CS511 – Web Engineering

Final Terms, Fall 2024 **+92 318 1683 839**

Short Questions

1. Write names of HTTP methods used in web service?
2. Create PHP script attempts to connect a database.
3. Write PHP code that check cookies user name is set.
4. Write three pattern names in design of web page?
5. Write two names of development stacks in which we use Node.js as backend server.
6. PHP code for class name student.
7. How to create a table in Laravel?
8. For which purpose serialization used?
9. BookID colum 1 and colum 2 how book name will be stored?
10. Explain XHTML and write syntax.
11. Different event handling approaches in JavaScript.
12. Difference between GET and POST in AJAX.
13. Purpose of PHP superglobal variables.
14. Role of jQuery in web development.
15. What are jQuery selectors? Provide an example.
16. How does PHP handle form validation?
17. What is the purpose of serialize() in jQuery AJAX?
18. What is object-oriented programming in PHP?
19. What is the difference between JSON and XML?
20. Explain the difference between fadeIn(), slideDown(), and animate() in jQuery.
21. What is the difference between synchronous and asynchronous requests in AJAX?
22. What are superglobal variables in PHP? Name at least four.
23. What is the purpose of isset() and empty() functions in PHP?
24. What is the difference between append(), prepend(), before(), and after() in jQuery?
25. What is the difference between innerHTML, textContent, and innerText in JavaScript?

Long Questions

1. Create a PHP class with methods and properties.
2. Write a PHP code and apply switch in it.
3. If HTML code given, in the paragraph separate data was written in the tags, so with the help of jQuery, the first paragraph had to be written in place of the data: 'Virtual University of Pakistan'.
4. Write PHP script to start a session and check if a user is logged in?
5. Write a JavaScript program to validate a form with fields (name, email, password).
6. Explain jQuery animations with an example. Implement a button that fades in text when clicked.
7. Explain how you would parse JSON data in PHP and display it on a webpage.
8. Explain AJAX with an example that fetches and displays user details from a server without reloading the page.

Answers:

Short Questions

1. **HTTP methods used in web service:** GET, POST, PUT, DELETE.
2. **PHP script to connect to a database:**

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "mydatabase";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully";
?>
```

3. **PHP code to check if a cookie named "username" is set:**

```
php
CopyEdit
<?php
if(isset($_COOKIE['username'])) {
    echo "Username: " . $_COOKIE['username'];
} else {
    echo "Cookie is not set!";
}
?>
```

4. **Three design patterns for web page design:** Adapter Pattern, Simple Factory Pattern, Template Method Pattern
5. **Two development stacks using Node.js:** MEAN (MongoDB, Express.js, Angular, Node.js) and MERN (MongoDB, Express.js, React, Node.js).
6. **PHP class named "Student":**

```
<?php
class Student {
    public $name;
    public $age;

    function set_name($name) {
        $this->name = $name;
    }

    function get_name() {
        return $this->name;
    }
}
?>
```

7. **Creating a table in Laravel:**

```
php artisan make:migration create_students_table
```

Then, inside the migration file:

```
public function up()
{
    Schema::create('students', function (Blueprint $table) {
        $table->id();
        $table->string('name');
        $table->integer('age');
        $table->timestamps();
    });
}
```

Run migration with:

```
php artisan migrate
```

8. **Purpose of serialization:** It is used to convert objects into a format (like JSON or XML) that can be easily stored or transmitted.
9. **How book names are stored with BookID column 1 and 2:** It depends on the database design; usually, a relational approach with a "Books" table storing `BookID` and `BookName` is used.

10. **XHTML explanation and syntax:** XHTML is a stricter version of HTML with proper nesting and closing tags.
    Example:
```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head><title>XHTML Example</title></head>
<body><p>Hello World!</p></body>
</html>
```

11. **Event handling approaches in JavaScript:** Inline events, traditional event handlers, event listeners (e.g., `addEventListener`).
12. **Difference between GET and POST in AJAX:** GET is used for retrieving data, while POST is used for sending data securely.
13. **Purpose of PHP superglobal variables:** These are built-in variables like `$_GET`, `$_POST`, `$_SESSION`, `$_COOKIE` used to access request and session data.
14. **Role of jQuery in web development:** It simplifies DOM manipulation, AJAX requests, and animations.
15. **jQuery selectors example:**
```
$("p").hide(); // Hides all <p> elements
```

16. **How PHP handles form validation:** By using `isset()`, `empty()`, and regular expressions to check input fields.
17. **Purpose of `serialize()` in jQuery AJAX:** Converts form data into a query string format.
18. **Object-oriented programming in PHP:** Uses classes and objects to encapsulate data and behavior.
19. **Difference between JSON and XML:** JSON is lightweight and faster, while XML is more structured and widely used in SOAP web services.
20. **Difference between `fadeIn()`, `slideDown()`, and `animate()` in jQuery:**
    - `fadeIn()`: Fades in an element.
    - `slideDown()`: Slides down an element.
    - `animate()`: Custom animations.
21. **Difference between synchronous and asynchronous AJAX requests:** Synchronous waits for a response before proceeding; asynchronous allows execution to continue while waiting for the response.
22. **Four superglobal variables in PHP:** `$_GET`, `$_POST`, `$_SESSION`, `$_COOKIE`.
23. **Purpose of `isset()` and `empty()` functions in PHP:**
    - `isset()`: Checks if a variable is set and not null.
    - `empty()`: Checks if a variable is empty.
24. **Difference between `append()`, `prepend()`, `before()`, and `after()` in jQuery:**
    - `append()`: Adds content inside an element at the end.
    - `prepend()`: Adds content inside an element at the beginning.
    - `before()`: Inserts content before an element.
    - `after()`: Inserts content after an element.
25. **Difference between `innerHTML`, `textContent`, and `innerText` in JavaScript:**
    - `innerHTML`: Includes HTML formatting.
    - `textContent`: Plain text without formatting.
    - `innerText`: Similar to `textContent`, but may vary slightly between browsers.

Long Questions

1. **PHP class with methods and properties:**
```
<?php
class Car {
    public $brand;
    public $model;
    public $year;

    function __construct($brand, $model, $year) {
        $this->brand = $brand;
        $this->model = $model;
        $this->year = $year;
    }

    function getCarDetails() {
        return "$this->brand $this->model ($this->year)";
    }
}

$car1 = new Car("Toyota", "Corolla", 2022);
echo $car1->getCarDetails();
?>
```

2. **PHP script using switch:**

```php
<?php
$day = "Monday";

switch($day) {
    case "Monday":
        echo "Start of the week!";
        break;
    case "Friday":
        echo "Weekend is near!";
        break;
    default:
        echo "Have a nice day!";
}
?>
```

3. **Use jQuery to replace paragraph content:**

```jquery
$("p:first").text("Virtual University of Pakistan");
```

4. **PHP script to start a session and check login:**

```php
<?php
session_start();
if(isset($_SESSION['username'])) {
    echo "Welcome, " . $_SESSION['username'];
} else {
    echo "Please log in.";
}
?>
```

5. **JavaScript form validation:**

```javascript
function validateForm() {
    let name = document.forms["myForm"]["name"].value;
    if (name == "") {
        alert("Name must be filled out");
        return false;
    }
}
```

6. **jQuery fade-in button example:**

```jquery
$("#btn").click(function() {
    $("#text").fadeIn();
});
```

7. **Parsing JSON in PHP:**

```php
<?php
$json = '{"name": "John", "age": 30}';
$data = json_decode($json, true);
echo $data["name"];
?>
```

8. **AJAX example fetching user details:**

```jquery
$.ajax({
    url: "getUser.php",
    type: "GET",
    success: function(data) {
        $("#userDetails").html(data);
    }
});
```