

- Q1 Attempt any four Marks
- a. Write HTML code to construct the following table 5

My favorite textbooks

No.	Name	Author
1	Database Management Systems	Henry Korth
2	Automata Theory	jeffrey ullman
3	Data Mining	jiawei Han

- b. Write HTML and CSS code to construct the following navigation menus 5
- Home About Us Products Services Portfolio Contact
- c. Write the code using appropriate language to change the color of a button to yellow as soon as the mouse pointer is kept on the button. 5
- d. Discuss the features of Laravel framework. 5
- e. Write short notes on JQuery getters and setters. 5

- Q2 a. Write HTML code to build the following content. The user should be able to select only one car from the available cars such as: Audi, Nano, Alto, WagonR. If the user clicks on "Submit" button, it should re-direct it to "www.google.com" 10

My favorite cars

Choose your favorite car:

Choose a car:

- b. Write HTML / CSS code to construct the following content with nav-bar. By default "Home" should be selected with black color. The color of other nav-bar menu should get changed automatically to "green", if you put the mouse cursor on that nav-bar menu. 10

Home	Hello World!
News	Some text..
Contact	
About	

- Q3 a. Create a form using basic HTML to accept a number from the user with or without '+' or '-' sign. If the user has entered a pop-up box should appear with the message "you have entered a valid number". If the number entered by the user is not a valid number, then a message "you have entered an invalid number" should appear in the pop-up box. 10

- b. Write short notes on bootstrap grid system. Write the code using bootstrap to divide a page into two parts vertically. In left hand side part, list your favorite cricketers (any 4). In right hand side part, list your favorite scientists (any 4). 10

- Q4 a. Write the code in PHP to create a web page as shown below. When a file has been successfully uploaded the message "No image has been uploaded" should be changed to "image has been successfully uploaded". Note: only the mentioned file formats should be allowed for upload. 10

Select a JPG, GIF, PNG or TIF File: No file chosen

No image has been uploaded

- b. Write short notes on MVC architecture. 10

- Q5 a. Write the code using PHP and JQuery, to create following web page. The "Hide/Unhide me" button should be used to hide and unhide the text displayed on the web page alternatively. 10

I like web programming.

I like Database Management Systems

- b. Create four buttons with different colors using bootstrap. All four button should contain spinner (loading) symbol. Out of these buttons, one button should be in permanently disabled state. 10

- Q6 a. Write the code in PHP to connect to the database called as "mydb". The database username is "root" and the password for the user root is: "Test123". If the connection is successful, it should display the message "Connection established" on the screen. Otherwise it should print the message "Error in establishing connection". Now In this database create a table called "myguests" with the attributes "id", "first_name", "mobile_no" using PHP. 10

- b. Write the code using HTML and JavaScript, which will display alert box with the message "Please enter a pass phrase". When clicked on "ok" button, the code should check whether the password entered by the user contains characters a-z and numbers 0-9. If so, it should print the message "password contains valid characters". If any special character is present in the password, it should print the message "invalid password". 10

Q1a) HTML code to construct the following table:-

⇒ <HTML>

<HEAD><TITLE> Table </TITLE></HEAD>

<STYLE>

table, th, td {

border: 1px solid black;

}

</STYLE>

</HEAD>

<BODY>

<TABLE>

<CAPTION> My Favourite Textbook </CAPTION>

<THEAD>

<TR>

<TH> No. </TH><TH> Name </TH><TH> Author

</TH>

</TR>

</THEAD>

<TBODY>

<TR>

<TD> 1 </TD><TD> Database Management System </TD>

<TD> Henry Korth </TD>

</TR>

<TR>

<TD> 2 </TD><TD> Automata Theory </TD><TD> Jeffrey

Ullman </TD>

</TR>

<TR>

<TD> 3 </TD><TD> Data mining </TD><TD> Jiwei Han </TD>

</TR>

</TBODY>

</TABLE>

</BODY> </HTML>

reference ↓

No.	Name	Author
1	DBMS*	H.K
2	AT*	J.U
3	DM*	J.H

Q2b) HTML & CSS code for nav menus.

```
<HTML>
<HEAD> <TITLE> Navigation Menu </TITLE>
<STYLE>
    li {
        display : inline ;
    }
    li a {
        padding : 2px ;
    }
    a {
        text-decoration : none ;
        font-weight : bold ;
        color : black ;
    }
</STYLE> </HEAD>
</HEAD>
<BODY>
    <NAV>
        <UL>
            <li> <a href = "home.html"> Home </a> </li>
            <li> <a href = "about.html"> About Us </a> </li>
            <li> <a href = "products.html"> Products </a> </li>
            /* list 4 links namely products, services,
            portfolio, contact */
        </UL>
    </nav>
</BODY>
</HTML>
```


c) <! Doctype HTML>

<HTML>

<HEAD><TITLE> Button </TITLE>

<STYLE>

button: hover {

background-color: Yellow;

}

</STYLE>

</HEAD>

<BODY>

<BUTTON> Hover over me! </BUTTON>

</BODY>

</HTML>

d) Discuss features of Laravel framework.

Ans Features of Laravel framework are as follows:-

1) MVC Architecture: Laravel follows model-view-controller (MVC) architectural pattern, which promotes the separation of concerns in web application development. This separation enhances code organisation and maintainability.

2) Database Migrations and seeding: Laravel offers a migration system that helps to manage database schema changes. Migration enables version control and collaboration on database structure. Database seeding allows for populating databases with initial data for testing and development.

3) Authentication and authorization: Laravel provides built-in authentication and authorization systems with customizable configuration. Developers can easily implement user registration, login and role based access control.

- 4] Middleware: It is Middleware in laravel allows for filtering of HTTP requests. It is commonly used for tasks like authentication, logging and request modifications.
- 5] Validation: Laravel's validation system offers a straightforward way to validate user input and handle validation errors.
- 6] Security features: Laravel incorporates essential security features such as cross sites Request Forgery (CSRF) protection, SQL injection prevention and Cross site scripting (XSS) protection to safeguard web application.
- 7] Caching: Laravel supports caching mechanisms. Caching can significantly improve web application performance by storing frequently used data in memory.
- 8] Task scheduling: Laravel's task scheduling automates repetitive tasks using a concise and expressive syntax. This is particularly useful for background processing and maintaining tasks.
- 9] RESTful Routing: Laravel's routing system simplifies the creation of RESTful routes. This is crucial for developing APIs web services following RESTful principles.
- 10] Testing and PHPUnit: Laravel encourages and supports testing through PHPUnit. This ensures the reliability and quality of web application by allowing developers to write unit & integration tests.

Q) Write short notes on JQuery getters and setters.

Ans Getters in JQuery:-

i) Getters in JQuery are methods that retrieve information or values from selected HTML elements.

ii) Commonly used getters are:

a) '.text()' Getter: The '.text()' method in JQuery is a getter used to retrieve the combined text content of selected HTML elements.

eg: `var textContent = $('p').text();`

b) '.html()' getter: This method is a getter that retrieves the HTML content of the first selected element.

eg: `var htmlContent = $('div').html();`

c) '.val()' Getter: The '.val()' method serves as a getter for retrieving the current value of form elements such as input, select and text area.

eg: `var inputValue = $('input').val();`

d) '.attr()' getter: The '.attr()' method, when used as a getter, fetches the value of specified attribute for the first element in the set of matched element.

eg: `var hrefValue = $('a').attr('href');`

iii) JQuery Setter: Setters in JQuery are methods that assign or update values to selected HTML elements.

a) '.text()' setter: The '.text()' method, when used as a setter, is employed to set the text content of selected HTML elements.

eg: `$('#p').text('New text content');`

b) '.html()' setter: The '.html()' method, when used as a setter, updates the HTML content of the selected elements.

eg: `$('#div').html('<p>New HTML content </p>');`

c) '.val()' setters: The '.val()' method serves as a setter for setting the value of form elements.

eg: ~~\$('input')~~ \$('input').val('New Value');

d) '.attr()' Setters: The '.attr()' method, when used as a setter, updates the value of a specified attribute for the selected elements.

eg: \$('a').attr('href', 'https://example.com');

Q2] a) HTML code:-

<HTML>

<H1 <HEAD> <TITLE> @Form </TITLE> </HEAD>

<BODY>

<div> <H2> My Favourite Cars </H2> </div>

<div> Choose your favourite car: </div>

<form action = "https://www.google.com">

<label for = "cars"> Choose a car: </label>

<select id = "cars" name = "cars">

<option value = "audi"> Audi </option>

<option value = "nano"> Nano </option>

< " " = "Alto"> Alto </option>

< " " = "WagonR"> WagonR </option>

</select>

<input type = "submit" value = "submit">

</form>

</BODY>

</HTML>

Q2b) <HTML>

<HEAD>

<TITLE> Navigation bar </TITLE>

<STYLE>

body {

margin : 0 ;

padding : 0 ;

font-family : Arial , sans-serif ;

display : flex /* to make the body a flex, container #1

flex-direction : row

}

nav {

background-color : lightgray ;

width : 100% ;

/*
nav a {

color : black ;

padding : 14px 16px ;

text-decoration : none ;

font-size : 17px ;

margin : 0 ;

width : 100% ;

box-sizing : border-box /* to include padding and border in the width */

display : inline-block ;

/* nav a : hover {

background-color : green ;

color : white ; /* optional */

/* .active {

background-color : black ;

color : white ;

/* .content {

padding : 10px ;

}

</style>

</HEAD>

<BODY>

<nav>

Home

News

Contact

About

</nav>

<div class="content">

Hello World!

Some text...

</div>

</BODY>

</HTML>

Q3]a) Valid number

<HTML>

<HEAD><TITLE> Valid number </TITLE></HEAD>

<BODY>

<form onsubmit="return validateNumber()">

<label for="number">Enter a number:</label>

<input type="text" id="number" name="number" required>

<button type="submit">Submit </button>

</form>

<script>

function ~~validateNumber~~ validateNumber() {

let number = document.getElementById("number").value

if (isNaN(number)) {

alert("You have entered an invalid number");

return false;

} else {

alert("You have entered a valid number");

return true;

}

</script>

</BODY></HTML>

Q3b) <HTML>

<HEAD>

<TITLE> Bootstrap Grid Example </title>

<link rel = "stylesheet" href = "https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">

</HEAD>

<BODY>

<div class = "container-fluid">

<div class = "row">

<div class = "col-sm-6">

<h2> ~~Java~~ Famous Cricketers </h2>

 Rohit Sharma

 Virat Kohli

 Sachin Tendulkar

 Yuvraj Singh

</div>

<div class = "col-sm-6">

<h2> Famous Scientists </h2>

 Charles Darwin

 Albert Einstein

 Cristiano Ronaldo

 Lionel Messi

</div>

</div></div>

</BODY>

</HTML>

Bootstrap Grid System:-

The bootstrap grid system is a responsive, front-end framework that facilitates the creation of flexible and consistent layouts in web development.

It is based on a 12 column grid layout allowing the developers to design responsive web design pages for various screen.

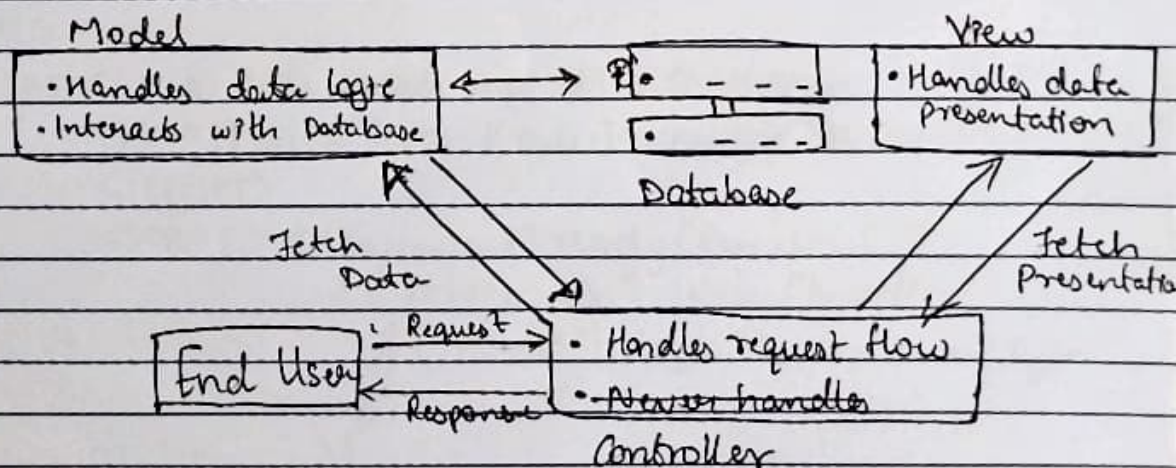
Responsive classes like 'col-sm', 'col-md' & 'col-lg' are used to control the column layout based on screen size.

Q7b) MVC Architecture:-

i) The Model-view-controller (MVC) framework is an architectural design pattern that separates an application into three main logical components: Model, View and Controller. Each architectural component is built to handle specific development aspect of an application.

ii) It isolates the business logic and presentation layer from each other. The main goal of this design pattern was to solve the problems of users controlling a large and complex data set by splitting a large application into specific sections that all have their own purpose.

iii)



iv) Controller :-

- Acts as an intermediary connecting views and model.
- Handles business logic and incoming requests.
- Instructs the model on what action to take.
- Processes data with model and interacts with the view for the final output.

v) View :-

- Manages UI logic and what users see on the screen.
- Generates the user interface based on data from the model.
- Only communication with controller, not directly with data.
- Displays information as instructed by the controller.

vi) Model :-

- Manages all data related logic.
- Handles data transfer between the view and controller.
- Interacts with the database for data retrieval or storage.
- Responds to the controller's request for data.

Q5a) <HTML>

```
<HEAD><TITLE> Hide/Unhide me </TITLE></HEAD>
<SCRIPT> src = "https://code.jquery.com/jquery-3.6.0-min.js"
</SCRIPT>
<SCRIPT> $(document).ready(function() {
    $("#hide-btn").click(function() {
        $("#txt").toggle();
    });
});
</SCRIPT>
</HEAD>
```


<BODY>

<button id = "hide-btn"> Hide/Unhide me </button>

<p id = "text"> I like web programming.

I like Database Management Systems

</p>

</BODY>

</HTML>

Q6]a) <?php

// connecting to database

\$conn = mysqli_connect("localhost", "root", "Test123", "mydb");

if (!~~co~~\$conn){

die("Error in establishing connection: ". mysqli_connect_error());

}

echo "Connection Successful established";
";

// creating myguests table

\$sql = "create table if not exists myguests (
id INT(6) UNSIGNED AUTO-INCREMENT PRIMARY KEY,
first-name VARCHAR (30) NOT NULL,
mobile-no VARCHAR (15) NOT NULL
);

if (my-mysqli-query(\$conn, \$sql)){

echo "Table myguest created successfully";

else {

echo("Error creating table: " . mysqli_error(\$conn);

}

// close connection.

mysqli_close(\$conn);

?>

optional }

Q66] <HTML>
<HEAD>
 <TITLE> Password Validation </title> </TITLE>
</HEAD>
<BODY>
 <SCRIPT>
 function checkpe checkPassword () {
 // prompt the user to enter a password
 var password = prompt("Please enter a password");

 // display message - Validate result.
 if (isValid) {
 alert("Password contains valid character.");
 } else {
 alert("Invalid Password");
 }
 }

 window.onload = checkPassword;
 </SCRIPT>
</BODY>
</HTML>