Scheme Of Evaluation

Internal Assessment Test 3 – JULY 2021

Sub:	Web Technology and its applications							**Code:**	18CS63
Date:	29 / 07 / 2021	Duration:	90mins	Max Marks:	50	**Sem:**	VI	**Branch:**	ISE

Note: Answer Any FIVE Questions

Question #		**Description**	**Marks Distribution**		**Max Marks**
1		Write a PHP program to do the multiplication of two matrices.	10 M	10 M	10 M
2		Write a PHP program to Upload a file File size restrictions Limiting type of file upload	3m 3m 4m	10m	10 M
3		Complete analyzing and depicting the UML	10 M	10 M	10 M
4		Explanation of append, Prepend & prependTo Wrap with Program in detail	3m 3m 4m	10M	10 M
5		Explanation of Dynamic updating of Web page using AJAX with figure	10 m	10M	10m
6		Write an XML program and define CSS in different file and display three students all information's like name, dob, email id, date of joining, usn etc.,	10m	10M	10M

1. **Write a PHP program to do the multiplication of two matrices.**

Solution :

```
<?php
$a = array(array(1,2,3),array(4,5,6),array(7,8,9));
$b = array(array(7,8,9),array(4,5,6),array(1,2,3));
$m=count($a);
$n=count($a[2]);
$p=count($b);
$q=count($b[2]);
echo "the first matrix :"."<br/>";
for ($row = 0; $row < $m; $row++) {
for ($col = 0; $col < $n; $col++)
echo " ".$a[$row][$col];
echo "<br/>";
}
echo "the second matrix :"."<br/>";
for ($row = 0; $row < $p; $row++) {
for ($col = 0; $col < $q; $col++)
echo " ".$b[$row][$col];
echo "<br/>";
}

echo "the transpose for the first matrix is:"."<br/>"; for ($row = 0; $row < $m;
$row++) {
for ($col = 0; $col < $n; $col++)
echo " ".$a[$col][$row];
echo "<br/>";
}
if(($m===$p) and ($n===$q)) {
echo "the addition of matrices is:"."<br/>";
for ($row = 0; $row < 3; $row++) {
for ($col = 0; $col < 3; $col++)
echo " ".$a[$row][$col]+$b[$row][$col]." "; echo "<br/>";
}
}
if($n===$p){
echo " The multiplication of matrices: <br/>";
$result=array();
for ($i=0; $i< $m; $i++) {
for($j=0; $j < $q; $j++){
$result[$i][$j] = 0;
for($k=0; $k < $n; $k++)
```

```
$result[$i][$j] += $a[$i][$k] * $b[$k][$j];
} }
for ($row = 0; $row < $m; $row++) {
for ($col = 0; $col < $q; $col++)
echo " ".$result[$row][$col];
echo "<br/>";
} }
?>
```

2. Write a PHP program to do the following operations

a. Upload a file

b. File size restrictions (max size= 10KB)

c. Limiting type of file upload (only PDF accepted)

Solution (a)

```
<?php

<form enctype='multipart/form-data' method='post'>
<input type='file' name='file1' id='file1' />
<input type='submit' />
</form>
?>
```

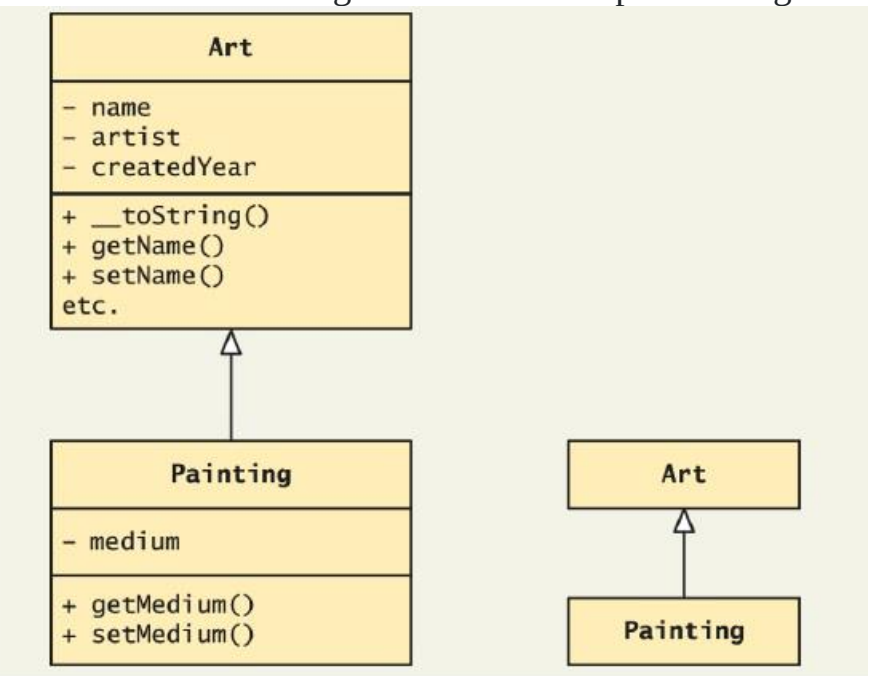
Solution (b)

```
<?php>
<form enctype='multipart/form-data' method='post'>
<input type="hidden" name="MAX_FILE_SIZE" value="10" />
<input type='file' name='file1' />
<input type='submit' />
</form>
?>
```

Solution (c)

```
<?php>
$validExt = array("pdf”);
$validMime = array(“text/pdf”);
foreach($_FILES as $fileKey => $fileArray )
{
$extension = end(explode(".", $fileArray["name"]));
if (in_array($fileArray["type"],$validMime) && in_array($extension,
$validExt))
{ echo "all is well. Extension and mime types valid";
}
else
{ echo $fileKey." Has an invalid mime type or extension";
}
}
?>
```

3. Observe the given UML and depict it using PHP




```
class Art {
private $name;
private $artist;
private $yearCreated;

function __construct($year, $artist, $name) {
$this->setYear($year);
$this->setArtist($artist);
$this->setName($name);
}
public function getYear() { return $this->yearCreated; }
public function getArtist() { return $this->artist; }
public function getName() { return $this->name; }

public function setYear($year) {
if (is_numeric($year))
$this->yearCreated = $year;
}
public function setArtist($artist) {
if ((is_object($artist)) && ($artist instanceof Artist))
$this->artist = $artist;
}
public function setName($name) {
$this->name = $name;
}
```

```
public function __toString() {
$line = "Year:" . $this->getYear();
$line .= ", Name: " .$this->getName();
$line .= ", Artist: " . $this->getArtist()->getFirstName() . ' ';
$line .= $this->getArtist()->getLastName();
return $line;
}
}
class Painting extends Art {
private $medium;
function __construct($year, $artist, $name, $medium) {
parent::__construct($year, $artist, $name);
$this->setMedium($medium);
}
public function getMedium() { return $this->medium; }
public function setMedium($medium) {
$this->medium = $medium;
}
public function __toString() {
return parent::__toString() . ", Medium: " . $this->getMedium();
}
}
```

4. **Describe how to modify DOM in jQuery**
   Solution :

Modifying the DOM

jQuery comes with several useful methods to manipulate the DOM elements themselves.

Creating DOM and textNodes

If you decide to think about your page as a DOM object, then you will want to manipulate the tree structure rather than merely manipulate strings. jQuery is able to convert strings containing valid DOM syntax into DOM objects automatically. The jQuery methods to manipulate the DOM take an HTML string, jQuery objects, or DOM objects as parameters, you might prefer to define your element as

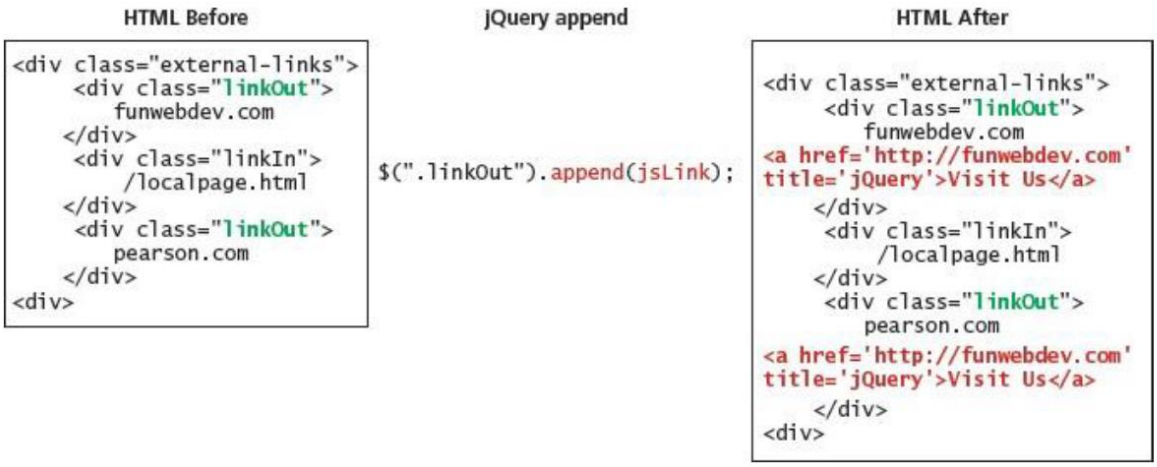
var element = $(" "); *//create new DOM node based on html*

This way you can apply all the jQuery functions to the object, rather than rely on pure JavaScript, which has fewer shortcuts. If we consider creation of a simple <a> element with multiple attributes, you can see the comparison of the JavaScript and jQuery techniques.

Prepending and Appending DOM Elements

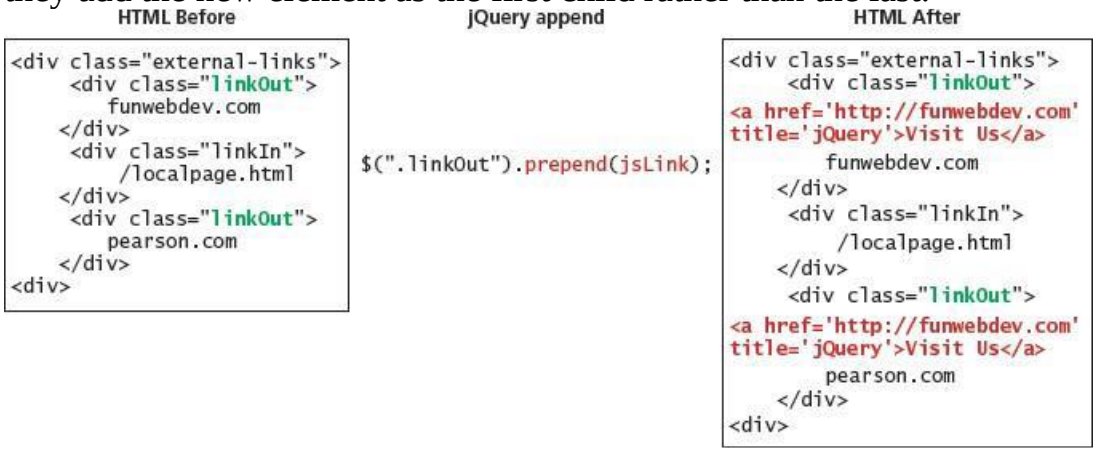
When an element is defined, it must be inserted into the existing DOM tree. You can also insert the element into several places at once if you desire, since selectors can return an array of DOM elements.

The append() method takes as a parameter an HTML string, a DOM object, or a jQuery object. That object is then added as the last child to the element(s) being selected. In below figure we can see the effect of an append() method call. Each element with a class of linkOut has the jsLink element appended to it.



The appendTo() method is similar to append() but is used in the syntactically converse way. If we were to use appendTo(), we would have to switch the object making the call and the parameter to have the same effect as the previous code:

jsLink.appendTo($(".linkOut"));
The prepend() and prependTo() methods operate in a similar manner except that they add the new element as the first child rather than the last.



Wrapping Existing DOM in New Tags

One of the most common ways to enhance a website that supports JavaScript is to add new HTML tags as needed to support some jQuery functions. Imagine for illustration purposes our art galleries being listed alongside some external links as described by the HTML below.

Uffuzi Museum

National Gallery

funwebdev.com

If we wanted to wrap all the gallery items in the whole page inside, another with class galleryLink we could write:

$(".gallery").wrap(' ');

which modifies the HTML

5. Describe how a web page is updated dynamically using AJAX.
Solution :

Asynchronous JavaScript with XML (AJAX) is a term used to describe a paradigm that allows a web browser to send messages back to the server without interrupting the flow of what's being shown in the browser. This makes use of a browser's multi-threaded design and lets one thread handle the browser and interactions while other threads wait for responses to asynchronous requests.
Below figure annotates a UML sequence diagram where the white activity bars illustrate where computation is taking place. Between the request being sent and the response being received, the system can continue to process other requests from the client, so it does not appear to be waiting in a loading state.

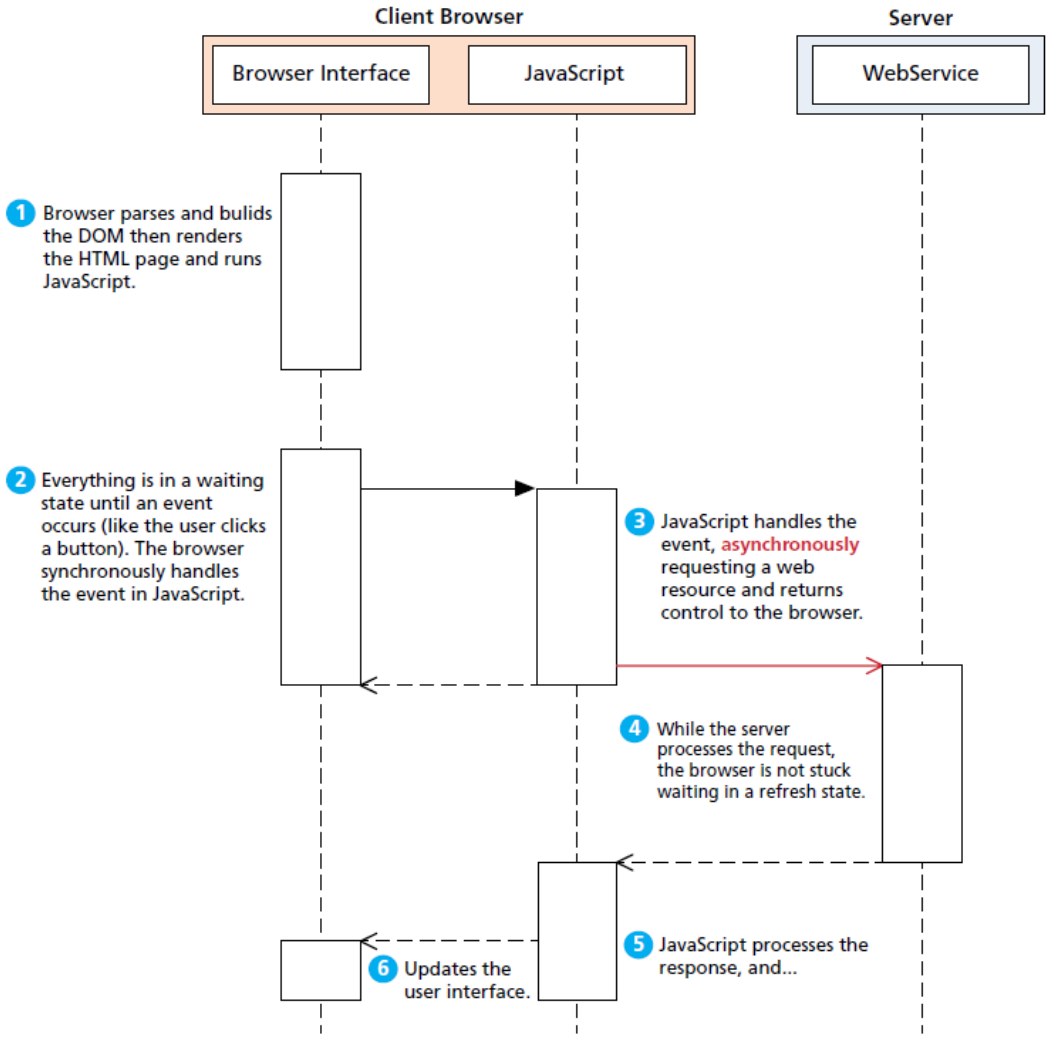


Figure 1 : UML sequence diagram of an AJAX request

Responses to asynchronous requests are caught in JavaScript as events. The events can subsequently trigger changes in the user interface or make additional requests.

6. **Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, and Name of the College, Programme, Year of Joining, and email id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.**

Soltion :

```
<?xml-stylesheet type="text/css" href="5.css" ?>
<!DOCTYPE HTML>
<html>
<head>
<h1> STUDENTS DESCRIPTION </h1>
</head>
<students>
<student>
<USN>USN :1CD15CS001</USN>
<name>NAME:AISHWARYA</name>
<college>COLLEGE:CITECH</college>
<branch>BRANCH :Computer Science and Engineering</branch>
<year>YEAR :2015</year>
<e-mail>E-Mail :aishwrya@gmail.com</e-mail>
</student>
<student>
<USN>USN :1CD15CS002</USN>
<name>NAME :ABHISHEK</name>
<college>COLLEGE:CITECH</college>
<branch>BRANCH :Computer Science and Engineering</branch>
<year>YEAR :2015</year>
<e-mail>E-Mail :abhi@gmail.com</e-mail>
</student>
<student>
<USN>USN :1CD15CS003</USN>
<name>NAME :BALU</name>
<college>COLLEGE:CITECH</college>
<branch>BRANCH :Computer Science and Engineering</branch>
<year>YEAR :2015</year>
<e-mail>E-Mail :balu@gmail.com</e-mail>
</student>
<student>
</students>
</html>
```

5.css

```
student{
display:block; margin-top:10px; color:Navy;
}
USN{
display:block; margin-left:10px;font-size:14pt; color:Red;
}
name{
display:block; margin-left:20px;font-size:14pt; color:Blue;
}
college{
display:block; margin-left:20px;font-size:12pt; color:Maroon;
}
branch{
display:block; margin-left:20px;font-size:12pt; color:Purple;
}
year{
display:block; margin-left:20px;font-size:14pt; color:Green;
}
e-mail{
display:block; margin-left:20px;font-size:12pt; color:Blue;
}
```