



# HTML-CSS-JS: The Language of Web Pages

Anoop M. Namboodiri  
Biometrics and Secure ID Lab  
IIIT Hyderabad





# Markup Languages

- ASCII for text
  - A: 65, B: 66, .. Z: 90, [: 91, .. a: 97, ... , z: 122.
- Unicode text
  - α: U+03B1, अ: U+0905, अ: U+0985, अ: U+0B85, अ: U+0C05
- How to represent information beyond characters?
  - Font, Style (bold, italics), Color, Semantics, Layout
- Solution: Markup Language
  - To represent a style such as bold, use tags
    - `<b> Hyderabad </b>`
    - `<li> Hindi </li>`





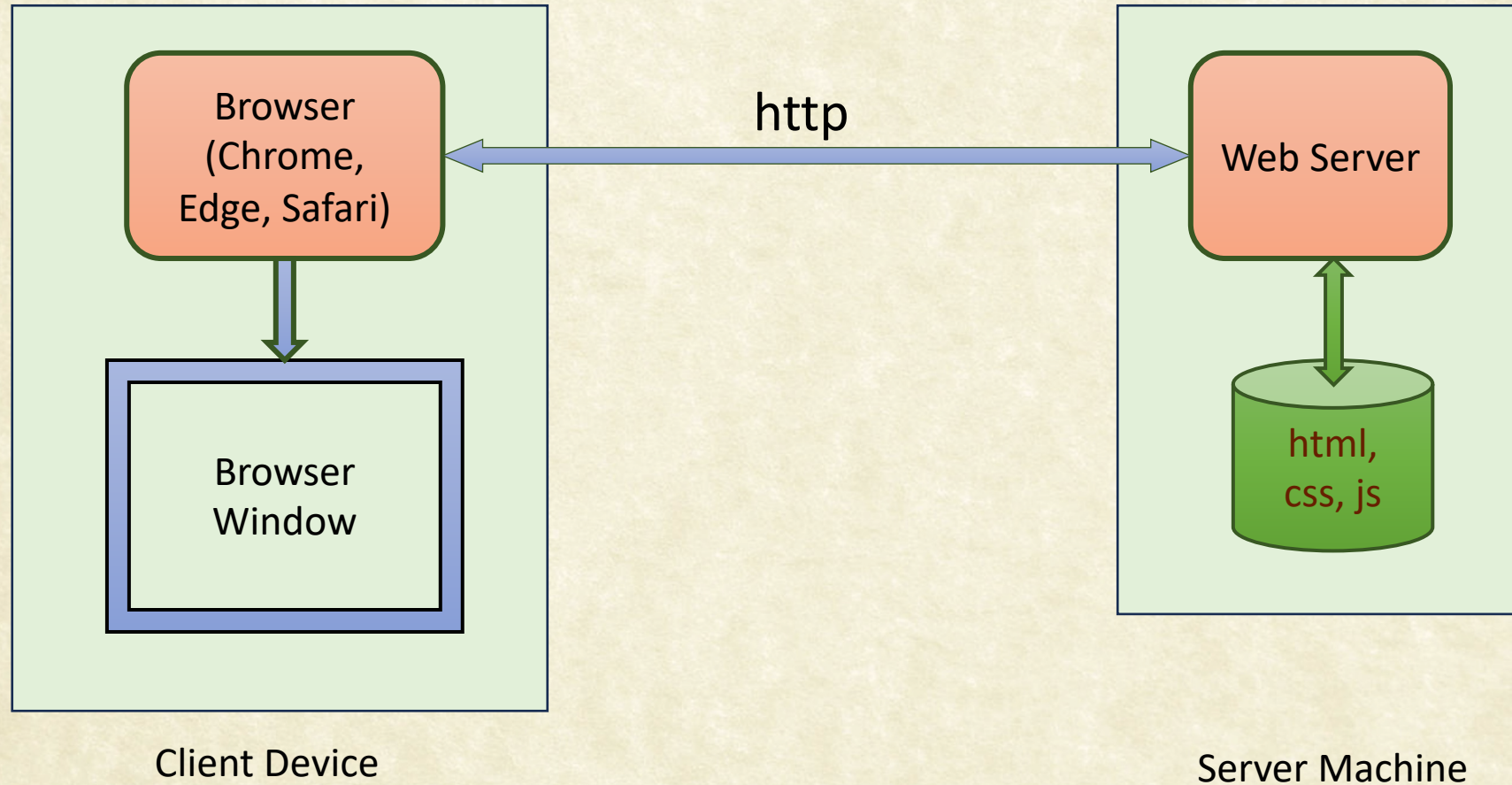
# Examples of Markup Languages

- LaTeX
- HTML: HyperText Markup Language
- XML: eXtensible Markup Language
- docx: Microsoft XML document format
- Markdown





# Accessing a Webpage







# HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Webpage </title>
  </head>
  <body>
    <h1> Welcome to ABC</h1>
    <p> Text goes here </p>
  </body>
</html>
```

- Lists
  - <ul> or <ol> and <li>
- Links
  - <a href="url">Link Text</a>
- Images
  - 
- Tables
  - <table> ... </table>
  - <tr> row </tr> ...
  - <td> data </td> ...
- Container: <div> </div>
- Forms





# CSS: Adding Style to WebPages

- HTML: Structure and Content; CSS: Style

- Inline Style

```
<p style="color:red; background:cyan">Text.</p>
```

- Internal Style: in header

```
<style>
  .classname{
    color:red;           # property:value
    background:cyan;
  }
</style>
```

```
<div class="classname"> Content </div>
```

- External Style: CSS

```
<link rel="stylesheet" href="./colorful.css">
```

- CSS Properties

- color:red;
- background:blue;
- font-size:100px;
- font-style:italic;
- width:640px;
- height:480px;
- margin:20px;
- padding: 10px;
- border:3px;
- float: left;

- More at: [w3schools.com](http://w3schools.com)





# Javascript: Adding Intelligence

- `<script src="location"></script>`

```
var count = 10;
```

```
var name = "IIIT";
```

```
var cols=['Red', 'Green','Blue'];
```

```
for(var i=0; i<cols.length; i++){  
    console.log(cols[i]);  
}
```

- Functions
- Event Listeners





Questions?





# Programming Python

“Python is an experiment in how much freedom programmers need. Too much freedom and nobody can read another's code; too little and expressiveness is endangered.”



- Guido van Rossum  
Benevolent Dictator for Life





# Why Python?

- Scripting Language
- Versatile
- Popular
- Simplicity
- Modules and Frameworks
- Data Visualization, Machine Learning, Cyber Security, Web Servers,
- Extensive Online Documentation
- Community Support





# Brief History of Python

- Invented in early 90s by Guido van Rossum
- Named after Monty Python (not the snake)
- Open sourced from the beginning
- A scripting language, but is much more
- Scalable, object oriented and functional from the beginning
- Used by Google from the beginning
- Increasingly popular





https://docs.python.org/3/

The screenshot shows a web browser window with the address bar set to `docs.python.org`. The page title is "Python 3.12.1 documentation". Below the title, a welcome message states: "Welcome! This is the official documentation for Python 3.12.1." The page is organized into two columns of links. The left column includes: "What's new in Python 3.12?" (with a subtitle "or all 'What's new' documents since 2.0"), "Tutorial" (with a subtitle "start here"), "Library Reference" (with a subtitle "keep this under your pillow"), "Language Reference" (with a subtitle "describes syntax and language elements"), "Python Setup and Usage" (with a subtitle "how to use Python on different platforms"), and "Python HOWTOs" (with a subtitle "in-depth documents on specific topics"). The right column includes: "Installing Python Modules" (with a subtitle "installing from the Python Package Index & other sources"), "Distributing Python Modules" (with a subtitle "publishing modules for installation by others"), "Extending and Embedding" (with a subtitle "tutorial for C/C++ programmers"), "Python/C API" (with a subtitle "reference for C/C++ programmers"), and "FAQs" (with a subtitle "frequently asked questions (with answers!)"). At the bottom left, there is a section header "Indices and tables:".

Python 3.12.1 documentation

Welcome! This is the official documentation for Python 3.12.1.

**Parts of the documentation:**

- [What's new in Python 3.12?](#)  
*or all "What's new" documents since 2.0*
- [Tutorial](#)  
*start here*
- [Library Reference](#)  
*keep this under your pillow*
- [Language Reference](#)  
*describes syntax and language elements*
- [Python Setup and Usage](#)  
*how to use Python on different platforms*
- [Python HOWTOs](#)  
*in-depth documents on specific topics*
- [Installing Python Modules](#)  
*installing from the Python Package Index & other sources*
- [Distributing Python Modules](#)  
*publishing modules for installation by others*
- [Extending and Embedding](#)  
*tutorial for C/C++ programmers*
- [Python/C API](#)  
*reference for C/C++ programmers*
- [FAQs](#)  
*frequently asked questions (with answers!)*

**Indices and tables:**





<https://docs.python.org/3/tutorial/>

The screenshot shows a web browser window displaying the Python 3.12.1 documentation. The address bar shows `docs.python.org`. The page title is "The Python Tutorial". The left sidebar contains navigation links: "Previous topic Changelog", "Next topic 1. Whetting Your Appetite", and "This Page Report a Bug Show Source". The main content area has the heading "The Python Tutorial" and three paragraphs of text. The first paragraph describes Python as an easy-to-learn, powerful programming language. The second paragraph mentions the Python interpreter and standard library are freely available. The third paragraph states the Python interpreter is easily extended with new functions and data types.

Python » English 3.12.1 3.12.1 Documentation » The Python Tutorial previous | next | modules | index

Theme Auto | Quick search Go |

## The Python Tutorial

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.

The Python interpreter and the extensive standard library are freely available in source or binary form for all major platforms from the Python web site, <https://www.python.org/>, and may be freely distributed. The same site also contains distributions of and pointers to many free third party Python modules, programs and tools, and additional documentation.

The Python interpreter is easily extended with new functions and data types implemented in C or C++ (or other languages callable from C). Python is also suitable as an extension language for customizable applications.

This tutorial introduces the reader informally to the basic concepts and features of the Python language and system. It helps to have a Python interpreter handy for hands-on experience, but all examples are self-contained, so the tutorial can be read off-line as well.





# Advantages of Python

- Easy to Learn and write error-free code
  - You can start coding today
- Encourages and Insists Coherence (clean/readable code)
  - Limited ways to do a specific task (unlike perl)
  - Forces you to indent
  - Language avoid unnecessary steps (declaration, semicolons)
- Powerful (batteries included)
  - Provides an ever-growing number of powerful modules
- Provides flexibility
  - You can extend the language with additional modules
  - Can make hybrid systems
- Provides Speed
  - Can compile to portable byte code
- Widely used / growing fast





# Uses of Python

- Shell tools
  - System admin tools, Command line programs
- Text processing
- Rapid prototyping and development
- Integration of modules from different languages
- Graphical user interfaces
- Database access
- Distributed programming
- Internet scripting





# What Gives?

- Slower than C; like any other scripting language
  - Although efficient built-in algorithms and Data structures might offset this
- Delayed error notification
- Lack of profiling tools





# Installing Python

- Pre-installed on Unix systems (Linux, Mac OSX).
- Binaries available for Windows
- Latest stable versions are 3.11.7 and 3.12.1
  - We will stick with 3.10 as it works with 3.11 and 3.12
- Several editors and IDEs
  - VIM / Emacs
  - IDLE
  - PyCharm
  - VS Code