

Client server architecture: client programs interact with the web servers.

Network Protocol: HTTP; designed to move quickly from document to document.

Addressing system (URL): Uniform resource Locators

Markup Language: HTML

Web browsers and Web servers communicate according to a protocol known as HTTP(HyperText Transfer Protocol) – current version is 2.0.

A web server can:

* Receive and reply to HTTP requests.
* Retrieve documents from specified directories
* Run programs in specified directories
* Handle limited forms of security

A web server cannot:

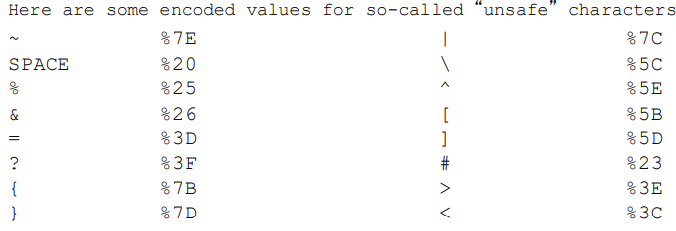
Know the contents of a document, links in a document, images on a document or whether a particular file, e.g. a gif file, is in the correct format.

URL:

A mechanism whereby an internet resource can be specifies in a single line of ASCII text.

Description:

* Scheme followed by a colon: - http:, ftp:, mailto:, etc.
* Double slash(required for http, ftp)
* Internet domain name, eg, [www.usc.edu](http://www.usc.edu)
* Port number(eg, www.usc.edu:8081) – optional
* Path (eg, /pub/docs)



Timeline:  
December 1990 – Tim Berners Lee releases WWW Prototype on NeXt Computer

1993 – Mosaic browser from NCSA is released.

1995 – Microsoft Windows 95 and Internet Explorer 1.0 released

1995 – JAVA was released.

1998 – Google is started

1999-2001 – Internet bubble

2004 – Firefox 1.0 is released

2005 – Youtube is founded

2008 – Google Chrome 1.0 is released.

HTML

Used to processing, definition, and display of data.

It is a tag based language

Developed by Tim Berners-Lee at CERN

Version:

HTML – 1991 (add highlighting and images)

HTML2.0 – 1995 (Forms)

HTML3.2 – 1997 (Tables)

HTML4.01 – 1999

HTML5 – 2014 (APIs and vocab)

HTML can be used to:

* Display and format text
* Display graphics
* Pointers to other html files
* Pointers to files containing graphics, digitized video and sound
* Forms that capture information from the viewers.

HTML Element:

Basic HTML node that adds semantics.

Two types:

* Without content: self-closing start tag.
* With context: start tag, content, end tag

Some elements do not require end tags, eg: - <p>

Some elements do not require content, eg: - <hr>

HTML Tag:

Composed by the name of the element, surrounded by angle brackets.

End tag has a slash after the opening angle bracket.

HTML Attribute:

Specified inside start tag

Controls element properties/behaviour or provides metadata.

Browsers like Egde, Safari, Firefox, Chrome are tolerant browsers:

* They ignore markup they don’t understand.
* They do not insist that the html document begin and end with <HTML> and contain doctype
* <HEAD> and <BODY> tags are not required.

Comments in HTML Start with <!-- and end with -->. White space is permitted between the – and the closing angle bracket >. It is not permitted between the opening angle bracket, exclamation point, and the --.

Attributes:

* Id, assigns a unique name to an element.
* Class assigns one or more names to an element.
* Lang is a language code that identifies a natural language spoken, written, or otherwise used.
* Title is a short description of the body
* Style, inline display information
* Bgcolor, background color
* Events include: onload, onunload onclick, ondblclick, onmousedown, onmouseup, onmouseover, onmousemove, onmouseout, onkeypress onkeydown, onkeyup
* Deprecated elements in include: background, text, link, vlink, alink.

The size and look of the text can be changed using CSS, so don’t use HTML heading tags solely to change the size of the font.

HTML List:

* Unordered list(<ul>)
* Ordered list(<ol>)
* Definition list(<dl>)
* The Menu(<menu>)

Format:

<ul>

<li></li>

</ul>

OR

<ul><li><li><li></ul>

Definition list and menu list were deprecated in html4 were reintroduced in html5.

HTML Table:

* <table> tag used to define a table
* <caption> tag is used to label a table. Attributes: align = top, bottom, right, left.
* <th> and <td> tags identify a table header cell and table data cell. Both are same except hat headers use bold font and are centered.
* <tr> is a tag that identifies a container for a row of table cells. Table cells can be text, lists, images, forms, figures, or even tables.

HTML uses the Universal Character Set(UCS). Latest specification: - Unicode 15.1.0. It includes Unicode Emoji.

HTML Hyperlinks:

An anchor tag is a way to designate a link to another document or specific place in the same document.

The link location is given by the required href(hypertext Reference) attribute. Link destination can be relative or absolute.

Rules or anchor name:

* Uniqueness: Anchor names must be unique within a document. Anchor names must differ only in case may not appear in the same document.
* String matching: Comparisons between fragment identifiers and anchor names must be done by exact (case-sensitive) match.

Title attribute of anchor tag helps you add information about the nature of the link.

URIs typically consist of three pieces:

* The scheme of the mechanism used to access the resource.
* The name of the machine hosting the resource.
* The name of the resource itself, given as a path.

HTML Link:

<link> appears only in the HEAD.

It conveys a relationship between two documents.

It provides a links to:

* Alternate versions of a document, written in another human language.
* Alternate versions of a document, designed for different media.
* The starting page of a collection of documents.
* Style sheets and ‘media queries’ used in Responsive Web Design.

Images:

Image formats supported by web browsers:

* X-pixelmaps
* Graphic Interchange Format(GIF)
* Joint Photographic Experts Group(JPEG)
* Portable Network Graphics(PNG)
* Scalable Vector Graphic(SVG)
* WEBP
* AVIF
* JPEG XL
* HEIF

<IMG> attributes:

* Src: the HREF or name of the image
* Srcset and sizes: HREF ans sized for different screen sized and types (align = top, bottom, middle)
* Height and width to control the dimensions of the image
* Alt to replace an image with text, if the image is unavailable or a text browser is used.

<META> element:

Allows you to insert Name/Value pairs describing document properties.

Eg: - <META name = “description” content = “ajdbiewgvfkc”>

Also used to move a web page to a new site.

CSS

Cascading Style Sheets

Not a markup language.

CSS 1 – 1996

CSS 2 – 1998

CSS 2.1 – 2011

CSS 3 modules – 2012-22

CSS 3 – ongoing

Can be used in 3 ways: inline, in the <style> element, external file using <link> tag

Class attribute assigns a name to one or more element.

ID attribute is used for one element. It can be used only once in the entire document.

Values assigned to class and ID are case sensitive.

A style for tag.class has higher precedence than one for .class, which has higher precedence than one for the tag itself.

Inline CSS has more precedence than <style> element, which has more precedence than external CSS.

DOCTYPE:

It makes your website look the same in all browsers. For HTML5: - <!DOCTYPE HTML>

JSON

JavaScript Object Notation is a lightweight, text-based data interchange format

Used for representing data structures in web apps.

Supports strings, numbers, Boolean, null, arrays and objects

Key-value pairs within {} for objects and [] for arrays.

JSON Objects Can be used to represent struct, record, hashtable, object.

JSON Array can have first index as 0 or 1.

It transmits data between server and client.

It integrates with various programming languages and frameworks.

Its main application is in AJAX web application programming, where it serves as an alternative to the use of XML Format.

JSON was based on subset of JavaScript (Standard ECMA-262 – on its 15th edition now).   
JSON is language independent data format. It is developed by Douglas Crockford.

Working on JSON involves 3 steps:

* Client side: the contents of a JSON file, or the definition of JSON data is assigned to a variable, and this variable becomes an object of the program.
* Server side: a JSON file on the server side can be operated upon by various programming languages, including PHP and JAVA thanks to parsers that process the file and may even convert it into classes and attributes of the language.
* Data exchange: Loading a JSON file from the server may be accomplished in JavaScript in several ways:
  + Directly including the file into the HTML page, as a JavaScript .json external file.
  + Loading by a JavaScript command
  + Using fetch() or XMLHttpRequest()

To convert JSON into object, it can be parsed with JSON.parse() or it can be passed to the JavaScript eval() function(insecure way).

Sending a file to the server can be accomplished by fetch() or XMLHttpRequest().

The file is sent as a text file and processed by the parser of the programming language that uses it.

JavaScript eval():

It is a function property of the global Object, and evaluates a string and executes it as if it was JavaScript code.

JSON Basic Data type:

* String(double-quoted Unicode with backslash escaping)
* Numbers(integer, real or floating point)
* Booleans(true and false)
* Object(collection of key:value pairs, comma-separated and enclosed in curly brackets)
* Array(an ordered sequence of values, comma-separated and enclosed in square brackets)
* Null(a value that isn’t anything)

• A JSON decoder must accept all well-formed JSON text

• A JSON decoder may also accept non-JSON text

• A JSON encoder must only produce well-formed JSON text

Disadvantages:

JSON doesn’t have namespaces

• JSON has no validator

– Every application is responsible for validating its inputs

– However, there exist standards like JSON Schema

• JSON is not extensible

– But it does not need to be

• JSON is not XML

– But a JavaScript compiler is a JSON decoder

JSON.parse() is safer than eval().

JavaScript

JavaScript is a simple, interpreted, programming language with prototype OOP inheritance.

It is case sensitive. It ignores spaces, tabs, newlines(can be minified)

Semicolon is optional

C and C++ comments are supported.

Runtime usually consists of:

* JavaScript engine and Event loop
* Web APIs

JS Engine provides:

* JS Interpreter + JIT Compiler
* Sandboxing
* WebAssembly support
* API Bridge

JS Engine types:

V8

SpiderMonkey

JavaScriptCore

Chakra

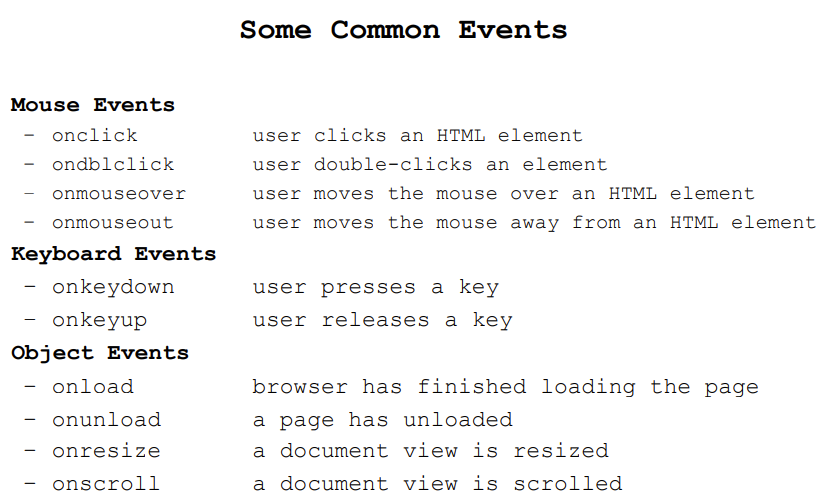
JS was developed by Brendan Eich at Netscape in May 1995.

Defined in HTML using <script> tag.

<script language=”JavaScript”>

//Script

</script>



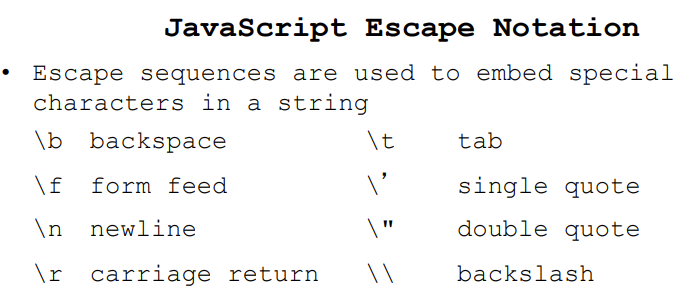
What JS Programs do:

* Control Web page appearance and content
* Control web browser, open windows, test for browser properties
* Interact with document content
* Retrieve and manipulate all hyperlinks
* Interact with the user, sensing mouse clicks, mouse movements, keyboard actions.
* Read/Write client state with cookies.

JS strings are immutable. Once created, they can’t be changed. You can search a string and extract substrings, but cannot modify it.

Properties of string:

* Str.length
* Str.toLowerCase()
* Str.toUpperCase()
* str.indexOf(searchstring[,startindex])
* str.charAt(index)
* str.substring(indexA, indexB)



Variable declaration: - var varname = value;

Ways to declare variable:

* Var
* Let
* Const

Objects:

Object is a list of zero or more pairs of property names and associated values of an object, enclosed in curly brackets.

Objects can be nested within objects.

The dot ”.” property is used to access the value of an object’s property or to assign it a value.

Predefined JS Objects:

* Array
* Boolean
* Date
* Function
* Math
* RegExp
* String
* Map
* Set
* WeakMap / WeakSet
* Symbol
* Promise
* Proxy, Reflect
* Atomics
* ArrayBuffer, DataView, SharedArrayBuffer
* TypedArray
* FinalizationRegistry

JS Popup Boxes:

* alert()
* confirm()
* prompt()

Window object is the highest-level built-in JS Object.

Some properties of the Window object include:

* window.closed – a boolean indicating if the window is closed or not
* window.history – returns the history object
* window.location – returns the location object
* window.navigator – returns the navigator object
* Window.parent – returns the parent window of the current window
* Window.self – returns the current window
* Window.status – sets the text in the statusbar window

Some methods of the Window object include:

* open() – opens a new window
* blur () – removes focus from the current window
* close() – closes the current window
* focus() – sets the focus to the current window
* resizeBy () – resizes the window by the specified pixels

The open() method includes three top level parameters:

newWin = open(url, name, [features, [replace]]);

3 possible ways to invoke a code to open window:

1. Triggered at page load time:

<body onload=”[open new window code];”></body>

1. Triggered by clicking on a hyperlink:

<a href=”javascript: [open new window code];void 0;”></a>

1. Triggered by clicking on a button:

<input type=”button” onclick=”[open new window code];”>

Navigator:

It is a built-in object that gives properties that describe the browser.

* navigator.appName is a string with the browser name
* navigator.appVersion is a string with the version number
  + to determine the correct version, you may need to convert from string to number; parseFloat returns a number from a string, and ignores any part of the string after the number
* navigator.cookieEnabled determines whether cookies are enabled
* navigator.language returns the language of the browser
* navigator.userAgent returns the user-agent header sent by the browser

Document:

Each HTML Document loaded into a browser window becomes a document object. The document object provides acess to all HTML elements in a page, from within the script.

Some properties of the document object include:

* Document.anchors – returns a collection of all anchors in the document
* Document.applets – returns a collection of all applets in the document –
* Document.body – returns the body element of the document
* Document.cookie – returns all name/value paris of cookies in the document –
* Document.forms – returns a collection of all forms in the document –
* Document.images – returns a collection of all the images in the document –
* Document.lastModified – returns the date/time the document was last modified •

Some methods of the document object include:

* Document.close() – closes the output stream previously opened
* Document.open() – opens an output stream to collect the output from document.write
* Document.write() – writes HTML expressions or JavaScript to a document

Location Object:

It is a reference to the URL of the current document.

Properties: hash, host, hostname, top, status, defaultStatus, window.

No Methods or Event Handlers.

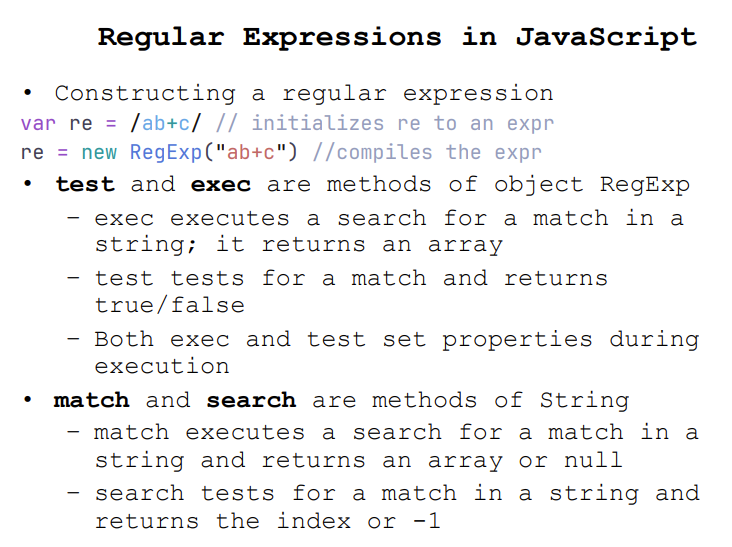
History Object:

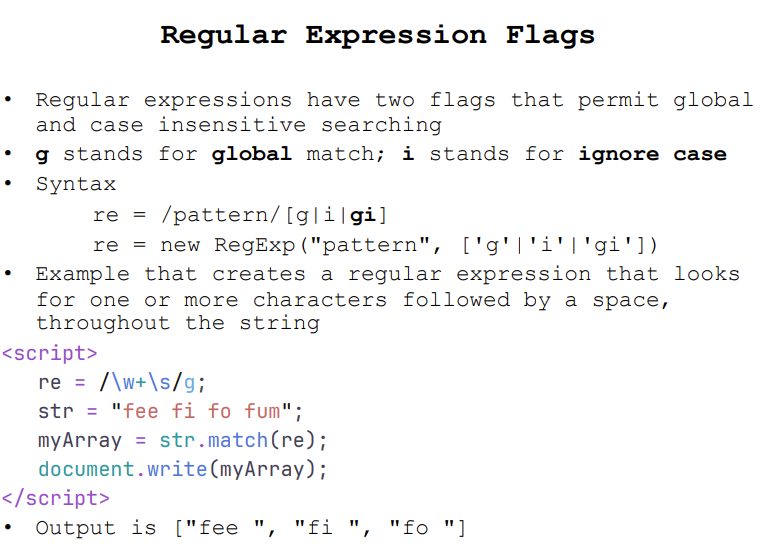
It contains the URL visited by the user within a browser window.

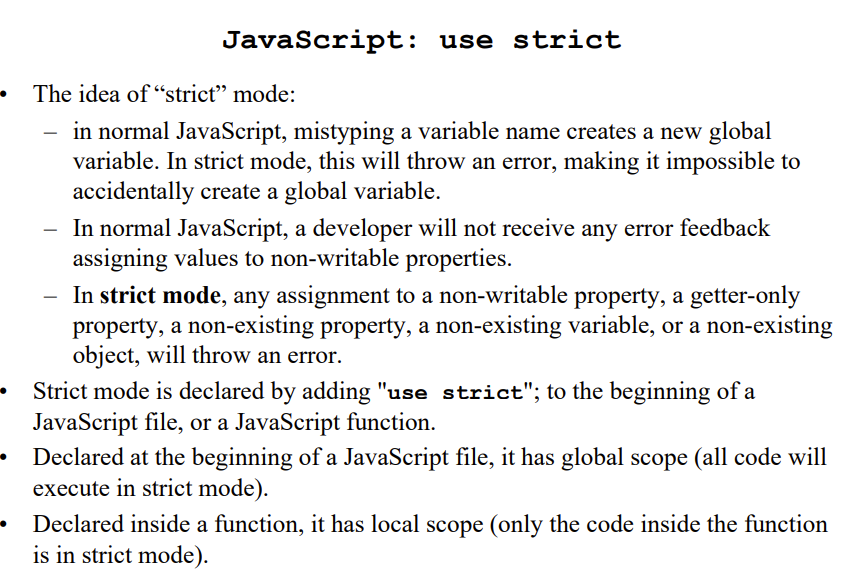


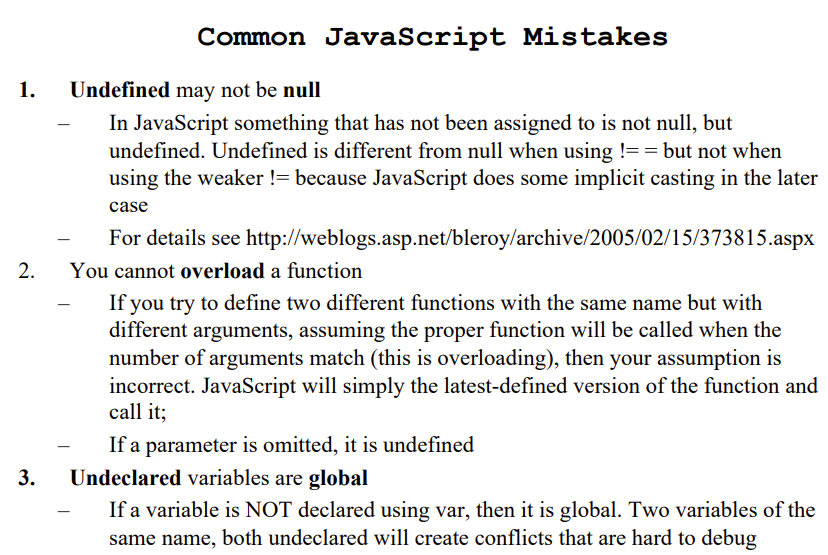
Function is a type of object.

Regular Expressions in JS:









DOM:

Document Object Model is a programming interface for XML documents.

It defines the way an XML Document can be accessed and manipulated. It includes HTML documents.

XML DOM is designed to be used with any programming language and any operating system.

DOM represents an XML file as a tree.

The documentElement is the top level of the tree. This element has one or many childNodes that represent the branches of the trees.

Versions:

* DOM Level 1: concentrates on HTML and XML Document models. Contains functionality for document navigation and manipulation.
* DOM Level 2: adds stylesheet object model to DOM Level 1, defines functionality for manipulating the style information attached to a document, and defines an event model and provides support for XML namespaces.
* DOM Level 3: Core, Load and Save, Validation.
* DOM Level 4(DOM4): Moves some to HTML5.

Document is the root element.

DOM Query Functions:

* Element.querySelector(), element.querySelectorAll()
  + Returns subtree nodes that match css selector.
  + All() version returns NodeList
* Element.getElementById(“sample”)
  + Returns the one location defined by id=sample
* Element.getElementsByClassName(“.myclass”)
* Element.getElementsByTagName(“font”)

DOM Manipulation methods:

* Document.createElement(tagName)
  + Only document can create nodes
* .append(element) / .prepend(element)
  + Prepend adds element to the beginning while append adds element to the end
* .replaceChildren(element1, element2, …)
* .contains(el)

DOM Properties:

* .innerHTML, .outerHTML, .innerText, .outerText
  + Assigns or retrieves a new value to html/text defined by id=””.
* .parentNode, .childNodes, .nextSibling
  + DOM tree navigation
* .classList{.add() .remove() .replace() .toggle()}
* .style.left, .style.color, …properties
  + Assign values to CSS properties

innerHTML:

Elements that do not have both, an opening tag and closing tag, cannot have an innerHTML Property.

The innerHTML property takes a string that specifies a valid combination of text and elements. When the innerHTML property is set, the given string completely replaces the existing content of the object. If the string contains HTML tags, the string is parsed and formatted as it is places into the object.

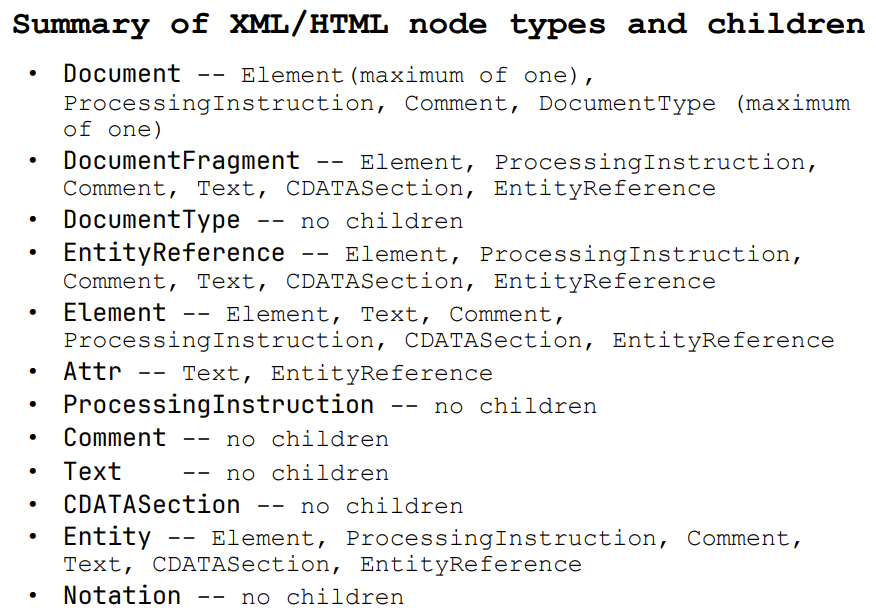
XMLHTTPRequest :

It is used to exchange data with the server.

Use:

* Update a web page without reloading the page
* Request data from a server after the page has loaded
* Receive data form a server after the page has loaded
* Send data to a server in the background

All modern browsers have built-in XMLHttpRequest object.



Python

It is interpreted, object-oriented, high-level programming language

It has high-level built-in data structures

Has dynamic typing and dynamic binding

Supports both procedural and object-oriented paradigm

Variables are case sensitive

No keyword called static is present

Constants are special variables that hold values that should not be changed

Strings are immutable.

Flask

Flask is a lightweight WSGI(Web Server Gateway Interface) web application framework.

WSGI specifies a standard interface between web servers and Python web applications or frameworks.

It is designed to make getting started quickly and easily, with the ability to scale up to complex applications.

Send static file using send\_static\_file or send\_from\_directory

Django

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design.

Django was designed to help developers take applications from concept to completion as quickly as possible.

Django takes security seriously and helps developers avoid many common security mistakes.

Some of the busiest sites on the Web leverage Django’s ability to quickly and flexibly scale

The Web Server

WWW server enables browser requests

It mainly provides:

* Support for retrieving hypertext documents
* Manages access to the web site
* Provides several mechanisms for executing server-side scripts.
  + Common Gateway Interface(CGI)
  + Application Program Interface(API)
  + Direct Module Interfaces(SAPI)
* Provides log files and usage statistics

Things to look for in a web server:

Main features:

* Platform they run on
* Complete support for HTTP 1.1/2/3
* Support for web sockets
* Multithreading, load balancing

Security features:

* Ability to provide IP Address/reverse dns restriction
* Support for secure transactions: TSL/SSL
* Support for client certificate auth

Ability to act as a proxy server

Application web server is a software that typically interfaces one or more databases to convey processed data to and from a presentation layer.

It performs business logic.

Nginx, Apache, Cloudfare, Openresty are on top.

Web Server Features

* Document Root

The Web server has access to a tree of files that it can deliver – the files are created by content providers – files may contain text, images, sound, video, other programs, etc.

The document tree is organized by the web site administrator

The root of the document tree is given to the web server when it starts

* Virtual Hosts

Hosting multiple web sites by the same web server

It uses the host name or IP address to distinguish the

document roots

* Proxy Servers

No matter which operating system or server, you will need to define the

1. location of the server (server root)

2. location of documents to deliver (document root)

3. location of CGI scripts or server-side components to execute

* Authentication / Authorization
* Caching
* CGI Scripting
* Application Program Interface
* File uploading support
* Generic TCP/UDP protocols support
* Load Balancing

Apache Web Server

Apache web server is based on code and ideas developed at NCSA in HTTP version 1.3.

Latest: 2.4.58.

The Apache httpd server

* is a powerful, flexible, HTTP 1.1 & HTTP/2 compliant web server
* implements HTTP/2, but not HTTP/3 / websockets
* is highly configurable and extensible with third-party modules can be customized by writing 'modules' using the Apache module API
* provides full source code and comes with an unrestrictive license
* runs on Windows, Netware, Mac OS X, Linux, and most versions of Unix, as well as several other operating systems

There are two methods for controlling access to directories

1. the file access.conf in the conf/ directory can be used

* placing access directives in a <Directory> block is the preferred method

1. per-directory access rules can be set by a file placed in a specific directory

• the name of the file is set by the directive AccessFileName

• .htaccess is the default name

• however, using .htaccess slows down the server;

• NOTE: initially .htaccess is turned OFF

When the server attempts to retrieve a document, it looks for an access control file in the directory or the parent directory.

The file it looks for is set by the directive AccessFileName

<Limit> controls which clients can access a directory; directives within LIMIT include:

– order in which deny and allow are evaluated

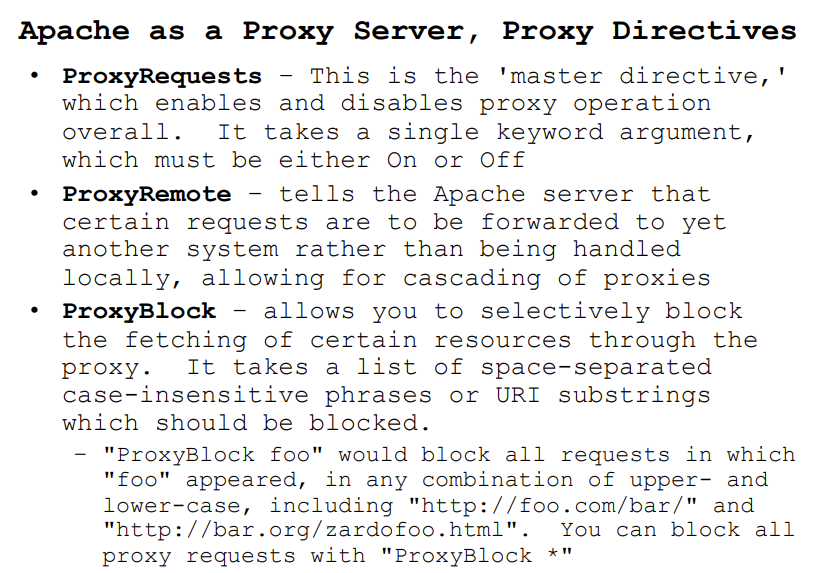
– deny from host1, host2, ...

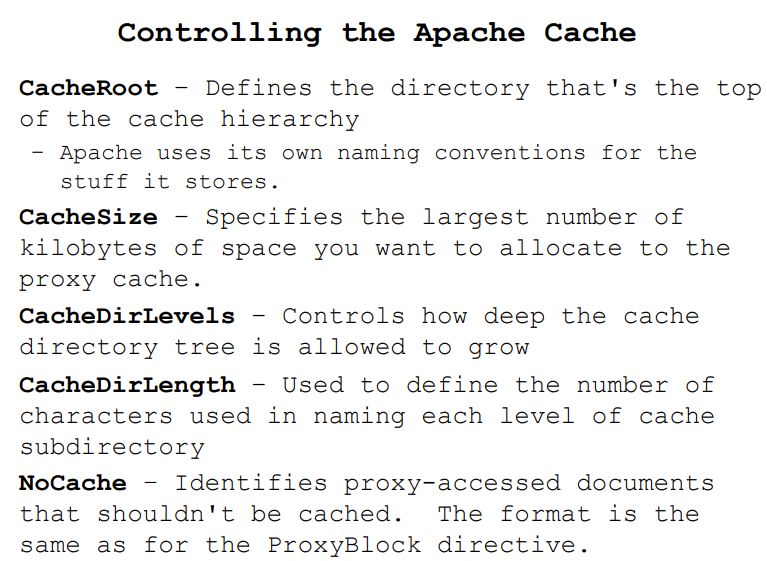
– allow from host1, host2, ...

– require named-users or group-users or AuthUserFile

– referer allows access only from this directory

– satisfy all or any





Forms

Used to create a set of pages that contain fields in which the iewer can select and supply information.

Attribute Specifications

* ACTION=URI (form handler)
* METHOD=[ get | post ] (HTTP method for submitting form)
  + GET is the default; form contents are appended to the URL
  + POST form contents to be sent as payload
* ENCTYPE=ContentType (content type to submit form as)
  + Defaults to application/x-www-urlencoded which returns name/value pairs, separated by &, spaces replaced by + and reserved characters (like #) replaced by %HH, H a hex digit
* ACCEPT-CHARSET=Charsets (supported character encodings)
* TARGET=FrameTarget (frame to render form result in, in HTML4)
* ONSUBMIT=Script (form was submitted)
* ONRESET=Script (form was reset)
* AUTOCOMPLETE (HTML5 ONLY) values completed by browser

The <input> tag also supports the Event Attributes in HTML.

– Window event attributes: onload, onunload, etc.

– Form event attributes: onchange, onfocus, etc.

– Keyboard / mouse events: onkeydown, onclick, etc.

– Drag, clipboard, media events: onpause, onplay, etc.

HTML5 adds more attributes to <select> tag:

• AUTOFOCUS: drop-down list should automatically get focus

• FORM: defines one of more forms the select fields belongs to

• REQUIRED: user is required to select a value before submitting the form

CGI

Common Gateway Interface (CGI) is a mechanism by which programs, called scripts, can be used to

create dynamic Web documents

– Scripts are placed in a server directory often named cgi-bin

– Scripts can deliver information that is not directly readable by clients

– Scripts dynamically convert data from a nonWeb source (e.g., DBMS) into a Web-compatible document

Current version of CGI is 1.1

Anchors to invoke CGI Scripts:

* A remote file
* An executable Script in a cgi-bin directory
* An executable script with arguments

Output from a script to the server could be:

– A document generated by a script

– The type of document could be HTML, plaintext, image, video or audio clip, and many other types

– Instructions to the server for retrieving the desired output elsewhere

– an error indicator

HTTP Protocol

Web browsers and servers communicate using the HypertText Transfer Protocol(HTTP)

HTTP is a lightweighted protocol.

HTTP/3 is the latest version.

HTTP tags al data that it sends with its Multipurpose Internet Mail Extensions(MIME) type.

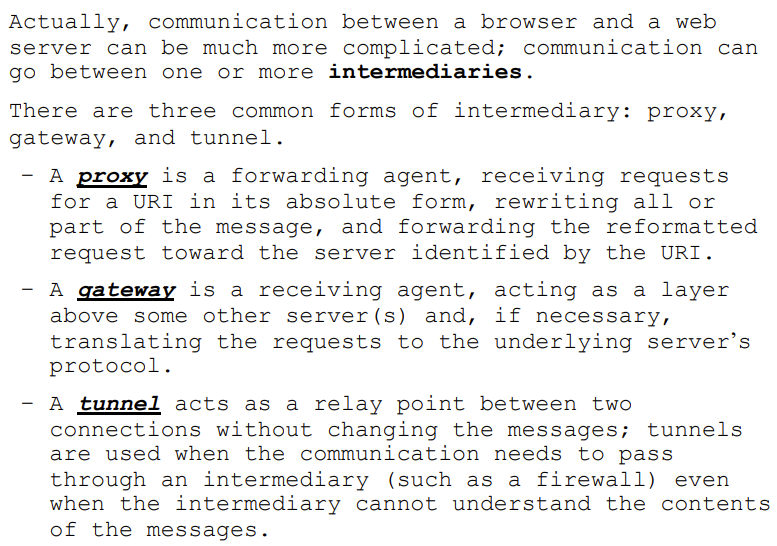
Content-Type: mime type header

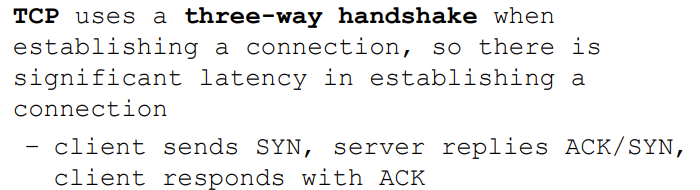
Some important MIME types are

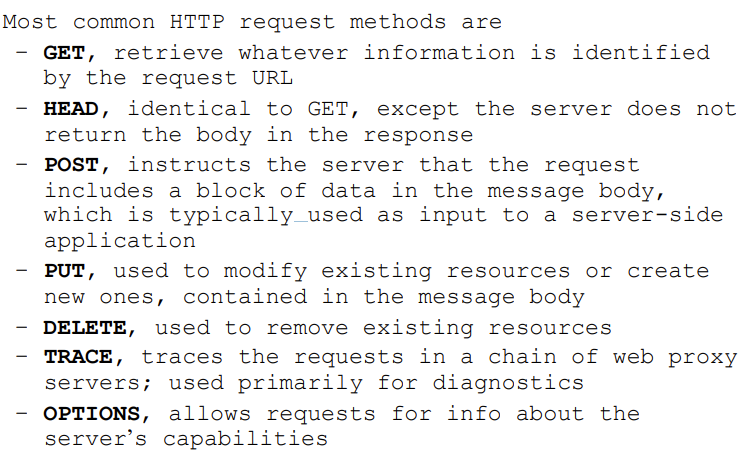
* text/plain, text/html
* image/gif, image/jpeg
* audio/basic, audio/wav, audio/x-pn-realaudio
* model/vrml
* video/mpeg, video/quicktime, video/vnd.rnrealmedia, video/x-ms-wmv
* application/\*, application-specific data that does not fall under any other MIME category, e.g., application/vnd.ms-powerpoint

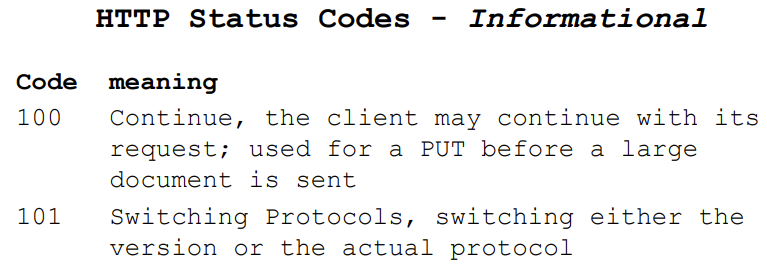
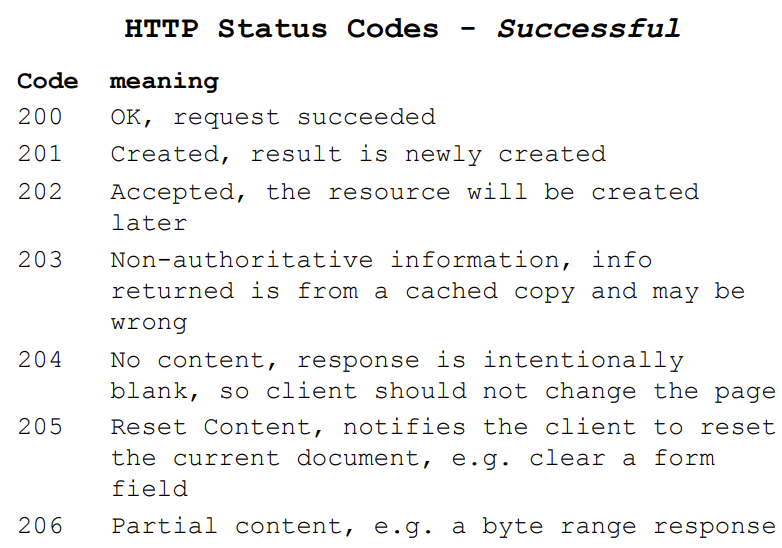
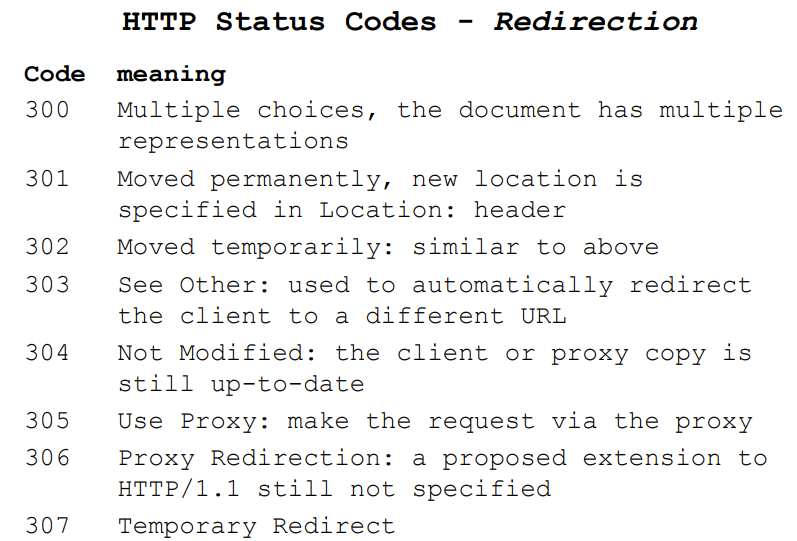
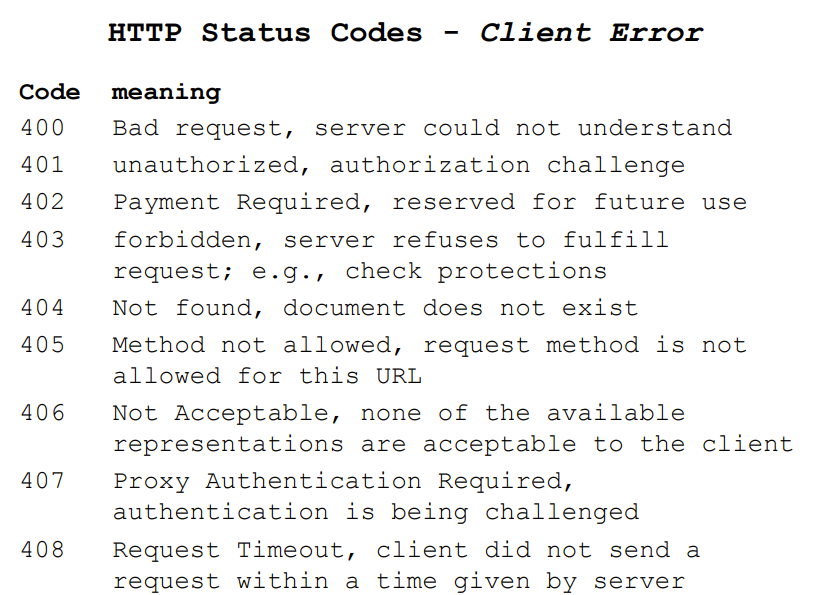
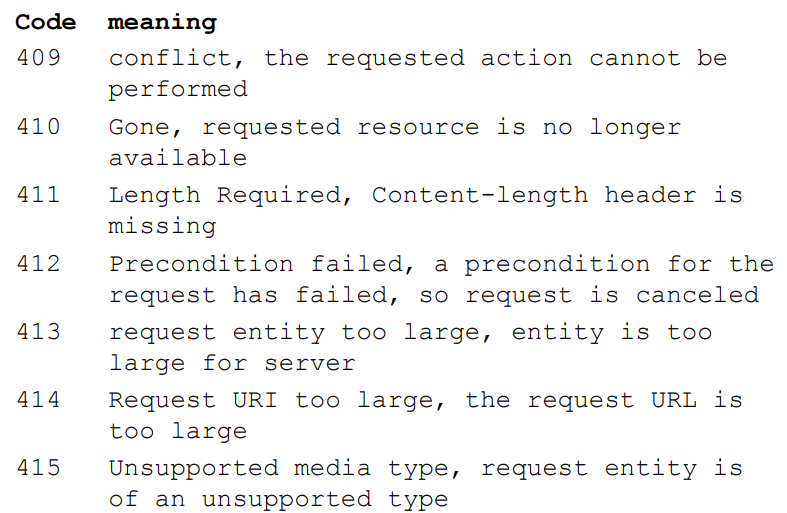
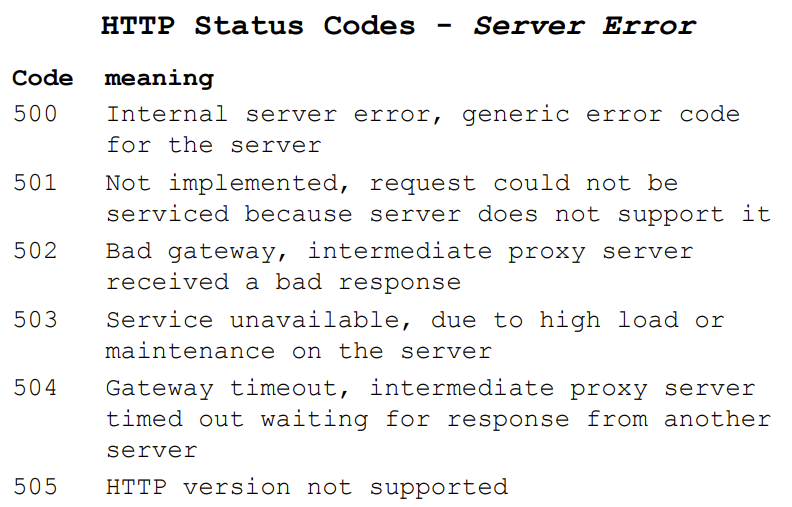
MIME converts data that uses all eight bits into 7-bit ASCII, sends it, and reconverts it at the other end.

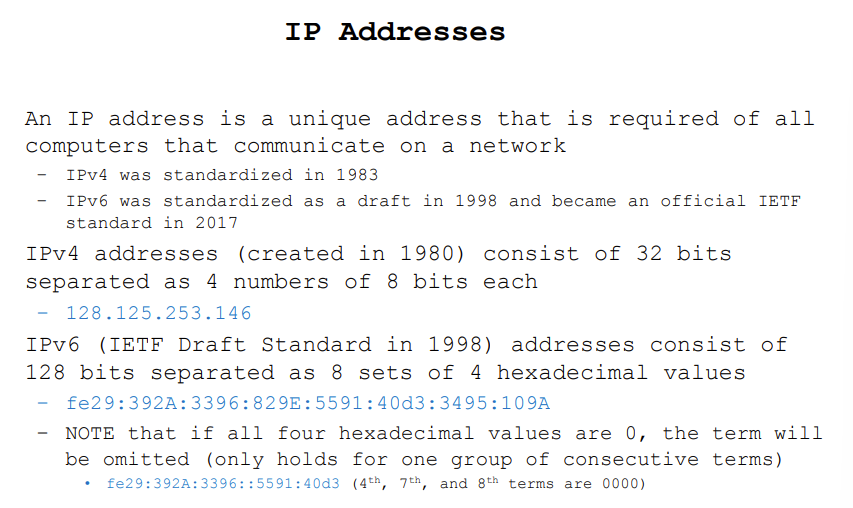
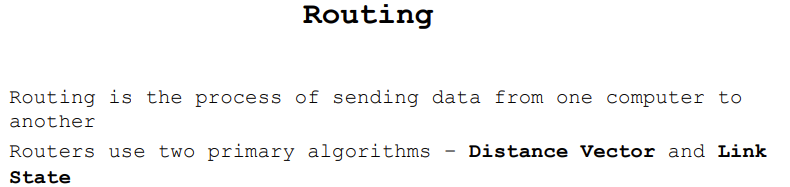
MIME headers at the front of the file define the type of data the message includes, e.g., here are a set of MIME types describing an attachment at an ftp site

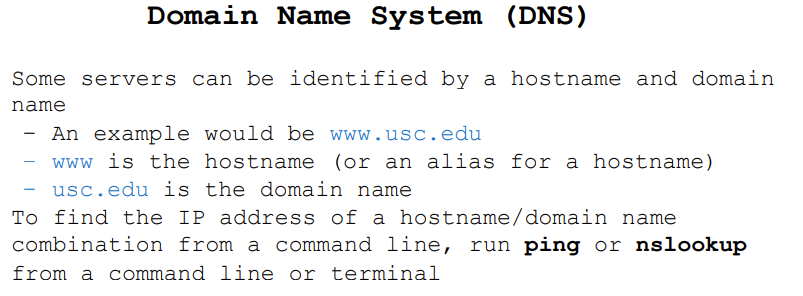


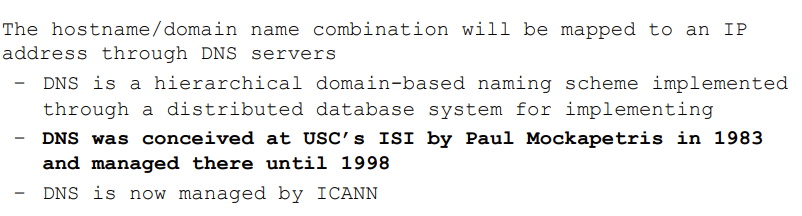


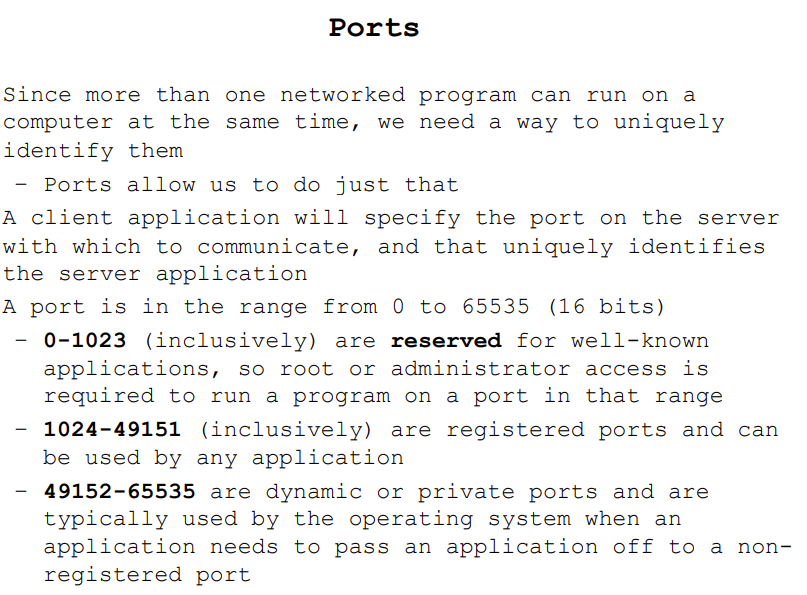


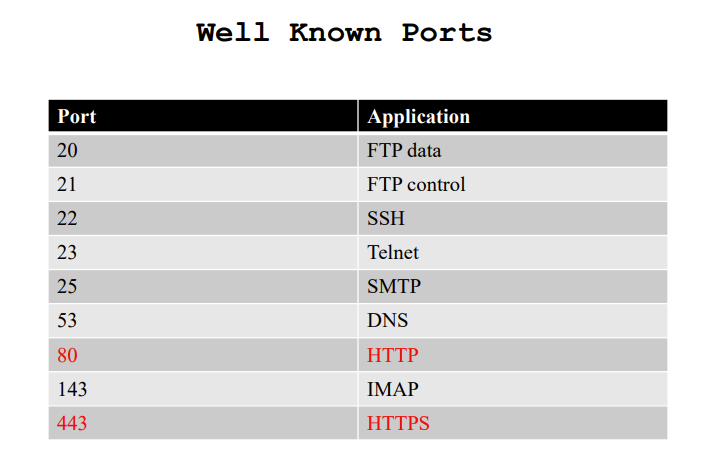
     

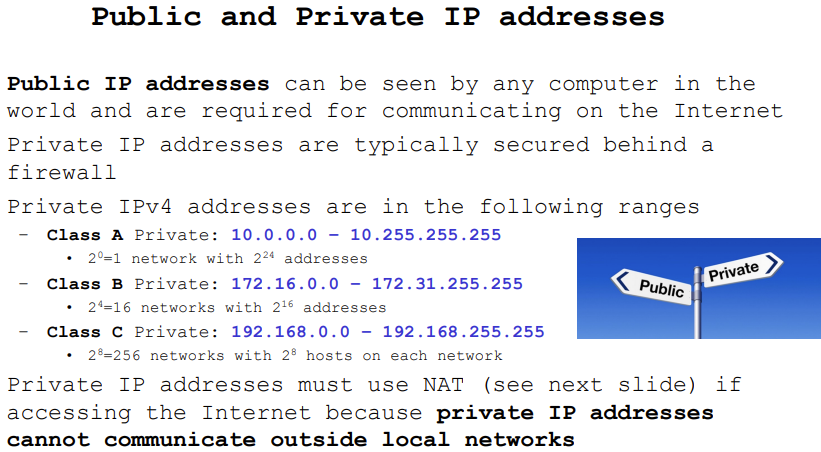
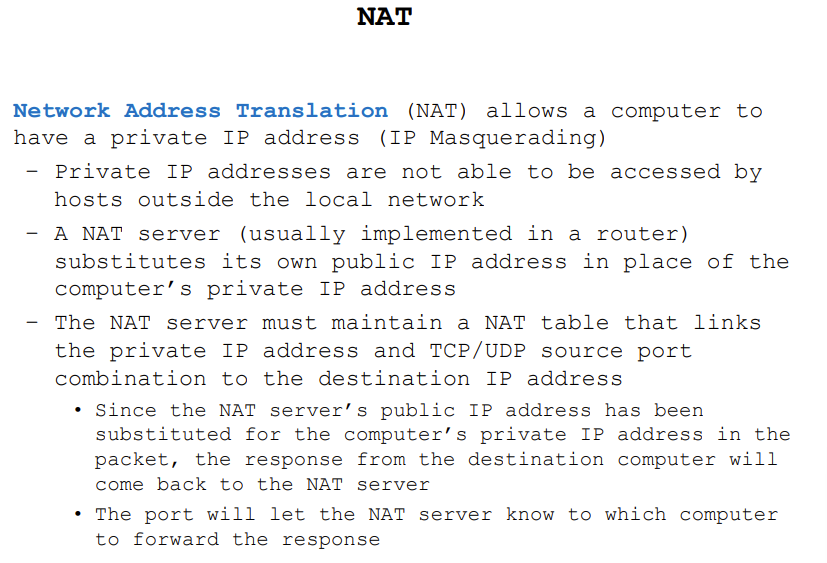
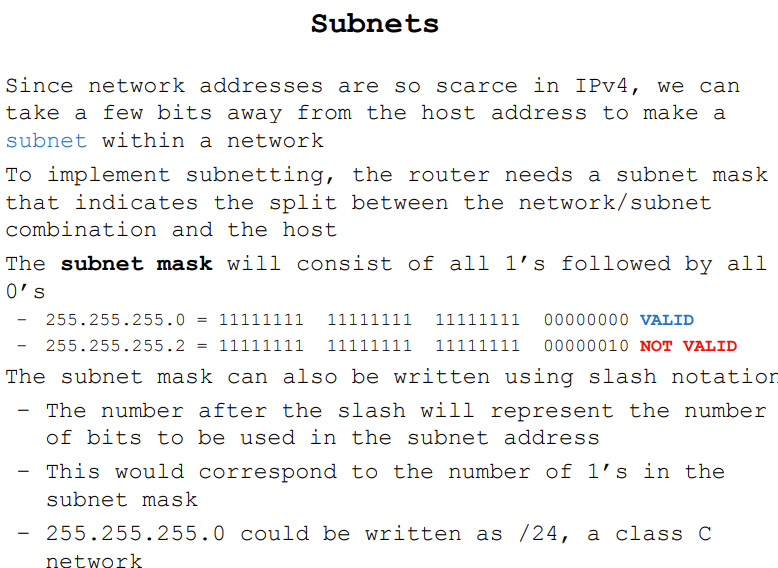
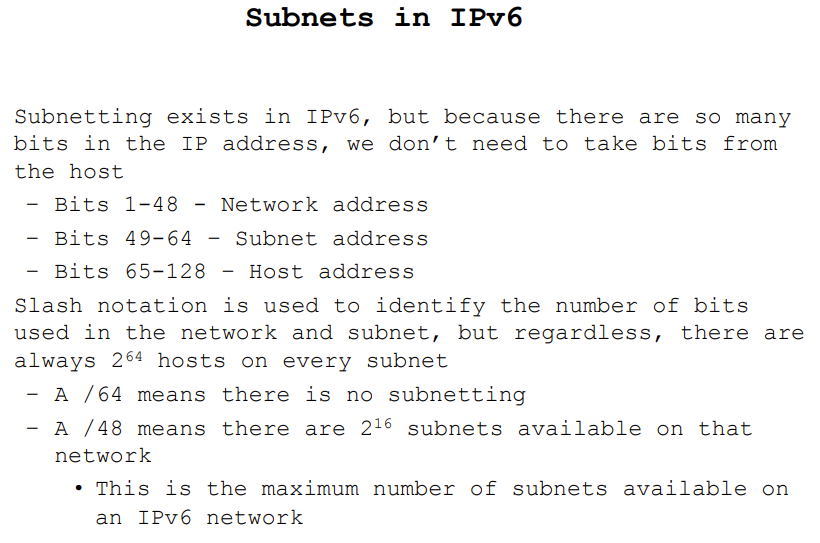
 

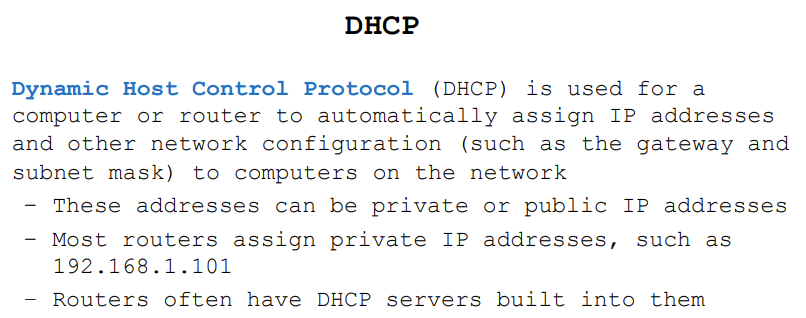
****

****

****

****

**** ****  

****