Code No.: 20-CS-PC-323 **R20** H.T.No. | | | R | 0 | | | | | | |

CMR INSTITUTE OF TECHNOLOGY: HYDERABAD

UGC AUTONOMOUS

III– B.Tech. – II – Semester End Examinations (Supply)– December– 2024

FULL STACK WEB DEVELOPMENT

(COMPUTER SCIENCE AND ENGINEERING)

[Time: 3 Hours] **[Max. Marks: 70]**

Note: 1. This question paper contains two parts A and B.
2. Part A is compulsory which carries 20 marks. Answer all questions in Part A.
3. Part B consists of 5 Units. Answer any one full question from each unit. Each question carries 10 marks and may have i, ii, iii as sub questions.
4. Illustrate your answers with NEAT sketches wherever necessary.

PART-A 10 X 2M = 20 M

S.No	Question	Blooms Taxonomy Level	CO
1	List any four text formatting tags and their usage with examples	L2	1

2	The following is a web-page:
	<html>
	  <head> <title>JavaScript</title></head>
	   <body bgcolor="Blue">
	    <script language="JavaScript">
	     <!—document. Write(""<h1> hello world </h1>""); //-- >
	    </script>
	   </body>
	</html>
	When the above web page is loaded into a browser, will it display something on the browser or it displays error message?
		L1	1

3	What is filtering in JQuery?	L1	2
4	Define Loops.	L1	2
5	Define node server.	L1	3
6	What is routing?	L1	3
7	Discuss various ng-directives of AngularJS.	L1	4
8	Define Http module.	L1	4
9	How can you initialize a repository in Git?	L1	5
10	What are the benefits of using Git?	L2	5

PART-B 5 X 10M = 50 M

11.A	(i) Create a simple HTML page which illustrates the three types of list. Try adding a defines uses unordered lists to define terms (ii) Write the sample code to tag the Geo location?	L4	1
	OR		
11.B	(i) Illustrate between <span> and <div> tags with suitable example. (ii)What are the different types of cascading style styles explain with examples in detail	L4	1
12.A	Explain the Arrays and String Processing using example.	L2	2
	OR		
12.B	List all JQuery event methods in detail with example.	L2	2
13.A	Explain about the MongoDB driver.	L2	3
	OR		
13.B	Describe the templet engine in Node.js.	L2	3
14.A	List the controllers in Angular and write the properties.	L3	4
	OR		
14.B	Create a UI for form validation. Explain in detail using steps.	L3	4
15.A	How to get start with GIT and work with Local repository. Write in detail?	L2	5
	OR		
15.B	Compare Git fetch and pull.	L2	5

----***----

CMR INSTITUTE OF TECHNOLOGY

(Approved by AICTE, Permanently Affiliated to JNTUH, Hyderabad, Accredited by NBA and NAAC with 'A' Grade)
Kandlakoya, Medchal Road, Hydarabad-501401
UGC Autonomous

COURSE: III - B.Tech / M.Tech / MBA II SEM

Date of Examination : 06-01-2025
Regulation : R20
Regular / Supply : Supply
Branch : Computer Science AND Engineering
Subject Name : Full Stack Web Development
Subject Code : 20-CS-PC-323.

Part A

1. List any four text formatting tags and their usage with example.

A. HTML contains Several elements for defining text with a special meaning. formatting elements were designed to display special types of text.

<b>, <strong>, <i>, <small> <mark> <sub> <sup>, <ins> <em>

HTML <b> and <strong> element & <i> and <small> element

The HTML <b> element defines bold text, without any extra importance.

Example: <b> This text is bold </b>

The html <strong> element defines text with strong importance.

Example: <strong> This text is strong </~~strong~~>

The HTML <i> element defines text displayed in italic.

Example: <i> This text is italic. </i>

The HTML <small> element defines smaller text.

Example: <small> This is smaller text. </small>.

5. Define node server.

A. Node is an open source server environment which is free and runs on various platforms such as Windows, Linux, Unix, Mac Os etc.. It uses javascript on the server. It can generate dynamic page content using create(), open(), read(), write(), delete() and close() files on the Server. It can collect form data, and add, delete, modify data in the database.

6. What is routing?

A. Routing defines how the client requests are handled by the application endpoints. Routing in Node refers to the process of determing how an application responds to client requests to different endpoints. In a web application, these endpoints typically correspond to different pages or functionalities within the application.

7. Discuss various ng-directives of Angular JS.

A. Angular JS lets you extend HTML with new attributes called directives. Angular JS has a set of built-in directives which offers functionality to your applications.

ng-app : directive initializes an Angular JS application.

ng-init : directive initializes application data.

ng-model : directive binds the value of HTML controls (input, select, textarea) to application data.

ng-repeat : directive repeats an HTML element.

8. Define Http Module.

A. The Http module provides functionality to create and manage HTTP servers and clients. It includes methods to handle incoming requests, send responses and interact with HTTP headers and status codes.

9. How can you initialize a repository in Git?

A. In Git local software to initialize a repository use a command init.

Useage: git init.

Output:

Three types of List

1. Ordered List

   1. first
   2. Second
   3. Third

2. Unordered list

   - Item 1
   - Item 2
   - Item 3

3. Definition List

   HTML : Hypertent Markup Language.

   CSS : Cascading Style sheets.

   XML : Entensionable Markup language.

ii, Write the Sample code to tag the Geo-location?

```
<!DOCTYPE html>
<html>
<body>
<h1> HTML Geolocation </h1>
<P> Click the button </P>
<button onClick = "getLocation()" > click here </button>
<p id ="demo" > </p>
<Script>
 const x = document. getElementById ("demo");
 function getLocation() {
    if (navigator. geolocation) {
       navigator. geolocation. getCurrentPosition ( showPosition);
    } else {
       x. innerHTML = "Geolocation is not supported";
    }
 }
```

Javascript using the class or id attribute.

<div> is a block-level element and <span> is an inline element.

Example: <h1> this is

< span> span element </span>

in HTML. </h1>

Output:

This is span element in HTML.

12.A. ~~Explain~~

(ii) What are the different types of Cascading style sheets. explain with examples in detail.

A. CSS is used to style and layout of web pages, controlling the appearance of HTML elements. CSS targets HTML elements and applies style rules to dictate their appearance

Types:

1. Inline CSS

Inline css involves applying styles directly to individual HTML elements using the style attribute.

Example

```
<p style = "color : "blue ; font-size : 50px ; font-family : Times New Roman ;
   >Inline css   </P>.
```

Output:

Inline CSS.

2. Internal (or) Embedded CSS

Internal css involves applying styles on individual HTML elements using the style tag. ~~and~~ the coresponding html element.

```
<style>
 P {
   Color : "blue";
   font-size : 50px;
   font-family : Times New Roman.
 } </style>
```

12 A. Explain the Arrays and String processing using example.

A. An array is a special variable, which can hold more than one value. An array can hold many values under a single name, and you can access the values by referring to an index number.

Creating an Array

Using an array literal is the easiest way to create a javascript array.

Syntax:

```
const arr_name = [item1, item2, ... item n];
```

Example:

```
const names = ["ram", "sita", "sai"];
```

Using Keyword new.

~~Syntax~~:

```
const names = new Array("ram", "sita", "sai");
```

Strings

Strings are for storing text and written with quotes.

Example:

```
let name = "CSM";

      (or)

let name = 'csm';
```

String length.

To find the length of a string use the built-in length property,

```
let alpha = "ABCD...XYZ";
let length = alpha.length;
```

Output

```
length = 26.
```

12 B. List all JQuery event methods in detail with enample.

A. All the different visitors actions that a web page can respond to are called events.

Mouse Events	Keyboard Events	form Events	Document/windows Events.
click	Keypress	Submit	load
dbclick	Keydown	change	resize
mouseenter	Keyup	focus	Scroll
mouseleave		blur	Unload

Mouse Events

click()method attaches an event handler function to an HTML element

```
En: $("p").click(function(){
        $(this).hide();
    });
```

dbclick(): method attaches an event handler function to an HTML element.

```
En: $("p").dbclick(function(){
        $(this).hide();
    });
```

mouseenter: method attaches an event handler function to an HTML element.

```
En: $("p").mouseenter(function(){
        alert("it's mouseenter event");
    });
```

mouseleave : method attaches an event handler function to an HTML element.

```
En: $("p").mouseleave(function(){
        alert("Bye!!!");
    });
```

Submit: Attempting to submit a form, bind to form element only

Example: $("form"). ~~focus~~ submit (function() {

});

Windows Events

load: to load the web page in the browser.

Example: $(" h1"). load (function(){

});

resize: To increase or decrease the size of the ~~maximize~~ window.

Example: $("h1").resize(function(){

});

Scroll: To move the window up or down.

Example: $("h1"). scroll (function() {

});

Unload: To unload the window.

Example: $("h1"). unload ('function() {

});

13.A. Explain about the MongoDB driver.

MongoDB Drivers

* It is a library designed to allow developers to interact with mongodb databases using the programming language of their choice.

* Each driver provides a set of APIs that are specific to the language, ensuring that developers can work with mongodb in a way that feels natural within the content of their language's ecosystem.

Key functions

1. Connection Management: Drivers manage connections to the mongodb server, handling things like connection pooling & reconnections.

Pug template Engine

Use pug template engine in node.js application using Express.js.

Install Pug

```
npm install pug --save.
```

* Pug template must be written inside .pug file and all .pug files must be put inside views folder in the root folder of Node.js application.

Example

index.pug

```
doctype html
html
   head
      title Simple Pug
   body
      h1 This is pug template engine.
       P lorem
```

file: server.js

```
var express =require ('express');
Var app = express ( );
 // set view engine.
  app.set ("view engine", "pug")
  app.get ('/', function (req, res) {
      res. render ( 'view.pug', index);
       res.render ('index');

    });
```

```
<div ng-app="myApp" ng-controller="myCtrl">
  First Name: <input type="text" ng-model="firstName"><br>
  Last Name: <input type="text" ng-model="lastName"><br>
  full Name: {{ firstName + "  " + lastName }}
 </div>
 <Script>
  Var app=angular.module('myApp',[]);
   app.controller('myctrl', function($scope) {
       $scope.firstName="CMRIT";
       $scope.LastName="College";
     });
   </script>
```

14B. Create a UI for form validation. Explain in detail using steps.

A.
```
<!DOCTYPE html>
 <html>
 <head>
  <title> Angular JS form Validation </title>
  <Script Src = "https://ajax.googleapis.com/... angular.min.js">
  </script>
  </head>
  <body ng-app="">
   <h1> CMRIT college </h1>
   <h3> Registration form Validation </h3>
   <form name="f1">
   <p> Name :
    <input name="username" ng-model="Username" required>
    <span ng-show =
```

15 A. How to get start with GIT and work with local repository. Write in detail?

A. To start with Git and working with a local repository involves Several Steps from installation to basic commands.

Step 1: Install Git.

- Go to official git website and download the version suitable for operating system.
- Follow the installation instructions in operating system.
- After installation, open a terminal and run the following command to verify installation.

```
-> git --version.
```

Step 2: Configure Git.

- Set username.

```
git config --global. user.name "name"
```

- Set email.

```
git config --global user.email "email@gmail.com"
```

- To check configuration.

```
git config --List.
```

Step 3: Create a local repository.

- Navigate to the project directory. using cd command

```
cd path/folder/project.
```

- Initialize a git repository.

```
git init.
```

- Cheack Status.

```
git status
```

15 B. Compare Git fetch and pull.

* git fetch and git pull are both commands used to update local repository with changes from a remote repository, but they serve different purposes and operate in different ways.

→ Git fetch.

* Purpose: It retrieves updates from a remote repository but does not merge them into your local branch.
* It updates remote-tracking branches with the latest changes from the remote repository.
* fetch changes from the remote repository.

```
git fetch origin.
```

* Chech status

```
git status
```

* View changes

```
git log
```

* Merge changes manually

```
git merge origin/main.
```

→ Git Pull

* Purpose: It is a combination of 'git fetch' followed by 'git merge'
* It retrieves updates from the remote repository and automatically merges them into current branch.
* To quickly update current branch with the latest changes from the remote repository without reviewing them first.
* Pull changes from the remote repository

```
git pull origin main.
```