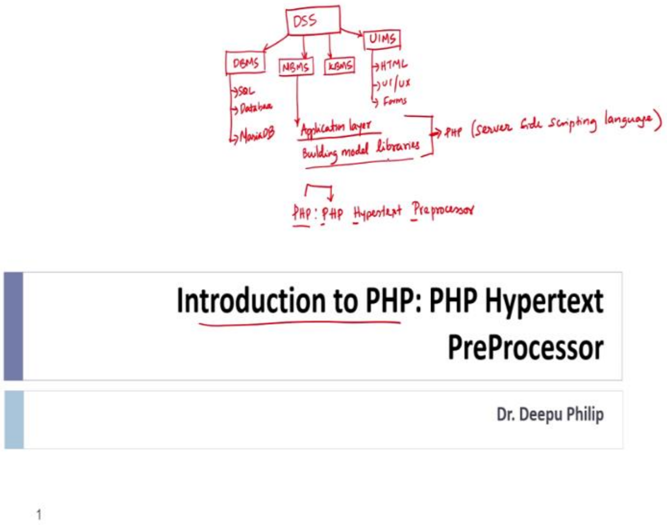
Computer Aided Decision Systems Industrial Practices using Big Analytics
Professor Deepu Philip
Department of Industrial and Management Engineering
Professor Amandeep Singh
Imagineering Laboratory
Indian Institute of Technology, Kanpur
Lecture No. 45
Introduction to PHP

Good afternoon, everyone. Welcome to yet another lecture of the MOOC's course titled Web-based Decision Support System for engineers and practitioners. In this course as you have already gone through it, we are focusing more towards how the Decision Support system can be applied for various business problems, especially, from a practitioner standpoint and how do we make decisions on the or about the unstructured decision-making problems associated with business. And, we have so far just a recap of what the modules I have been covering.

(Refer Slide Time: 00:53)



Introduction to PHP: PHP Hypertext PreProcessor

Dr. Deepu Philip

1

We have seen that if you look into the screen, that we have a DSS or the Web-based DSS as the central component in which we told we were having 4 components is part of it and the first component was what we call as the DBMS (the Database Management System). Then, we talked about what we call MBMS (the Model-based Management System). Then, we talked about the KBMS (Knowledge-based Management System) and the last one we discussed was the UIMS (User Interface Management System).

And, the DBMS we have already seen things like SQL, we have seen the Database and we have seen Normalization and we have also seen what Maria DB is, etcetera, and how do we create database etcetera as part of this.

And, the User Interface Management System, we have already seen aspects of HTML, the UI, UX, what it is and how we design at UI and that kind of a thing so, forms, etcetera, we have already seen that.

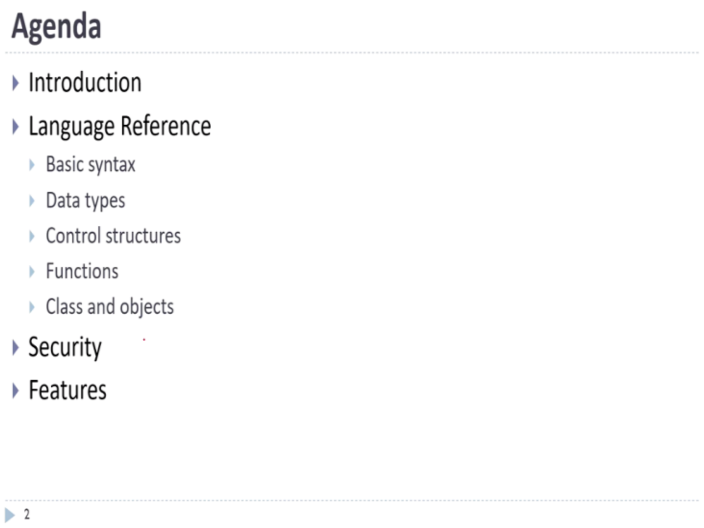
And, we have mentioned that at some point of time the Model-based Management System will compromise an application layer and build model libraries.

So, there are multiple ways to do this. But, for the purpose of this course we are going to do mostly here as a part of a PHP or it is server-side scripting language, that is what we are going to study in the part of it. And, in the Knowledge-based Management System, we will study towards the end of it.

And, as mentioned earlier, this lecture is all about the Introduction to PHP and the name PHP, it is a very interesting name because the name again the first P stands for PHP again. And then, the next is Hypertext and the Preprocessor. So, this is the PHP expansion.

So, PHP appears again in the expansion of acronyms. It is a very interesting way to talk about all these kinds of things in this lecture. So, we are discussing specifically the PHP, how PHP is used as an application program or how it actually builds an application layer and how it also builds the model, how do you create the libraries of different decision models.

(Refer Slide Time: 03:44)



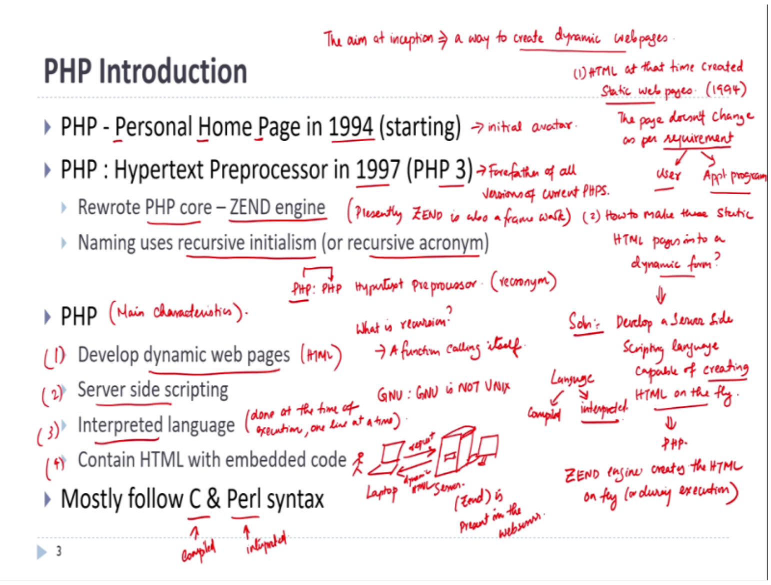
Agenda

- Introduction
- Language Reference
  - Basic syntax
  - Data types
  - Control structures
  - Functions
  - Class and objects
- Security
- Features

2

So, today we have a brief introduction then, references of basic language and syntax, data types etcetera, etcetera, security and some features is what we will discuss today.

(Refer Slide Time: 03:54)



The aim at inception ⇒ a way to create dynamic webpages.

PHP Introduction

(1) HTML at that time created Static web pages. (1994)

The page doesn't change as per requirement → User, Appl program

- PHP - Personal Home Page in 1994 (starting) → initial avatar.
- PHP : Hypertext Preprocessor in 1997 (PHP 3) → Forefather of all versions of current PHPS.
  - Rewrote PHP core – ZEND engine (Presently ZEND is also a frame work)
  - Naming uses recursive initialism (or recursive acronym)

PHP: PHP Hypertext Preprocessor. (recronym)

(2) How to make these Static HTML pages into a dynamic form?

⇓

Soln: Develop a Server Side Scripting language capable of creating HTML on the fly.

⇓

PHP.

- PHP (Main characteristics).

What is recursion? → A function calling itself.

(1) Develop dynamic web pages (HTML)

(2) Server side scripting

GNU : GNU is NOT UNIX

Language → Compiled, interpreted

(3) Interpreted language (done at the time of execution, one line at a time).

(4) Contain HTML with embedded code

(Zend) is Present in the webserver.

ZEND engine creates the HTML on fly (or during execution)

- Mostly follow C & Perl syntax

↑ compiled ↑ interpreted

3

- ➢ So, originally if you think about it the PHP started as the Personal Home Page 1994, this is the beginning, this is the initial avatar of PHP. And, the aim of this was at inception, a way to create dynamic web pages. This is the main interest of the time, because remember a few things:

1) HTML at that time created static web pages. So, once you create a web page, it is static, it means the page does not change as per requirement, whose requirement? The requirement can be of two types: it is either the user requirement or the application program requirement. Either of the requirements, the HTML page did not change, it remained the same.
2) So, then, the question was how to make these static HTML pages into a dynamic form, this is the fundamental question. From there, what actually created was that, develop server-side scripting language capable of creating HTML on the fly. So, this was a solution that was taught through. So then, that gave rise to PHP originally, that is how PHP originated. So, we think about it that we wanted to bring the dynamic web pages or we wanted to create dynamic web pages, but at that time or in 1994 and all, HTML resulted in what we call a static web page; the web page content did not change, once you create the webpage it remained.

But the web page there were certain times or a requirement is at the requirement by the user or by the application program to make the web page content change or dynamic. So then, the question arose was how to make these static HTML pages into a dynamic form or a dynamic page and the solution developed was creator servers as scripting language that is capable of creating HTML on the fly or as the pages executed, it should create the HTML. And, that is how the PHP came into picture. So, 1994 started as a personal home page if you have seen the web page.

- Then, in 1997 the PHP 3, which is the forefather of all versions of PHP, we see now, all versions of current PHP. There is PHP 4, PHP 5 whatever it is you talk about, it is all the PHP, so, that became what you call as Hypertext Preprocessor.

i) So, the core of PHP, which is called the ZEND engine. Presently, it is also a framework, ZEND is also a framework, but the ZEND engine is the one that actually creates, ZEND engine creates the HTML on fly or during execution. So, that is what we are talking about.

ii) So, the entire PHP code, the ZEND engine we wrote in 1997 and this naming of PHP as I mentioned earlier, then, the PHP appearing again, then, Hypertext Preprocessor. This type of naming is known as recronym or what you call as recursive acronym. What is recursion? Recursion is a function calling itself by this recursion. So, what is a recursive acronym? So, a recronym is an acronym that calls in itself. So, the PHP acronym calls it again. So, that is why it is called recursive acronym or recronym or sometimes it is also known as recursive initialism. So, there are a couple of other examples also. So, GNU is an example of it, GNU is not UNIX, so, this is another example of it. So, that kind of thing is there. So, PHP is an example of this, so, we can see quite a lot of the things that came out of the MIT lab actually followed this.

- So, what does PHP do? So, the main characteristics: what are the main things?

i) The first thing is, it develops dynamic web pages. It builds the web pages that are on the fly, dynamic web pages or HTML. When we talk about web pages we are specifically talking about HTML.

ii) Second thing is, it is the server-side scripting. What actually happens is, if you think about it, what is the server-side scripting, let us say it is a laptop and you have a client sitting on it, you are working on the laptop and then, there is a connection. So, somewhere here, there is a big server, computer, a big server connected to this. So, this is server and the server-side scripting means the ZEND engine of PHP, the ZEND engine is present in the web server.

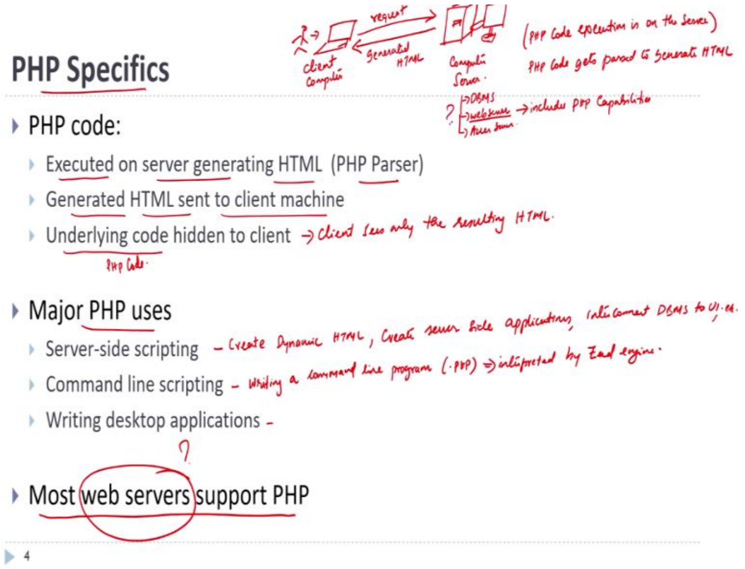
So, the Web Server which is the server-side application, ZEND engine is present there. So, when the user accesses that particular web page, the Web Server passes it or uses the ZEND engine to dynamically create the HTML and sends it back. So, the user request comes this way. So, this is the request comes in and the dynamic HTML goes back that is what it is.

iii) Now, the third one is that it is an interpreted language. So, there are two types of language you may remember in your earlier stuff. One is compiled and another one is called an interpreter. So, PHP is an interpreted language, this means, it is interpreted one line at a time. So, it is like done at the time of execution. One line at a time, there is no compilation, absolutely no compilation going on.

iv) And then, the fourth point is that it contains HTML or it can embed HTML in your code that is the main part of it.

- ➢ It mostly follows C and Perl syntax. C and Perl are two languages, but C is a compiled language and Perl is an interpreted language. So, it follows a syntax of both of these, but in reality, it is an interpreted language. So, I hope you guys have a reasonably okay idea about this. So, now let us move forward.

(Refer Slide Time: 12:42)


request
Generated HTML
client Computer
Computer Server.
(PHP Code execution is on the Server)
PHP Code gets parsed to generate HTML
? →DBMS
→web server →includes PHP Capabilities
→Access Server.

PHP Specifics

- PHP code:
  - Executed on server generating HTML (PHP Parser)
  - Generated HTML sent to client machine
  - Underlying code hidden to client → Client sees only the resulting HTML.

PHP Code.

- Major PHP uses
  - Server-side scripting – Create Dynamic HTML, Create server side applications, interconnect DBMS to UI, etc.
  - Command line scripting – Writing a command line program (.PHP) ⇒ interpreted by Zend engine.
  - Writing desktop applications –

?

- Most web servers support PHP

4

So, what are some of the specifics of PHP, what are some of the specialties associated with PHP? So, as I said earlier you have a client, laptop and then, there it is connected to a server. The server has a lot of things residing on it, it may have a DBMS, it has a web server, it has an access server, etcetera, many aspects it will have.

So, among the web servers, it will also include PHP capabilities which are realized by the ZEND engine. So, what happens is when the user sends a request.

i) Then, the PHP code gets executed on the server. So, PHP code is executed on the server. And, what happens is as part of the execution it generates HTML. So, that process is called parsing. So, the PHP code gets parsed to generate HTML.

ii) So then, this HTML is sent to the client machine, the generated HTML at the server side is sent to the client machine. So, the request from the client goes to the server and the generated HTML is sent back to the client machine and what happens is that the client gets to see them.

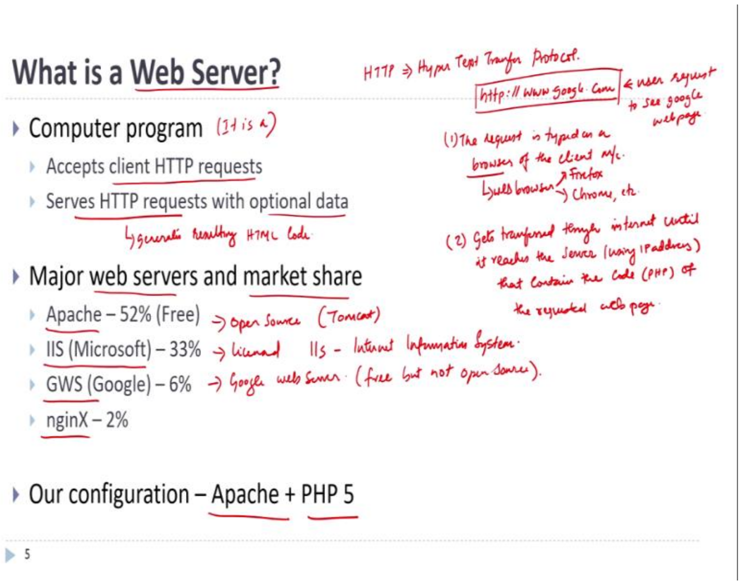
iii) So, the client sees only the resulting HTML, the underlying PHP code, that PHP code is hidden to the client the client never gets to see the PHP code. So, the logic of how the web page is created is hidden from the client.

➢ So, what are major purposes or major uses of PHP?

i) It is used in server-side scripting. So, specifically create dynamic HTML pages, create server-side applications, interconnect DBMS to UI, so, those kinds of things, etcetera, all these are part of the server-side aspects of PHP.

ii) You can also do command line scripting which means you can write a command line program. So, it is like writing a command line program. It is a .PHP program and can be executed using the PHP, that is also doable this will be an interpreted program by ZEND engine. And then, writing the desktop application, we can also write standalone desktop applications that are also a third part of the PHP, but the important thing is, most of the web servers that are available in the world now, they all support PHP. So, what is a web server? So, this question is obviously web server, this keyword, we talked about this Web Server here also, what is it, that is our next question.

(Refer Slide Time: 16:58)



What is a Web Server?

HTTP ⇒ Hyper Text Transfer Protocol.

http://www.google.com ← user request to see google webpage

- Computer program (It is a)
  - Accepts client HTTP requests
  - Serves HTTP requests with optional data
    - ↳ generates resulting HTML code

(1) The request is typed on a browser of the client m/c.
↳ web browser → Firefox → Chrome, etc.

(2) Gets transferred through internet until it reaches the server (using IP address) that contain the code (PHP) of the requested web page.

- Major web servers and market share
  - Apache – 52% (Free) → open Source (Tomcat)
  - IIS (Microsoft) – 33% → licensed IIS - Internet Information System.
  - GWS (Google) – 6% → Google web server. (free but not open source).
  - nginX – 2%
- Our configuration – Apache + PHP 5

5

So, what is a web server?

- ➢ The simplest answer to it is a computer program. Web Server is a computer program and what does it do? The two things;

i) It accepts HTTP requests; HTTP stands for Hypertext Transfer Protocol. So, if you see a web address, it will say https://www.google.com/ this is the request. This means this is the user request to see Google web page. So, you type it on this, so where does this get created?

1) So, the request is typed on a browser of the client machine, it could be an internet explorer. So, what is a Web Browser? It can be Firefox, it can be this one which is called Chrome, etcetera, they are all browsers. So, this request is created in the browser

2) Gets transferred through the internet until it reaches the server. How does it reach? Using an IP address that contains the PHP code of the requested web page, that is what happens. Then, what happens?

ii) So, once this is done, the HTTP request, it serves, the HTTP request, it says, show me the Google page and what it does? It generates resulting HTML code and if there is optional data, there is another data nothing to be fetched, then, the system will fetch that data and send it back.

- ➢ So, as of now, the major web servers, they are present in the world and their market share as follows:

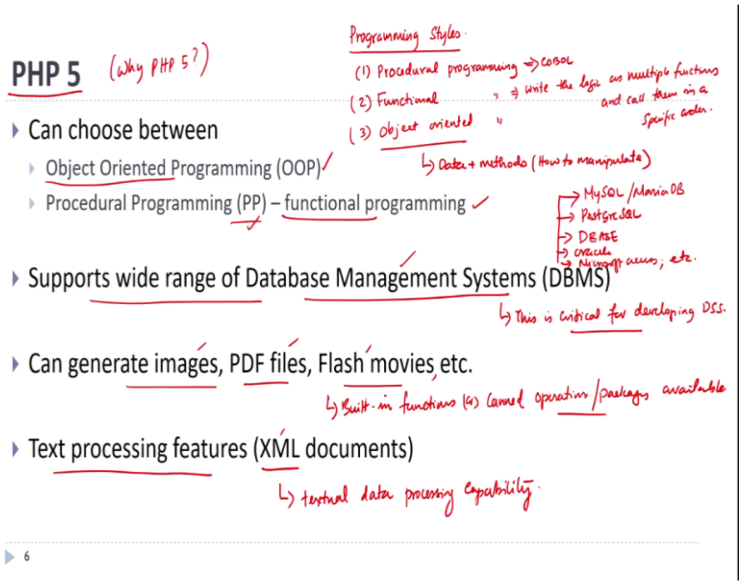
i) Apache, this is the open source, the free software foundation which is mostly 52, now, it is 55 percent or something, some people also call it the Tomcat as well, so, it is another word, actually it is a simpler version of Apache Tomcat either way. So, the majority is with them.

ii) Second one is, IIS which is proprietary. This is not open source; this is licensed software. Microsoft creates it, it is an Internet Information System. IIS stands for Internet Information System. It is the second most popular but the popularity is now somewhere about 30 percent or something like that.

iii) And then, there is GWS (Google web server), it is about 6 percent to 8 percent now.

iv) And then, there is another nginX Google Web Server. This is also free, but not open source. And then, nginX is a minor, 2 percent. So, in our discussion, mostly what we do in this class is, we are focusing on Apache plus PHP 5. We will consider it as the standard or we assume that this configuration is available for us and then, we build the application accordingly.

(Refer Slide Time: 21:32)



PHP 5 (why PHP 5?)

- Can choose between
  - Object Oriented Programming (OOP)✓
  - Procedural Programming (PP) – functional programming ✓
- Supports wide range of Database Management Systems (DBMS)
- Can generate images, PDF files, Flash movies etc.
- Text processing features (XML documents)

Programming Styles.

(1) Procedural programming ⇒ COBOL

(2) Functional " ⇒ write the logic as multiple functions and call them in a specific order.

(3) object oriented "

↳ Data + methods (How to manipulate)

→ MySQL / MariaDB

→ PostgreSQL

→ DBASE

→ Oracle

→ Microsoft access; etc.

↳ This is critical for developing DSS.

↳ Built-in functions (or) canned operations / packages available

↳ textual data processing capability.

6

Now, why we choose PHP 5, why PHP 5? This is an interesting question for us because of the two different types of programming styles, there are multiple programming styles many people depending upon whom you follow, you will see different. I broadly take three things:

1) One is called Procedural Programming
2) Second one is what you call Functional Programming
3) Third one is called Object Oriented Programming.

As far as I am concerned, these three are fine with me, there is some distinction between procedural and functional, I would like to distinguish them.

Procedural Programming is very common. You can think about languages like cobalt or basic and all of them are good in that, they have a go to kind of usage also.

Functional Programming is where you write the logic as multiple functions and call them in a specific order. So, that is Functional Programming.

So, Object Oriented Programming is data + method combined. So, both the data and the method or manipulation methods means how to manipulate. Both are part of the Object-Oriented Programming.

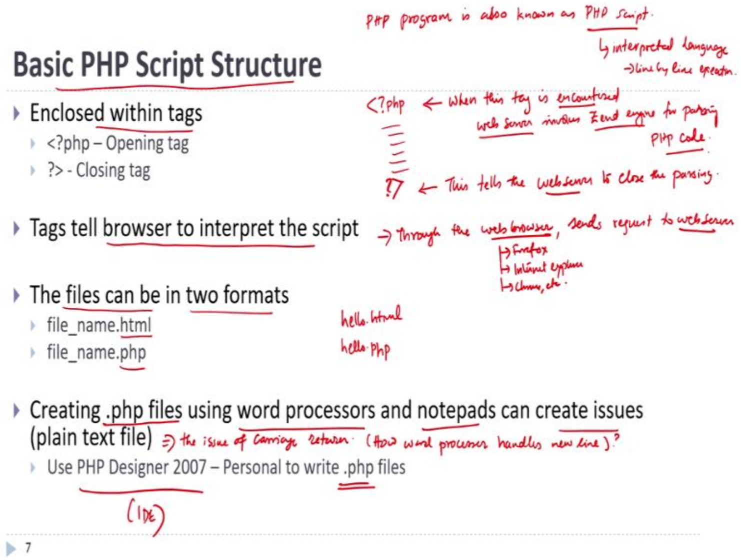
i) The PHP 5 allows both styles of Object-Oriented Programming and Functional Programming.

ii) I wrote Procedural Programming because it also supports Procedural Programming. So, all sorts of programming, the three major programming styles that I am more worried about, are supported by PHP 5.

- Second one is, it supports a wide range of Database Management Systems and this is critical for developing DSS. What is the wide range of DBMS that we support? It supports MySQL, Maria DB. It supports Postgres or PostgreSQL, it supports DBASE, Oracle, even to an extent, it supports even what we call Microsoft Access, this also happens, etcetera. The wide variety of Database Management Support embedded into PHP, makes it a very favorite choice of the people who are developing DSS or Decision support systems.
- Then, third capabilities it can generate images, PDF files and flash movies, etcetera, on the fly. So, this is built-in functions or canned operations, packages available. So, you do not have to write a specific code for this anymore. All this is already built-in, you just have to call that operation or the package and say I want to create this particular PDF file and PHP will create it for you.
- And, it also can process capabilities to process text or XML documents or the XML stands for Extended Markup Language. So, textual data processing capability, that is also another important aspect of PHP 5. Because of all these kinds of reasons, because it can support Object Oriented Programming, it can support the Functional Programming, it can support the Procedural Programming, it supports wide varieties of DBMS (Database Management Systems), it can create images, PDF files, flash movies, etcetera in a fly because all these things are built-in as a package, you do not have to write the code, it is available.

And, it can also do textual data processing, all these aspects which are very, very critical for the Decision Support System, it is readily available with PHP and especially in enhanced mode in PHP 5. So, we focus on PHP 5 or higher versions of it, so it acts for us to design the Web-based Decision Support System.

(Refer Slide Time: 26:27)



PHP program is also known as PHP Script.
↳ interpreted language
→ line by line execution.

Basic PHP Script Structure

- Enclosed within tags
  - <?php – Opening tag
  - ?> - Closing tag

<?php ← When this tag is encountered
web server invokes Zend engine for parsing
PHP code.

?> ← This tells the webserver to close the parsing.

- Tags tell browser to interpret the script

→ Through the web browser, sends request to webserver
↳ Firefox
↳ Internet explorer
↳ Chrome, etc.

- The files can be in two formats
  - file_name.html
  - file_name.php

hello.html
hello.php

- Creating .php files using word processors and notepads can create issues (plain text file) ⇒ the issue of carriage return. (How word processor handles new line)?
  - Use PHP Designer 2007 – Personal to write .php files

(IDE)

7

So, what is the basic scripting structure or how is the basic PHP program? So, a PHP program is also known as a PHP script, we are telling it a PHP script. The reason why we call it as the script is because it is an interpreted language, line by line execution. Just like a movie script, you go line by line. That is why it is called a PHP script.

- ➢ Now, the basic structure is, it is enclosed within two tags. It is an opening tag and a closing tag.

i) So, the opening tag is <?php. When this tag is encountered, Web Server invokes ZEND engine for parsing PHP code, as soon as the PHP tag is less than or equal to sign question mark and PHP comes across. This tag, when it encounters the Web Server, whether it is IIS, whether it is Apache, whether it is GWS, it does not matter, they invoke this ZEND engine and that ZEND engine parses the PHP code. And, how long it will continue to parse.

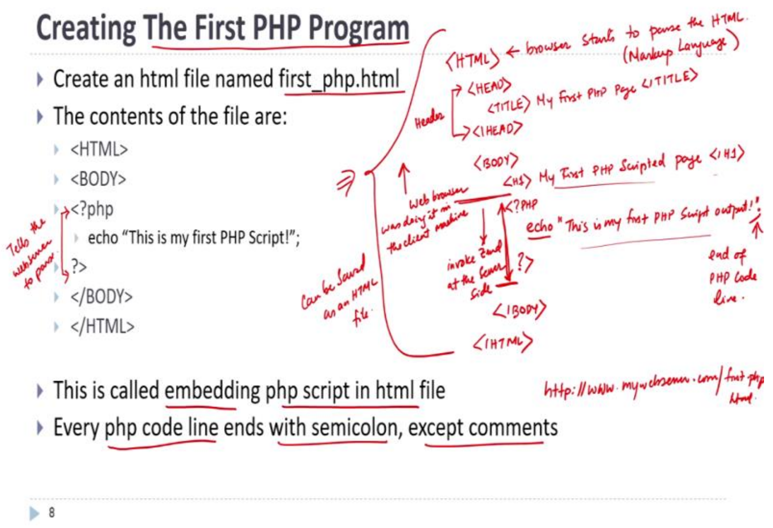
ii) You have PHP code like this and at some place you will have a closing tag ?>. This tells the Web Server to close the parsing. So, that is what the Web Server does.

- ➢ So, how does the Web Server know? The intermediary is the Web Browser. So, through the Web Browser. Which are the Web Browsers? Firefox, Internet Explorer, Chrome, etcetera. They are Web Browsers. Through the Web Browser, this Web Browser sends requests to the Web Server. So, where is the Web Server residing? It is residing on the computer server and then, it parses using ZEND engine and then, sends the code back to the client or the Web Browsers which it displaces.

- And, the PHP script files, these files can be in two formats. There are two approved formats for this with a dot html extension or a dot PHP extension. So, if I can say hello.html, I can put PHP in this or I can create a file called hello.php, and then, put the PHP file in that or PHP code in that.
- But, also remember a lot of the time people have this or when the people creating this application, they create these dot php files using word processors or notepad. Word processor typically what I am talking about is Microsoft Word and you can create an issue. Usually, the code may not parse, purely because of the fact that the issue of carriage return. The way in which they enter for a new line, how word processor handles new lines is an issue especially for the word processor or a notepad can create an issue for your ZEND engine.

So, hence we recommend you to use it, if you are writing the PHP file, think about this open IDE. This is an IDE (Integrated Development Environment), PHP designer 2007. It is free, it is easily available in the website on the internet and download it and write code accordingly using this one. So, this will prevent the issue of the carriage return when you are trying to learn the PHP code. So, that is the basic PHP structure.

(Refer Slide Time: 30:59)



Creating The First PHP Program

- Create an html file named first_php.html
- The contents of the file are:
  - <HTML>
  - <BODY>
  - <?php
    - echo "This is my first PHP Script!";
  - ?>
  - </BODY>
  - </HTML>
- This is called embedding php script in html file
- Every php code line ends with semicolon, except comments

Tells the webserver to parse
Can be saved as an HTML file.
<HTML> ← browser starts to parse the HTML. (Markup Language)
Header
<HEAD>
<TITLE> My first PHP Page </TITLE>
</HEAD>
<BODY>
<H1> My First PHP Scripted page </H1>
Web browser was doing it on the client machine
<?PHP
echo "This is my first PHP Script output!";
invoke Zend at the Server side
?>
end of PHP Code line.
</BODY>
</HTML>
http://www.mywebserver.com/first.php.html
8

So, let us create the first PHP program and once we go through this you may have a good idea. So, first name the file as first_php.html.

So, as I remember, we can elaborate on this file. So, the content of the file is the first line is HTML, then BODY and there is this statement PHP. So, this tells the Web Server to parse and then, it goes to HTML. So, now let me elaborate this script, let us elaborate this.

```
<HTML>
<HEAD>
<TITLE> My first PHP page </TITLE>
</HEAD>
<BODY>
<H1> My first PHP Scripted page </H1>
<?PHP
```

The minute this is up to this point the Web Browser was doing it in the client machine, up to this point the Web Browser was doing it in the client machine. Below this, it will invoke ZEND at the server side.

So, in the PHP we will say that using the keyword called Echo, stands for Print, that is what it says to display on the CRT, what it is.

```
Echo “This is my first PHP script output!”;
?>
</BODY>
</HTML>
```

So, this from <?PHP to ?> the server-side parsing happens.

So, this kind of an approach where you are integrating or embedding PHP script in the HTML file. So, this is one way of programming. In the HTML file itself you can create a PHP script by giving this PHP opening and closing tag. But remember, every PHP code line ends with a semicolon (;). So, this is the critical, end of PHP critical code. Every line should be done with a semicolon except comments. Comment lines should not be done with the help of a semicolon. So, that is the other aspect.

So, this whole file, this entire thing can be saved as an HTML file and when you open this HTML file, let us say you put it on some server. So, you will write it as http:/www.my webserver.com/first-php.html. The minute you do this, assume that this file is saved in the root directory, then, this will be taken, the Web Browser will show you and print this heading will be shown, this will also be shown and that will be the end of the program. So, this gives you an idea of how to create the first PHP program.

We will take a short break here and then we will do the next portion to continue with the more complicated PHP programs in the next session. Thank you.