VIT®

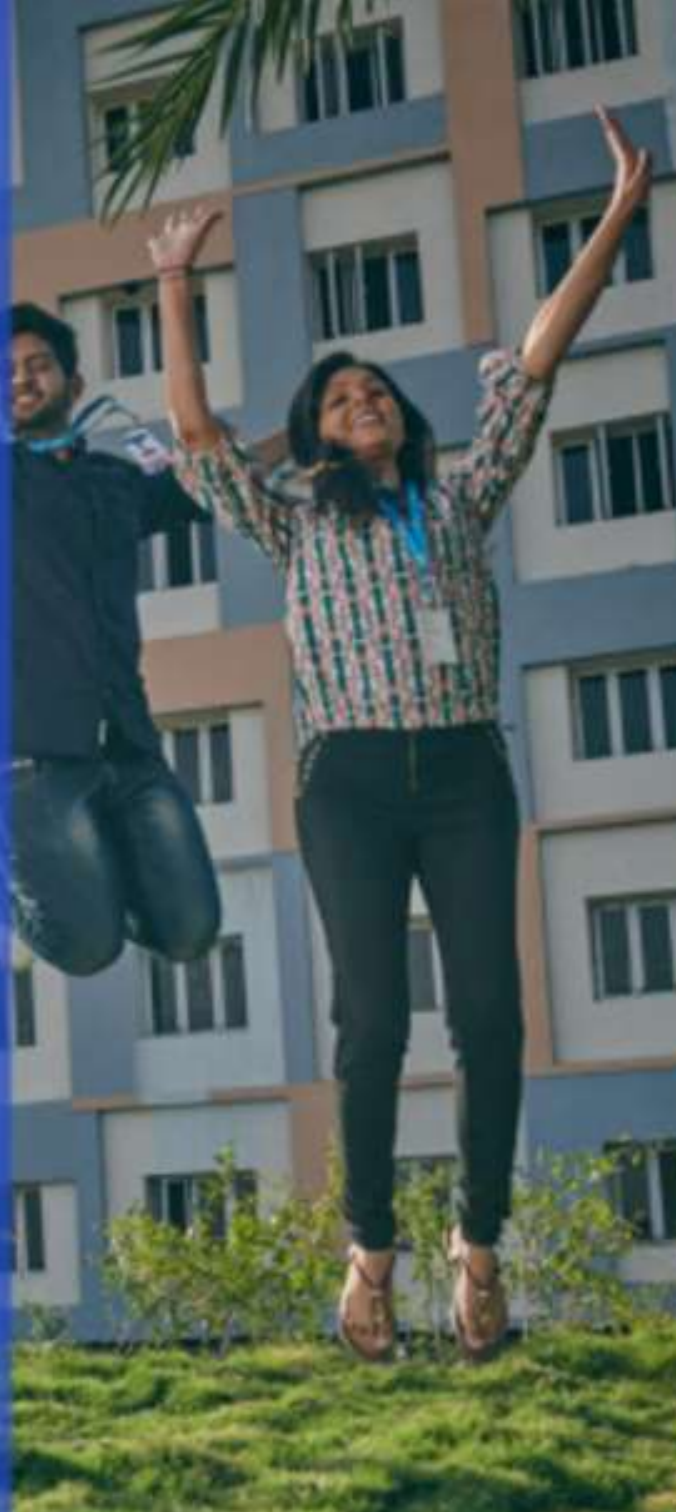
BHOPAL

www.vitbhopal.ac.in

VIT BHOPAL

UNIVERSITY

A PLACE TO LEARN; A CHANCE TO GROW



welcome to

CSE 4001 - Internet and Web Programming

Unit 1

Introduction to Web System

Internet Overview- WWW - Web Protocols Web Browsers and Web Servers - Web System Architecture – URL - Domain Name – Client and Server-side Scripting.

Text Books:

- 1. Thomas Powell, HTML and CSS, Complete Reference, Fifth Edition, Mc Graw Hill, 2010
- 2. Thomas Powell, Fritz Schneider , JavaScript The complete reference, Mc Graw Hill, 2013
- 3. Tom Christiansen, Nathan Torkington, Perl Cookbook, O'Reilly, 2012
- 4. David Powers, PHP Solutions, Dynamic web page design made easy, Apress, 2010
- 5. Joe Fawcett, Danny Ayers, Liam R. E. Quin, Beginning XML, 5th Edition, Wrox, 2012

Reference Books:

- 1. Paul Dietel, Harvey Dietel and Abbey Dietel, Internet and World Wide Web How to program, 5th International Edition, Pearson, 2012

- Who founded WWW consortium?
- What is mean by web?
- Name some web protocol?
- What is mean by URI?
- HTTP?

World Wide Web

- In October 1994, Tim Berners-Lee founded an organization—called the World Wide Web Consortium (W3C)—devoted to developing nonproprietary, interoperable technologies for the World Wide Web.
- One of theW3C's primary goals is to make the web universally accessible—regardless of ability, language or culture.

- The W3C home page (www.w3.org) provides extensive resources on Internet and web technologies.
- The W3C is also a standardization organization. Web technologies standardized by the W3C are called **Recommendations**.

W3C Recommendations include the

- ✓ Extensible HyperText Markup Language (XHTML),
- ✓ Cascading Style Sheets (CSS),
- ✓ HyperText Markup Language (HTML—now considered a "legacy" technology) and the
- ✓ Extensible Markup Language (XML).

- A recommendation is not an actual software product, but a document that specifies a technology's role, syntax rules and so forth.

- Distinctive feature of Web: support for hypertext (text containing links)
  - Communication via Hypertext Transport Protocol (HTTP)
  - Document representation using Hypertext Markup Language (HTML)
- The Web is the collection of machines (Web servers) on the Internet that provide information, particularly HTML documents, via HTTP.
- Machines that access information on the Web are known as Web clients. A Web browser is software used by an end user to access the Web.

Web Protocols

Hypertext Transport Protocol (HTTP)

- HTTP is based on the request-response communication model:
  - Client sends a request
  - Server sends a response
- HTTP is a stateless protocol:
  - The protocol does not require the server to remember anything about the client between requests.

- Normally implemented over a TCP connection (80 is standard port number for HTTP)
- Typical browser-server interaction:
  - User enters Web address in browser
  - Browser uses DNS to locate IP address
  - Browser opens TCP connection to server
  - Browser sends HTTP request over connection
  - Server sends HTTP response to browser over connection
  - Browser displays body of response in the client area of the browser window

- The information transmitted using HTTP is often entirely text
- Can use the Internet's Telnet protocol to simulate browser request and view server response
- Telnet is a client-server protocol, based on a reliable connection-oriented transport. used on the Internet or local area network to provide a bidirectional interactive text-oriented communication facility using a virtual terminal connection.

HTTP

Connect

```
$ telnet www.example.org 80
Trying 192.0.34.166...
Connected to www.example.com
(192.0.34.166).
Escape character is '^]'.
```

Send Request

```
GET / HTTP/1.1
Host: www.example.org
```

Receive Response

```
HTTP/1.1 200 OK
Date: Mon, 24 Jan 2022
20:30:49 GMT
...
```

HTTP Request

- Structure of the request:
  - start line
  - header field(s)
  - blank line
  - optional body

Start line

- Example: `GET / HTTP/1.1`

- Three space-separated parts:
  - HTTP request method
  - Request-URI (Uniform Resource Identifier)
  - HTTP version

HTTP version

- We will cover 1.1, in which version part of start line must be exactly as shown

HTTP Request

- Uniform Resource Identifier (URI)
  - Syntax: *scheme : scheme-depend-part*
    - Ex: In `http://www.example.com/`

      the scheme is `http`
  - Request-URI is the portion of the requested URI that follows the host name (which is supplied by the required Host header field)
    - Ex: / is Request-URI portion of

      `http://www.example.com/`

URI

- URI's are of two types:
  - Uniform Resource Name (URN)
    - Can be used to identify resources with unique names, such as books (which have unique ISBN's)
    - Scheme is `urn`
  - Uniform Resource Locator (URL)
    - Specifies location at which a resource can be found
    - In addition to `http`, some other URL schemes are `https`, `ftp`, `mailto`, and `file`

HTTP Request

- Start line
  - Example: `GET / HTTP/1.1`
- Three space-separated parts:
  - **HTTP request method**
  - Request-URI
  - HTTP version

HTTP Request

- Common request methods:
  - **GET**
    - Used if link is clicked or address typed in browser
    - No body in request with GET method
  - **POST**
    - Used when submit button is clicked on a form
    - Form information contained in body of request
  - **HEAD**
    - Requests that only header fields (no body) be returned in the response

HTTP Request

- Structure of the request:
  - start line
  - **header field(s)**
  - blank line
  - optional body

HTTP Request

- Header field structure:
  - *field name* : *field value*
- Syntax
  - Field name is not case sensitive
  - Field value may continue on multiple lines by starting continuation lines with white space
  - Field values may contain MIME types, quality values, and wildcard characters (*'s)

Multipurpose Internet Mail Extensions (MIME)

- Convention for specifying content type of a message
  - In HTTP, typically used to specify content type of the body of the response
- MIME content type syntax:
  - *top-level type / subtype*
- Examples: text/html, image/jpeg

HTTP Quality Values and Wildcards

- Example header field with quality values:

```
accept:
   text/xml,text/html;q=0.9,
   text/plain;q=0.8, image/jpeg,
   image/gif;q=0.2,*/*;q=0.1
```

- Quality value applies to all preceding items
- Higher the value, higher the preference
- Note use of wildcards to specify quality 0.1 for any MIME type not specified earlier

HTTP Request

- Common header fields:
  - Host: host name from URL (required)
  - User-Agent: type of browser sending request
  - Accept: MIME types of acceptable documents
  - Connection: value `close` tells server to close connection after single request/response
  - Content-Type: MIME type of (POST) body, normally application/x-www-form-urlencoded
  - Content-Length: bytes in body
  - Referer: URL of document containing link that supplied URI for this HTTP request

HTTP Response

- Structure of the response:
  - status line
  - header field(s)
  - blank line
  - optional body

HTTP Response

- Status line
  - Example: HTTP/1.1 200 OK
- Three space-separated parts:
  - HTTP version
  - status code
  - reason phrase (intended for human use)

HTTP Response

- Status code
  - Three-digit number
  - First digit is class of the status code:
    - 1=Informational
    - 2=Success
    - 3=Redirection (alternate URL is supplied)
    - 4=Client Error
    - 5=Server Error
  - Other two digits provide additional information
  - See http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

HTTP Response

- Structure of the response:
  - status line
  - **header field(s)**
  - blank line
  - optional body

HTTP Response

- Common header fields:
  - Connection, Content-Type, Content-Length
  - Date: date and time at which response was generated (required)
  - Location: alternate URI if status is redirection
  - Last-Modified: date and time the requested resource was last modified on the server
  - Expires: date and time after which the client's copy of the resource will be out-of-date
  - ETag: a unique identifier for this version of the requested resource (changes if resource changes)

Jokes

"Hello Mr Programmer", the donkey said, "how are you?".

"mighty fine, thank you donkey", the HTML dev replied.

Immediately the donkey started crying.

"What's the matter little friend?" the HTML dev asked.

"I called you a programmer, at least you could call me horse" the donkey bawled.

Jokes

- Guy 1 - 'How can you tell the difference between HTML and HTML5?'

  Guy 2 - 'Open it in Internet Explorer'

  Guy 1 - 'Ok'

  Guy 2 - 'Did it work?'

  Guy 1 - 'No'

  Guy 2 - 'It's HTML5'

Thank You