then use post method  By default value of method attribute is get |
| Action | It is a url of a program on the server, to which you want to send the data |
|  |  |

To practice regular expression

<https://regex101.com/>

CSS3(Cascading styles sheet)

Rules for formatting

Selector{

Property:value;

Property:value;

}

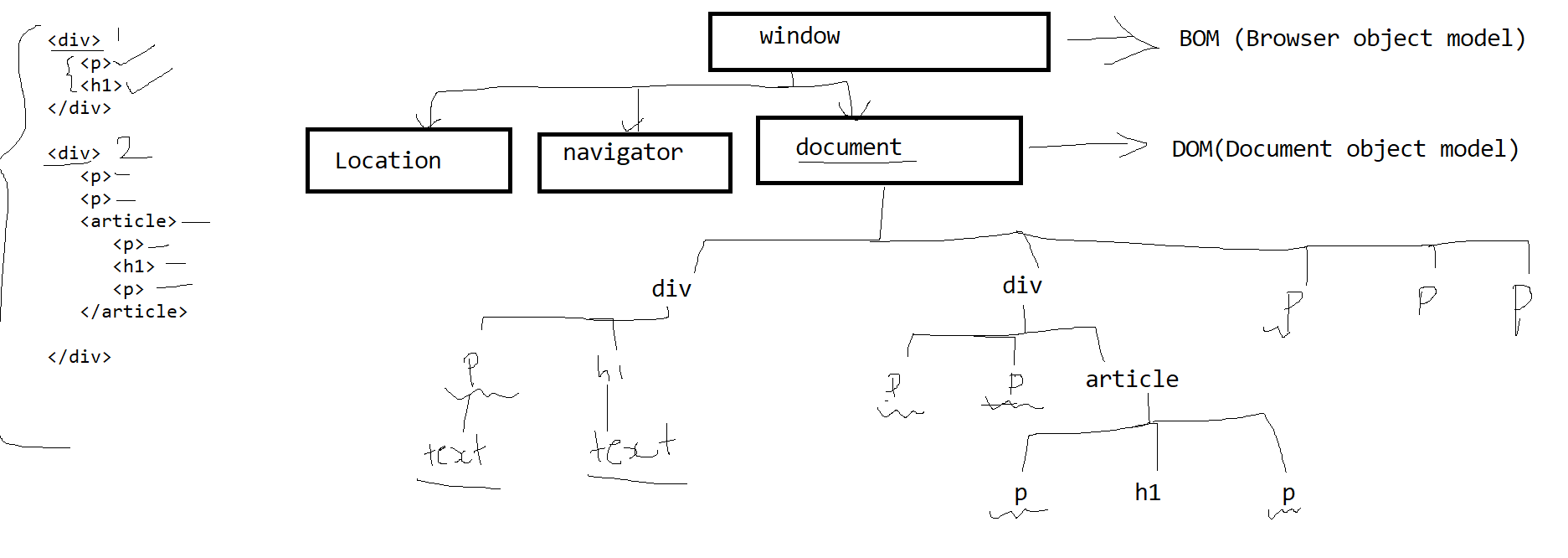
Adding style attribute in the tag

Css are of 3 types

1. Inline- if the rules of css are added in the begin tag, then it is called as inline css, It is useful if you want to apply the style only to one tag
2. Embedded- if the rules of css are added in the style tag, inside head tag, then it is called as embedded css, It is useful if you want to apply the style to multiple tags within same page
3. External css---if the formatting is written in a external file, and then the file is linked with the HTML page. Then it is called as external css.
   1. The extension of the file must be .css
   2. When you want to give same look and feel to multiple pages in the web application, then use external css
   3. To link external css in HTML page use <link rel=”stylesheet” href=”<cssfilename>”>

Types of selector

|  |  |
| --- | --- |
| Tag selector | P, h1, div |
| Universal selector | \* |
| Id selector | #myid |
| Class selector | .myclass |
| Ancestor descendent | Div p |
| Parent child | div>p |
| Attribute selector | Div[name] |
| Attribute value | Div[name=xxx] |
| Pseudo classes | :hover, :first-child :nth-child :last-child |
| a:link | To default color |
| a:visited | The color after visiting the link |
| a:active | The color when the link is active |
| P::first\_line | The first line of p tag |
| P::first\_letter | The first letter in p tag |



10 p tags -- 5 p tags

5 h1 tags --- 2 h1 tags

Position property

Values are ----static, relative, absolute, sticky, fixed

Left, right, top, bottom ---- these properties are applicable only when the position is not static

1. Static value ---left right top and bottom will not work
2. Relative ----left , right, top , bottom will work, it will follow natural ordering
3. Fixed-🡪 tis will keep the element always at the fixed position
4. Absolute-🡪 top, left, right, bottom will be calculated which respect to nearest positioned parent
5. Sticky--🡪 it toggles between relative and fixed, it is relative till the given top and bottom are false, otherwise it will change to fixed

To make the web responsive use media queries

@media (only) <device> and (min-width:<pixel>) and (max-width:<pixel>){

Body{

}

H1{

}

}

JavaScript

It is a scripting language.

To give dynamic effects to the page

AJAX requests can be used.

Validation of a data can be done

Types of javascript

1. External javascript
2. Internal javascript

<script> tag can be added in <head> section as well as <body> section

<script> tag can occur any number of times in the file;

If the code is written inside the function, the it is preferable to add in the head section

Otherwise put <script> tag in the body tag

Javascript uses DOM(Document Object Model)

DOM is in memory representation of HTML page, when HTML page gets loaded in browser, internally it gets converted into a tree structure, whose root is document object, it is called as DOM

Variables in javascripts

1. Variables are case sensitive
2. Variable should start with alphabet, and can be a combination of alphabet and numbers
3. Punctuation characters are not allowed, space is not allowed

To declare variables, we can use var, let and const

|  |  |  |
| --- | --- | --- |
| var | let | const |
| Duplicate declaration is allowed | Duplicate declaration is not allowed | Duplicate declaration is not allowed |
| Var v; initialization is not compulsory | let a; initialization is not compulsory | Const c=23; initialization is compulsory |
| These variables are global, or function scoped | These variables are always block scope | These variables are always block scope |
| Hoisting of variable is allowed | Hoisting of variable is not allowed | Hoisting of variable is not allowed |

Data types in JavaScript

Number--🡪 int or float type numbers

String ---🡪 strings are enclosed in “ “ or ‘’

Boolean---🡪 true, false

Object

The variable is dynamically typed, the type of variable will be decided at runtime based on value assigned to it

Var v=23;

V=”sdjfjklds”

Operators

Arithmetic operators

+, -, \* , /, %, ++, --

Logical operators

&& , || , !

Relational operators

>, < , >=, <=, == , !=, ===, !==

=== and !== are strict checking, it checks value and data type both

A html element can be disabled or it can be readonly

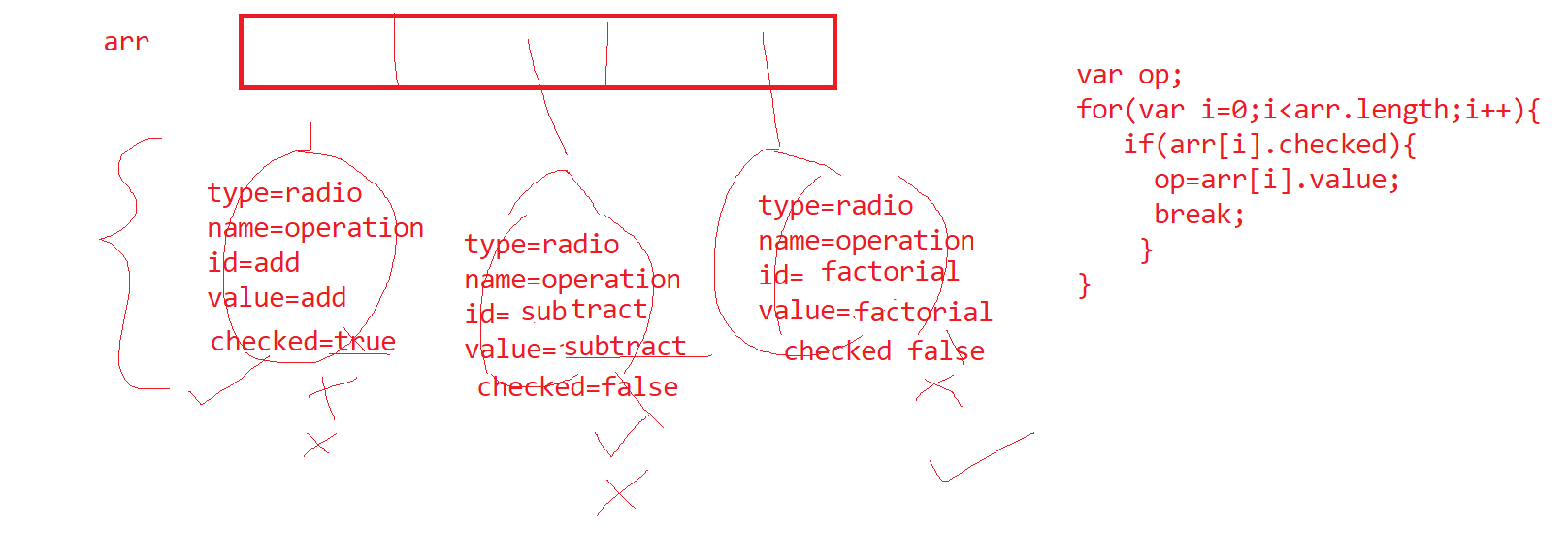
In both cases user will not be able to usethe element in html page

Disabled-🡪 then on submit event, data will not be send to the server

Readonly-🡪 then on submit event, data will be send to the server

Number function

|  |  |
| --- | --- |
| isNaN() | To check whether data contains minimum one alphabet, then this function will return true, but if all are digits then it returns false |
| isFinite() | To check whether data contains all digits, then this function will return true, but if it contains minimum one alphabet , then it returns false |
| Math.floor() | It will remove all the digits after decimal point and keep the number as it is |
| Math.ceil() | It will remove all the digits after decimal point and shows the next number |
| Math.round() | It will keep the number of digits after decimal point based on precision and shows the next number if the digit> 5 otherwise keep it as it is |
| Math.truncate() | It will keep the number of digits after decimal point based on precision and shows the same number even if the digit> 5 or not |
| Math.random() | To generate random number between 0 and 1 |



1. Rest parameter (…) it is added in ECMA 6 version to use variable number of parameters in function.

String functions

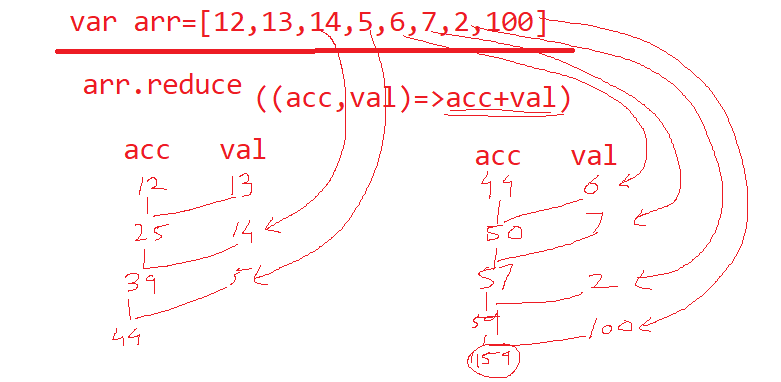
|  |  |
| --- | --- |
| charAt | To retrieve the character at the given index position |
| toUpperCase | To convert String in uppercase |
| toLowerCase | To convert String in lowercase |
| split | To break the string in parts at some delimiter |
| replace | To find and replace, it finds the pattern and replace it with new string |
| match | To find the pattern, returns array of all occurrences |
| Search | To find the pattern, and display the position of first occurrence |
| slice | To get the portion of the string |

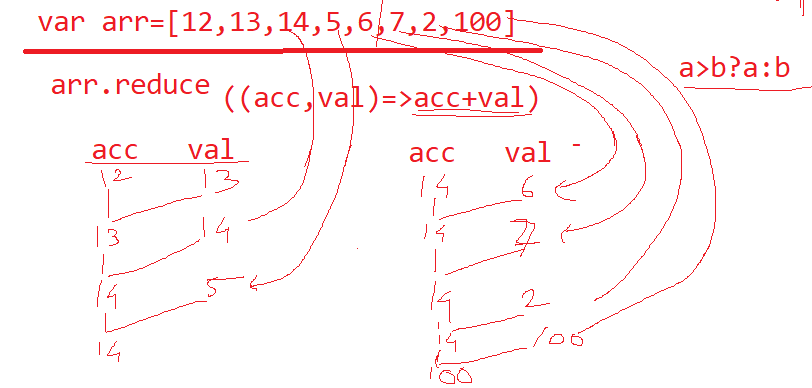
1. Array functions

|  |  |
| --- | --- |
| indexOf | It finds the index position of the first occurrence of the given value |
| Push | It adds the element at the end of array |
| Pop | It removes the element from the end of array |
| Shift | It removes the element at the begining of array |
| unshift | It adds the element at the begining of array |
| Splice | It is used for add element at the particular position, delete the element from the given position, replace the elements at the given position by new values |
| Sort | Sort the array in ascending/descending order, by default array gets sorted based on ascii values |
| Reverse | Reverse the array |
| findIndex | Find the index position of the first value which satisfies the given condition |
| find | Find the first value in the array which satisfies the given condition |
| Filter | Find all the values which satisfies the given condition |
| Map | It maps every value in the array to new value based on given expression |
| Reduce | It reduces all the values in the array to a single value |

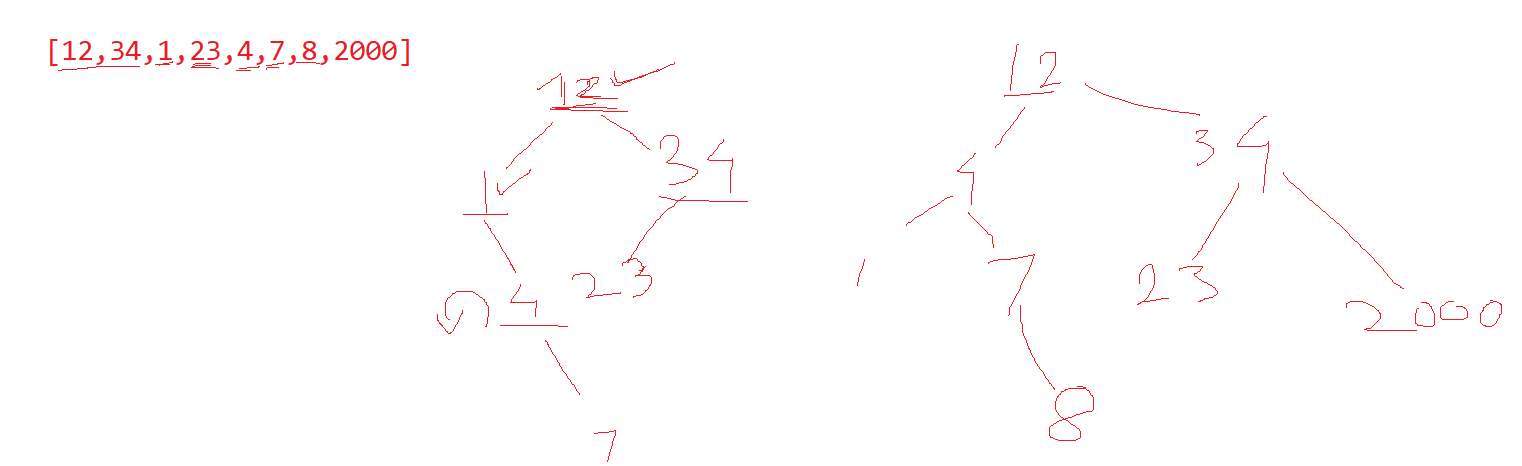
Spread operator

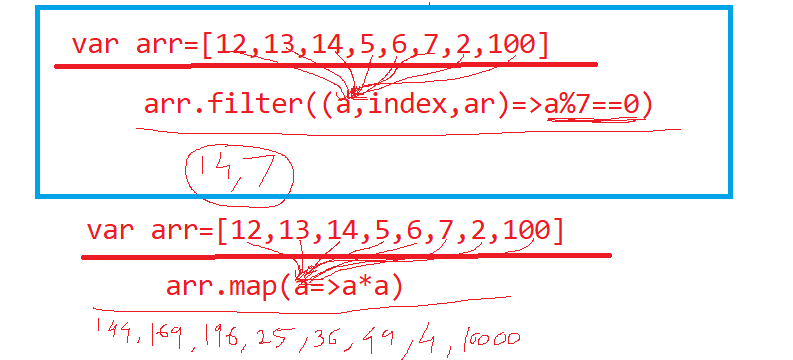
… --🡪 spread operator



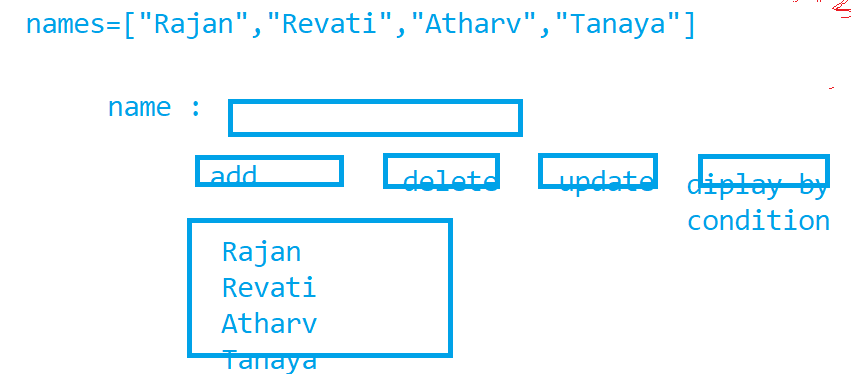


Sorting

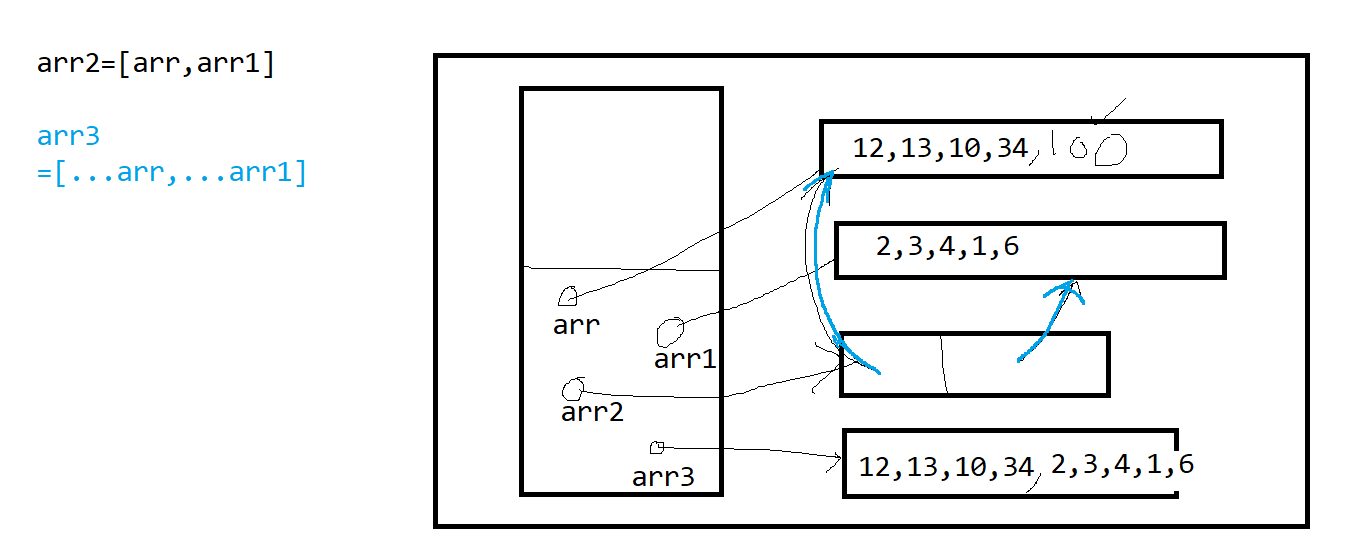


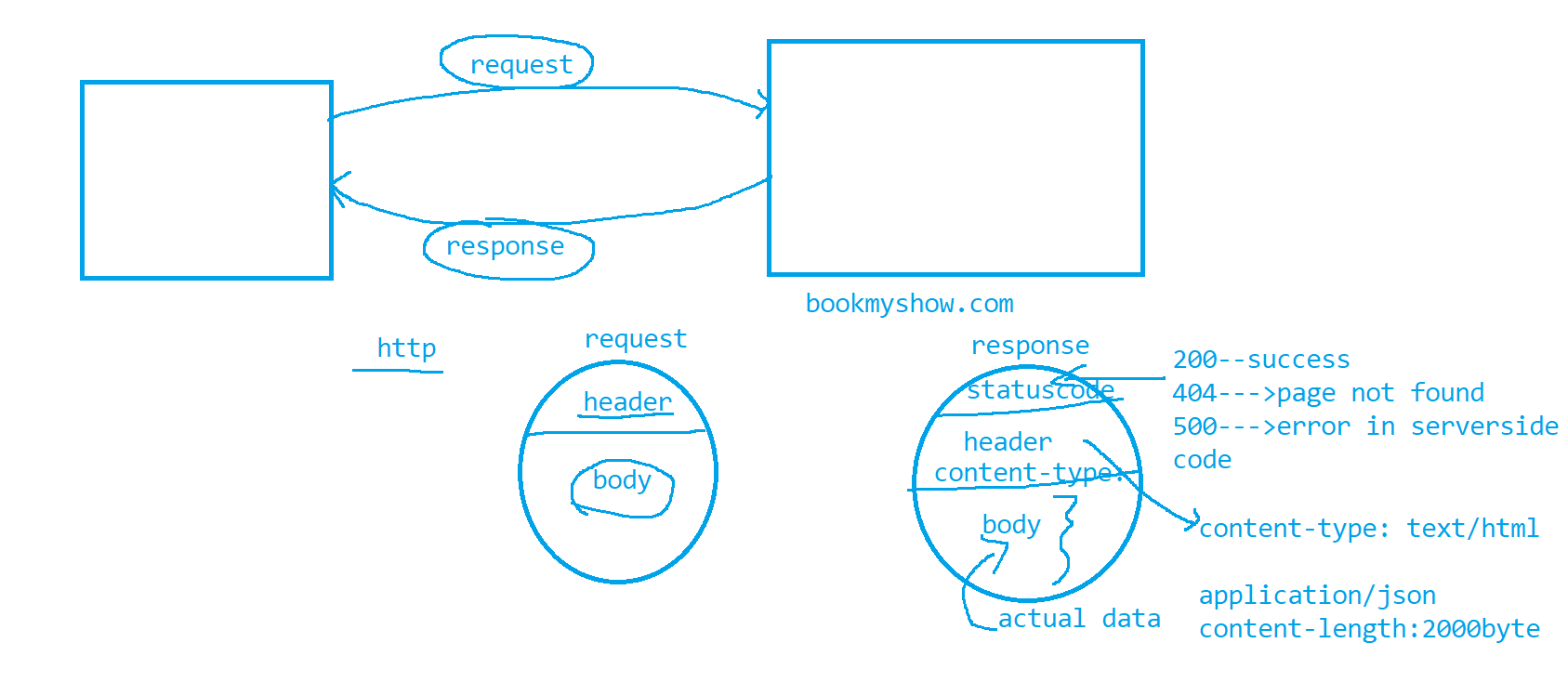


Design the application to perform crud operation on names array



Spread operator





* http protocol is used to transfer data between client and server.
* It is stateless protocol
* Client sends the data to server via request object
* Server sends o/p to client via response object
* When server receives request it reads data from request object and writes o/p in response object and sends to client. Data received via request object will be remembered by server, till it sends the response, hence http protocol is called as stateless protocol
* If you login to some site , and sends multiple requests to server, and the logs out or closes browser window, it will be considered as one session
* So if the data given by user, we need to remember till session ends then developer has to use techniques to retain the data.
* If you want to store the data on the server, then use session object
* If you want to store the data on client machine then use cookies/ localStorage/sessionStorage
* data stored in the sessionStorage will remain available, till you closes the browser window, once you close the window, the data from sessionStorage will be automatically deleted
* data stored in the locaStorage will remain available, till you explicitly delete it
* in developers tool, localStorage and sessionStorage values can be seen in application tab
* localStorage and session Storage objects are browser specific.
* Maximum size for cookies is 4kb, and for localStorage and sessionStorage it is 5 MB

JQuery

To download jQuery

<https://jquery.com/download/>

|  |  |
| --- | --- |
| Tag slector | P, div, input |
| Id selector | #myid |
| Class selector | .myclass |
| Attribute selector | P[name] |
| Attribute value | P[name=myp] |
| Anscestor descendent | Div p |
| Parent child | div>p |
| pseudoclasses | The first child --🡪 p:first-child |
|  | Last-child---🡪p:last-child |
|  | Nth child---🡪p:nth-child(3) |
|  | Even position p:even |
|  | Odd position p:odd |
|  | P[id^=my] --- will select all p tags with id attribute starts with my |
|  | P[id$=t] --- will select all p tags with id attribute ends with t |
|  | P:contains(“para”) will select all p tags which has para somewhere in innerHTML property |

AJAX and JSON

The applications which retrieves data from database and transfers data along with presentation logic(HTML), is called as web Application, these are used for B2C communication

Examples-🡪 amazon, flipkart, makemytrip.com,bookmyshow.com

The application which sends data in the form of XML or JSON to other applications, then it is called as web service

It is used for B2B communication.

In B2B communication since heterogeneous applications communicate with each other, so we need some common format which is known to all the technologies, hence the data transfer happens in XML, JSON format.

XML stands for Extensible markup language

JSON stands for Javascript Object notation

|  |  |
| --- | --- |
| Product data in XML format | Product data in json format |
| <products>  <product>  <pid>11</pid>  <pname>shoes</pname>  <qty>11</qty>  <product>  <product>  <pid>12</pid>  <pname>bags</pname>  <qty>234</qty>  <product>  </products> | [  {  “Pid”:11,pname:”shoes”,”qty”:11  },  {  “Pid”:12,pname:”bags”,”qty”:234  }  ] |

Web Services are of 2 types

1. SOAP web services🡪

These services were using WSDL(web service definition Language) files,

The data transfer was only in XML format

These applications uses SOAP(simple object access protocol) protocol

1. REST(Representational state transfer) webservices

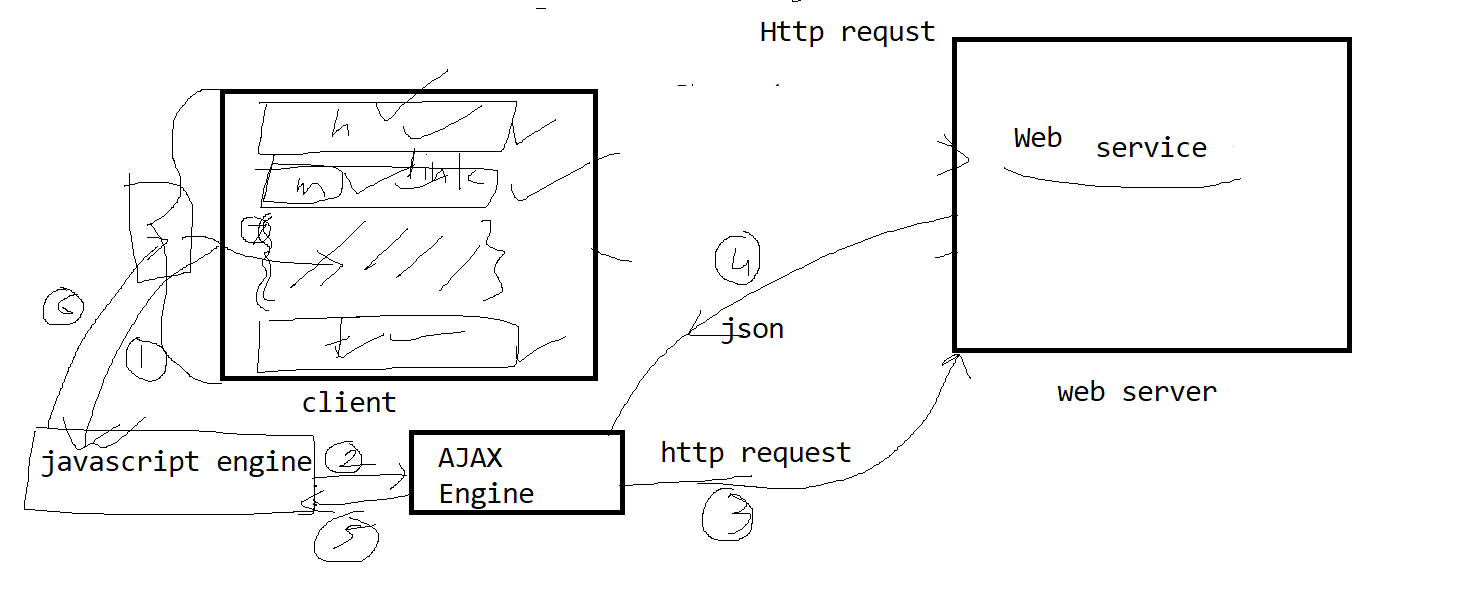
The data can be transferred in either XML,JSON or text format

It uses http protocol

It accepts 5 types of requests.

1. GET----Read data
2. POST-🡪Create data
3. PUT--🡪 update data
4. DELETE--🡪 delete data
5. PATCH-🡪 for partial update

AJAX request



Step by step for AJAX

1. Javascript XMLHttpRequest(XHR) object will get created
2. Configure the object
   1. Save the url to send the request.
   2. Decide the method for sending the request.
   3. Set the headers if any are needed.
3. Handle the event onReadyStateChange

readyState-🡪 1 -🡪 XHR object is ready

-🡪 2 🡪 request is send to the server

-🡪3 -🡪 server started sending data

-🡪 4 -🡪 the data received

JQUERY function

$.ajax({

async:true,

type:’GET’,

url: ‘<https://jsonplaceholder.typicode.com/posts/>’,

success: function(result){

}

error:function(err,xhr,status){

}

})

Node js

1. Install nodejs on your machine

<https://nodejs.org/en/download>

1. Once download msi file , follow step by step process to install it
2. Open cmd prompt and give following command.

node -v

Nodejs is used for serverside javascript programming. Using nodejs you can write web application or web services.

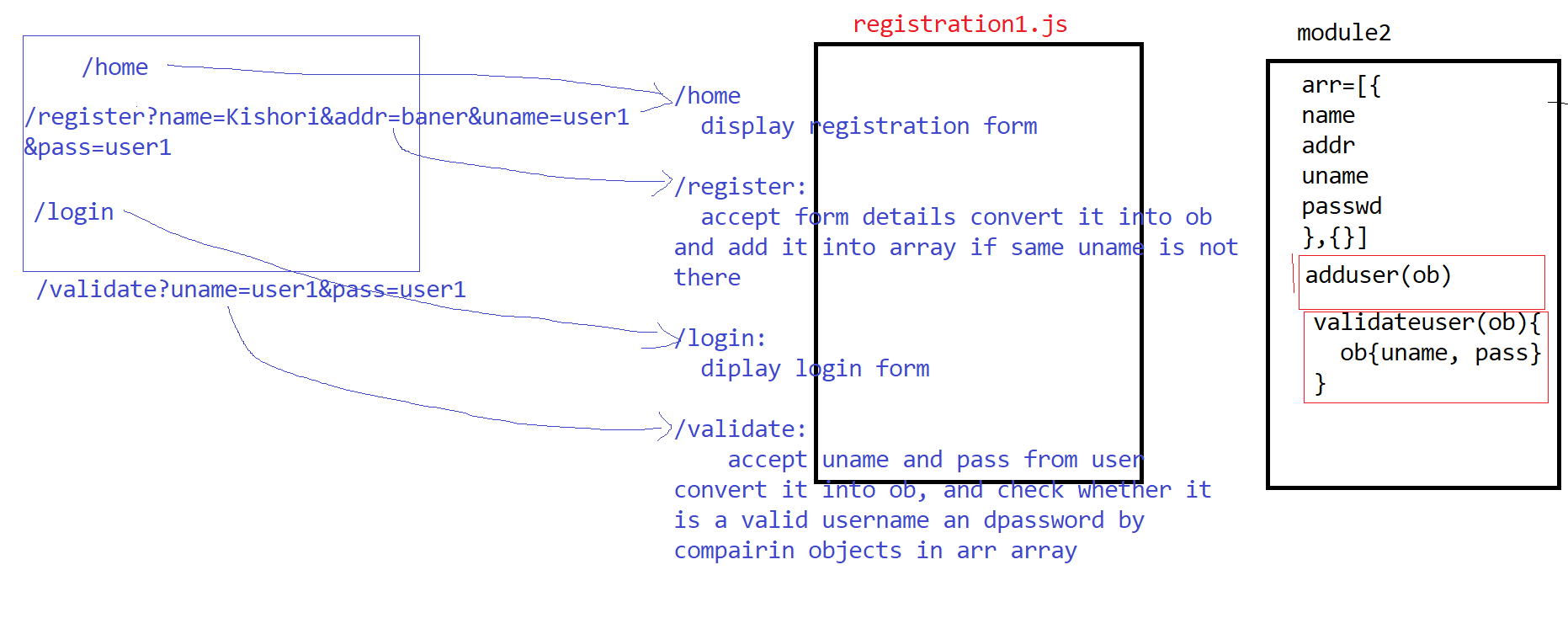
Nodejs uses asynchronous programming, so that request for multiple users can be processed parallelly.

It uses Google chrome’s V8 javascript engine

Nodejs provides REPL (Read Evaluate print loop), to open REPL use command

C:\system>node

Registration application



To use express js

1. Download expressjs on your machine.
2. Download body-parser
3. Download mysql driver to connect to mysql database
4. If you want to connect to mongodb, then use mongoose

Step1: create package.json

npm init

Step 2:

Install expressjs and body-parser locally

npm install express --save

npm install body-parser –save

npm install mysql –save