

Q1. Explain the differences between External & Internal DDT's.

Internal DDT	External DDT
1. In internal DDT, elements are declared within the XML files.	In an External DDT, elements are declared outside the XML file.
2. To reference it as internal DDT, standalone attribute in XML declaration must be set as 'yes'.	To reference it as external DDT, standalone attribute in XML declaration must be set as 'no'.
3. It means the declaration works independent of external source.	It means the declaration works / includes information from external source.
4. Syntax for internal DDT:	Syntax for external DDT:
<!DOCTYPE root-element [declaration] element-declaration>	<!DOCTYPE root-element SYSTEM "File-name">

Q2. List and describe any 5 HTML tags with 1 example.

- 1) HTML tag - It is the root of HTML document which is used to specify that the document is HTML.

Syntax: <html> statements ... </html>

HTML
Structure

Summary

TOP

Ex: <html> *content enclosed with html tag*

<head>

<title> Web Technology </title>

</head>

</html>

2) Head tag: Used to contain all the **head** element in the html file. It contains title, style etc.

Syntax:

<head> statements...</head>

Ex: <html> *content enclosed with html tag*

<head> *content enclosed with head tag*

<title> Web Technology </title>
</head>

3) Body tag: Used to define the **body** of HTML document. It contains Image, tables, etc.

Syntax: <body> statements...</body>

Ex: <html> *content enclosed with html tag*

<head>

<title> Web Technology </title>

<body> HTML contents </body>

</head> *content enclosed with head tag*

</html> *content enclosed with html tag*

4) Title tag: Used to define title of HTML document.

Syntax: <title> statements...</title>

<title> statements...</title>

Ex: <title> tab name </title>

Q3] Heading tag :- Used to define heading of HTML document.

Syntax:

It will contain **<h1>** statements. **</h1>** is self closing tag.

It will contain **<h2>** statements. **</h2>** marks.

It will contain **<h3>** statements. **</h3>** end marker.

It will contain **<h4>** statements. **</h4>** end marker.

Example: **<h1>Heading 1</h1>**

<h2> Heading 2</h2> and **<h3> Heading 3</h3>**

<h4> Heading 4</h4> and **<h5> Heading 5</h5>**

<h6> Heading 6</h6> and **<h7> Heading 7</h7>**

Q3. What are DTDs? Explain how they work.

i) DTD stands for Document Type Definition. It describes the structure of a document & something about the data.

ii) It is set of markup declarations that actually define a type of document for the SGML family, like GML, SGML, HTML, XML.

iii) A DTD can be declared inside an XML document as inline or as an external recommendation.

iv) DTD determines how many times a node should appear, and how their child nodes are ordered.

v) There are two data types,

1) PCDATA - parsed character data

2) CDATA - character data, not usually parsed.

Syntax: **<!DOCTYPE element DTD identifier**

[first declaration

:

]> nth declaration

Q4 Define the terms i) Website ii) Web page iii) Web server
iv) URL v) Home page.

- i) Website - It is a place connected to the Internet, where company, organisation etc. put information that can be found on the world wide web.
- ii) Web Page - It is a specific collection of information provided by a website & displayed to a user in a web browser.
- iii) Web server - It is a server software or hardware dedicated to running this software, that can satisfy client request on the world wide web.
- iv) URL - (also known as web address), is an internet or intranet name that points to a location where a file, directory, etc. is stored.
- v) Home page - It is a webpage that serves as a starting point of website.

Wt

Assignment 2 (Unit II)

Aditi Dinesh Mwany
T.E. comp Div: A
Roll no: 02

Q1. Explain the difference between JavaScript & jQuery.

Difference between JavaScript and jQuery	
1. It is a javascript library.	It is a dynamic and interpreted web-development programming language.
2. It is quite an easy/simple and fast approach.	It is weakly typed programming approach.
3. There is no requirement for handling multi-browser capabilities issues.	Developer develops their own code for handling multi-browser capability.
4. It contains only few line of codes.	The code can be complicated as well as long.
5. jQuery is an optimized technique for web designing.	JavaScript is one of the popular web designing programming languages for developers that introduced jQuery.

Q2. List and describe any five functions in Javascript.

- 1) eval() - Evaluates a string / arithmetic expression and returns a value.
- 2) parseInt() - Parse a string argument and returns

Ques 2) Explain String class.

Ans 2) String class

↳ Contains a constructor

↳ Contains a destructor

↳ Contains a method to convert an integer of specified base.

↳ parseFloat() - Parse a string argument & returns a floating-point number.

↳ escape() - Returns a hexadecimal encoding of an input string argument.

↳ unescape() - Returns the ASCII string for the specified value.

Q3: How to create the objects in JavaScript? Explain with ex.

→ 1) Define and create a single object, using object literal.
It is a list of name:value pairs inside curly braces { }.

Ex: var person = {firstName: "Aditi", lastName: "Mulay", age: 20}

2) Using keyword new to create an object.

Ex: var person = new Object();

person.firstName = "Aditi";

person.lastName = "Mulay";

person.age = 20;

person.eyeColor = "black";

3) By defining an object constructor, then creating objects of constructed type.

i) Objects are mutable, They are addressed by reference not by value.

Ex: var x = person.

QUESTION

Q1. Here, the object `x` is not copy of `person`. It is `person`. They both are objects.

Example:

```
let person = {firstName: "Aditi", lastName: "Mulay",  
    address: "Mumbai", age: 10};  
let x = person; // x is not copy of person  
x.age = 11; // x.age is now 11, not 10;
```

Q2. Explain the different Data Types available in Javascript.

→ 1) String - represents textual data.

ex: 'Hello', "HelloWorld", "1234567890"

2) Number - an integer or a floating-point number.

ex: 3, 3.2234, 3e-2

3) BigInt - an integer with arbitrary precision.

ex: 9007199251249n, BigInt('1234567890')

4) Boolean - Any of two values: true or false.

ex: true and false

5) null - denotes a null value.

ex: let a=null;

6) Symbol - data type whose instances are unique & immutable.

ex: let value=Symbol('hello');

7) undefined - a data type whose variable is not defined.

ex: let a;

WT

Assignment - 3 (Unit III)

Aditi Dinesh Mwaj
T.E.comp, Div: A
Roll no: 02

Q1. Write Difference between `<include action` & `<include directive` in JSP.



Include Directive

1. It is one of the three directives supported by JSP ex. @taglib, @include, @page.
2. It is processed at the translation time.
3. It can use both relative & absolute path.
4. In this, we cannot pass any other parameter i.e. request, response.
5. It includes contents of resources like HTML, CSS file, but it won't process the dynamic resource.

Include Action

- It is one of the many std actions supported by JSP ex. <jsp:include>, <jsp:plugin>
- It is processed at the request time.
- It always uses relative path.
- In this, we can pass another parameter also using <jsp:param> tags.
- It processes the dynamic resource & result will be added to calling JSP.

Q2. List and elaborate any 5 JSP implicit objects with ex.



1) JSP out - For writing any data to buffer, JSP provides an implicit object named out. It is the object of JspWriter.

Ex: , displaying date & time.

index.jsp.

```
<html>
```

```
  <body>
```

```
    <%.out.print("Today is:" + java.util.Calendar.get  
    getInstance().getTime()); %>
```

```
  </body>
```

```
</html>
```

Output: Today is: Fri May 28 15:37:29 IST 2024.

2) JSP request - It is of type HttpServletRequest. It can be used to get request information such as parameter, server name, server port. It can also be used to set, get and remove attributes from jsp request scope.

Ex: index.html.

```
<form action = "welcome.jsp">  
  <input type = "text" name = "uname">  
  <input type = "submit" value = "go"><br/>  
</form>
```

Welcome.jsp

```
<%
```

```
String name = request.getParameter ("uname");
```

```
out.print ("welcome" + name);
```

```
%>
```

3) JSP response - It is of type HttpServletResponse. The instance of it is collected by web

Container for each jsp request. It can be used to add or manipulate response.

Ex: index.html

```
<html>
<body>
<form action = "welcome.jsp">
<input type = "text" name = "uname">
<input type = "submit" value = "go"><br/>
</form>
```

welcome.jsp

```
<%@ page import = "java.util.*" %>
<%@ page import = "java.io.*" %>
response.sendRedirect("http://www.google.com");
%>
```

↳ JSP config. - It is of type `ServletConfig`. This object can be used to get initialization parameter for a particular JSP page. Used to get initialization parameter from web.xml file.

Ex: web.xml file

```
<web-app>
<servlet>
<servlet-name>aditimay</servlet-name>
<jsp-file>/welcome.jsp</jsp-file>
<init-param>
<param-name>dname</param-name>
<param-value>sun.jdbc.odbc.JdbcodbcDriver</param-value>
</init-param>
<servlet-mapping>
<servlet-name>aditimay</servlet-name>
<url-pattern>/welcome</url-pattern>
</servlet-mapping>
</web-app>
```

Welcome.jsp! To display database with a variable
<%>

```
out.print("Welcome "+request.getParameter("uname"));
String driver = config.getInitParameter("dname");
out.print("driver name is "+driver);
%> // <%> is a JSP directive, not a tag
```

Q3] JSP application - It is of type ServletContext. Used to get initialization parameter from configuration file.

Ex:-

Q3. What is use of JSP action tags? Discuss in details

→ jsp:useBean action tag - Used to locate or instantiate a bean object.

↳ Each JSP action tag is used to perform some specific tasks. The action tags are used to control the flow between pages and to use Java Bean.

jsp:useBean action tag - Used to locate or instantiate a bean class. If bean object of the Bean class is already created, it doesn't create the Bean depending on the scope. But if object of bean is not created, it instantiates the bean.

Syntax of jsp:useBean action tag.

```
<jsp:useBean id="instanceName" scope="page|request|session|
class="packageName.className" type="packageName.className"
beanName="packageName.className | <%> = expression >">
</jsp:useBean>
```

Q4.

Write advantages of JSP over servlets? explain life cycle of JSP.

→

- i) Servlets use print statements for printing an HTML document which is usually very difficult to use. JSP has no such tedious task to maintain.
- ii) JSP needs no compilation, CLASSPATH setting & packaging.
- iii) In a JSP page visual content and logic are separated which is not possible in a servlet.
- iv) Usually with JSP, Java Beans and custom tags web application is simplified.

Life cycle of JSP:

- i) It is defined as the process from its creation till the destruction. This is similar to a servlet life cycle with an additional step which is required to compile a JSP into servlet.
- ii) Path followed by JSP:
 - 1) Compilation
 - 2) Initialization
 - 3) Execution
 - 4) Cleanup.

Wt

Assignment 4 (Unit IV)

Aditi Dinesh Mulay
T.E. Comp Div: A
Roll no: 02

Q1 What is multidimensional arrays in PHP? Explain with simple PHP code.

- i) An array that contains one or more arrays is multidimensional arrays.
- ii) A multidimensional array of each element in the main array can also be an array. And each element in sub-array can be an array and so on.
- iii) Values in the multi-dimensional array are accessed using multiple indexes.

Example:

```
<?php  
$marks = array (119, 90, 85);  
"Aditi" => array ("Physics" => 95, "maths" => 90, ),  
"Aarya" => array ("Physics" => 92, "maths" => 97, );  
echo "Marks for Aditi in physics : ";  
echo $marks ["Aditi"] ['Physics'] . "In";  
echo "Marks for Aarya in maths : ";  
echo $marks ["Aarya"] ['Maths'] . "In";
```

?> Marks for Aditi in physics : 95

Marks for Aarya in maths : 97

Q2. Write differences between client side and scripting & server side scripting.

→ Client side scripting is controlled by browser.

Client Side Scripting

JavaScript

Client Side Scripting

1. Source code is visible to user.

2. It usually depends on browser and its version.

3. It runs on user's computer.

4. HTML, CSS & JavaScript are used.

Q3. Classify data types of PHP and describe various datatypes in each type.

→

↳ Data Types define the type of data a variable can store. There are three main that are,

1) Predefined data types. :- It includes four more,

1) Boolean - Used in conditional Testing. Hold two values: true & false.

2) Integer - Hold positive & negative numbers. range - -2^{31} to 2^{31}

3) Double - hold nos. containing fractional or decimal parts including exponential form.

4) String - Hold letters or any alphabets.

Server Side Scripting

Source code is not visible to user because it's output of server side is a HTML page.

In this any server side technology can be used & it does not depend on client.

It runs on web server.

PHP, Python, Java, Ruby are used.

Q3. User-defined data types

1) Arrays - Array is a compound data-type that can store multiple values of same data type.

2) Objects - They are instances of user-defined classes that can hold both values & functions & information for data processing specific to class.

3) Special -data types -

1) NULL - Hold only one value i.e. NULL.

2) Resources - Used to store references to some external functions call or to external PHP resources.

Q4. What is an associative array in PHP? Give code.

→ i) Associative arrays are used to store key value pairs.

Ex: PHP code for above array :-

<?php \$arr = array("Maths"=>95, "Physics"=>90, "Computer"=>

"Maths"=>95, "Physics"=>90, "Computer"=>98, "English"=>93); ?>

echo "Marks for student one is: \n";

echo "Maths:" . \$student-one["Maths"], "\n";

echo "Physics:" . \$student-one["Physics"], "\n";

echo "Computer:" . \$student-one["Computer"], "\n");

echo "English:" . \$student-one["English"], "\n";

Output:- Marks for student one is:

Maths : 95

Physics : 90

Computer: 98

English : 93

Wt

Assignment V5(Unity)

Q1. Angular JS supports single page application via multiple views on a single page. Justify answer.



- i> To do this, AngularJS has provided ng-view and ng-template directives and \$routeProvider services.
- ii> ng-view Directive - It simply creates a place holder where a corresponding view can be placed on the configuration.

Usage: <div ng-app = "mainApp">

-----<div ng-view></div>-----

- iii> ng-template Directive - Used to create an HTML view using script tag. It contains id attributes which is used by \$routeProvider to map a view with a controller.

- iv> \$route - Provider Service - It is a key service which sets the configuration of URL's, maps them with the corresponding HTML page or ng-template & attaches a controller with the same.

Q2. List and explain events supported by AngularJS to enrich form filling and validation.



- j> We can use ng-click event to handle the click button and use \$dirty and \$invalid flags to do the validation in a seamless way. It supports following events, ng-click, ng-dbl-click, ng-mousedown, ng-mouseup, ng-mouseenter, ng-mouseleave, ng-mousemove, ng-mouseover, ng-keydown, ng-keyup

ng-keypress, ng-change.

- i) for the validation we can use,
 - \$dirty - states that values has been changed.
 - \$invalid - states that value enter is invalid.
 - \$error - states the exact error.

Q3. Write short notes on Node.js.

→ Node.js

- i) Node.js is a platform built on Chrome's JavaScript runtime for easily building fast and scalable network applications.
- ii) Node.js uses an event-driven, non blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.
- iii) All APIs of Node.js library are asynchronous that is non-blocking.
- iv) Node.js library is very fast in code execution.
- v) Node.js applications never buffer any data It simply output the data in chunks.

Q4. List and explain different directives in AngularJS.

→ AngularJS Directives

- i) ng-app Directive - It starts an AngularJS application. It defines the root element. It automatically initializes or bootstraps the application when the web page containing AngularJS application.

Ex:

```
<div ng-app="">  
---  
</div>
```

2) ng-init Directive - It initializes an AngularJS application's data. It is used to assign values to the variables.

3) ng-model directive - It defines the model variable to be used in AngularJS application.

Ex: <div ng-app="">

<p> Enter your Name: <input type="text" ng-model = "name"> </p>

</div>

4) ng-repeat Directive - It repeats the HTML elements for each item in a collection.

Wt

Assignment VI (Unit VI)

Aditi Dinesh Mulay
T.E. Comp. Div: A
Roll no: 02

Q1.

Identify and justify the benefits of using web services.



i) Here are some benefits of using web services.

1) Interoperability - Web services allows various applications to talk to each other and share data and services among themselves. Web services are used to make the application platform and technology independent.

2) Standardized Protocol - Web services uses standardized industry standard protocol for the communication.

3) Low-cost communication - Web services use SOAP over HTTP protocol, so you can use your existing low-cost internet for implementing web services.

4) Exposing the Existing Function on network - A web service is a unit of managed code that can be remotely invoked using HTTP. That is, it can be activated using HTTPS requests.

Q2.

Write short note on Spring.

i) It is an open source Java platform, used by developers to create high, performing, easily testable and reusable code.

ii) It is lightweight when it comes to size and transparency. Basic version of Spring framework is around 2MB.

iii) Spring framework targets to make J2EE development easier to use and promotes good programming practices by enabling a POJO-based

programming model.

- *) It is organized in a modular fashion. Even though the number of packages and classes are substantial,
- *) Spring does not reinvent the wheel, instead it truly makes use of some of existing technologies like several ORM frameworks, logging frameworks, JEE, Quartz and JBoss timers.

Q3. Write short note on Bootstrap.

→

- i) Bootstrap is the most popular CSS framework for developing responsive and mobile-first websites.
- ii) Bootstrap 4 is the newest version of Bootstrap. It supports all major browsers except Internet Explorer 9.
- iii) The container class is one of the most important Bootstrap classes. It provides margins, padding, alignments, and more to HTML elements.
- iv) It is absolutely free to download and use. It can also use Javascript plug-ins. It facilitates you to create responsive designs.

Q4. What are the web services? List and Discuss components of web services.

→

- i) A web service is any piece of software that makes itself available over the internet and uses a standardized XML messaging system. XML is used to encode all communications to a web service. For ex. a client invokes a web service by sending a

- i) Client creates an XML message, then waits for a corresponding XML response.
- ii) Web services are self-contained, modular, distributed dynamic applications that can be described, published, located or invoked over the network to create products, processes and supply chains.
- iii) All standard web services work using following components.
- 1) SOAP (Simple Object Access Protocol)
 - 2) UDDI (Universal Description, Discovery and Integration.)
 - 3) WSDL (Web Services Description Language).