

**ATSS’s Institute of Industrial and Computer Management and Research, Nigdi Pune**

**MCA Department Academic Year: 2022-23**

**Practical Journal on Advanced Internet Technology**

**(SEM- II): IT 21 L**

Submitted By: Roll no – 29 Name – Nitin

Sanjay Bhopale Date :

**2**

**Course Outcomes:**

Student will be able to

CO1: Implement python programming concepts for solving real life problems. (Apply) CO2: Implement Advanced Internet Technologies (Apply)

**ATSS’s**

**Institute of Industrial and Computer Management and Research, Nigdi Pune MCA Department**

**INDEX**

**Students Name: Roll No.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr. No** | **Program Title** | **Course Outcome** | **Page No.** | **Teacher’s Sign with**  **Date** | **Remarks** |
| **1.** | WAP to change font size of a text as per the “range” control value, ranging from 8 to 50 | **CO2** |  |  |  |
| **2.** | Write a program to embed audio and video  elements use controls. | **CO2** |  |  |  |
| **3.** | WAP to draw a circle considering mouse click  position as a centre.(Radius can be considered as 50) | **CO2** |  |  |  |
| **4.** | Write a program to draw the rectangle shape  using SVG and apply animation to it. | **CO2** |  |  |  |
| **5.** | Apply pseudo selectors as per instructions to set the background color for visited and unvisited links to ‘lightblue’, and the background color for the hover and active link  states to 'green’. | **CO2** |  |  |  |
| **6.** | Consider the heart shape as an image and apply Animation on hover effect where image will increase the size and revert back to original  size with shadow effect. | **CO2** |  |  |  |
| **7.** | Apply transition and transformation to perform following task:   1. On hover of an image, it should rotate in 360 degrees. 2. On hover of a div, text should increase the size and removing cursor from div, size of   the text should be restored. | **CO2** |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **8.** | Write a PHP script to generate simple random password of 7 characters from a given string. Assume a string contains characters from A- Z, a-z, 0- 9. Generated password must contain at least 1 number, 1 capital letter and  1 small letter. | **CO2** |  |  |  |
| **9.** | Write a PHP program to print Fibonacci Series using both ways: loops, recursive function | **CO2** |  |  |  |
| **10.** | Create a table Emp with fileds: empno, name, address, salary, deptno. Design HTML & PHP pages for following: a. Insert a record b. Delete record c. View record d.  Display department number wise group  report. | **CO2** |  |  |  |
| **11.** | Create book shopping cart application in PHP. Display all books details on page using Book  table. | **CO2** |  |  |  |
| **12.** | Program to show current date and time using user defined module. | **CO2** |  |  |  |
| **13.** | Program using built-in modules to split the query string into readable parts. | **CO2** |  |  |  |
| **14.** | Program using NPM which will convert entered string into either case. | **CO2** |  |  |  |
| **15.** | Create angular project which will demonstrate the usage of component directive, structural  directive and attribute directives. | **CO2** |  |  |  |
| **16.** | Create angular project which has HTML template and handle the click event on click of  the button. | **CO2** |  |  |  |
| **17.** | Write Program for Form validation in Angular. | **CO2** |  |  |  |

# WAP to change font size of a text as per the “range” control value, ranging from 8 to 50.

Code:

<!DOCTYPE html>

<html lang="en">

<head>

<title>FontRange</title>

</head>

<body>

<p id="demo">Hello Good Morning</p>

<input type="range" min="8" max="50" id="range" onchange="fun()">

<script>

function fun(){

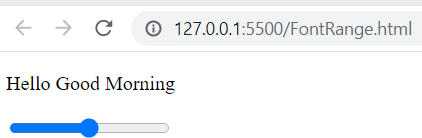
var demo = document.getElementById("demo");

var range = document.getElementById("range").value; demo.style.fontSize = range+"px";

}

</script>

</body>

</html> Output:

# Write a program to embed audio and video elements use controls.

Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Question-2</title>

</head>

<body>

<audio controls loop muted autoplay >

<source src="kabirsingh.mp3" type="audio/mp3">

</audio> <br>

<video width="320" height="240" controls>

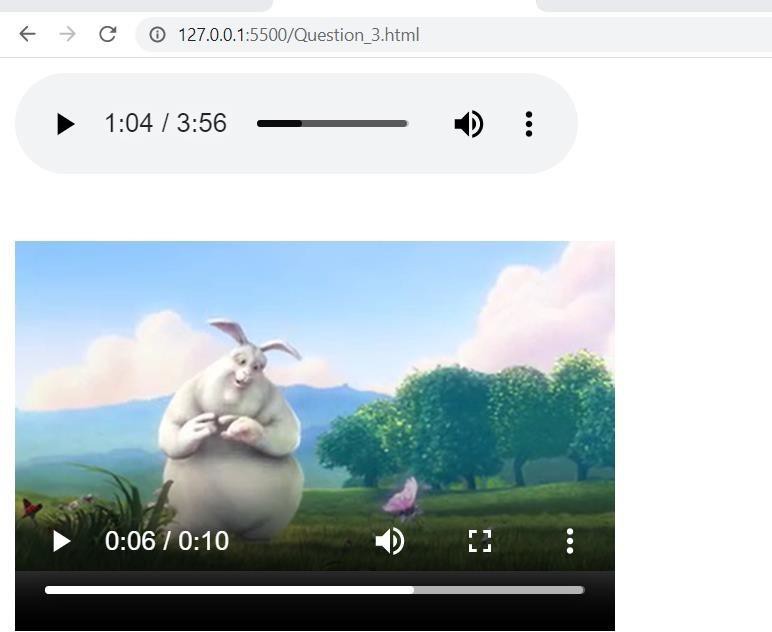
<source src="mov\_bbb.mp4" type="video/mp4">

</video>

</body>

</html>

Output:



# WAP to draw a circle considering mouse click position as a centre.(Radius can be considered as 50).

Code:

<!DOCTYPEhtml>

<html>

<head>

<style>#myCanvas{ border: 1pxsolidblack;

}

</style>

<title>Question-3</title>

</head>

<body>

<canvasid="myCanvas"width="500"height="500"></canv as>

<script>var canvas = document.getElementById("myCanvas"); var ctx = canvas.getContext("2d");

canvas.addEventListener("click", function(event) { var rect = canvas.getBoundingClientRect(); var mouseX = event.clientX - rect.left;

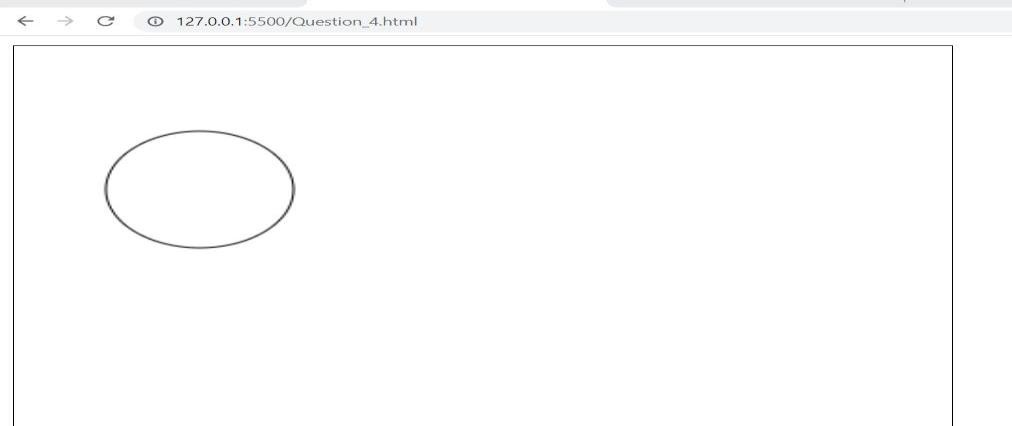
var mouseY = event.clientY - rect.top; ctx.clearRect(0, 0, canvas.width, canvas.height); ctx.beginPath();

ctx.arc(mouseX, mouseY, 50, 0, 2 \* Math.PI); ctx.stroke();

});

</script>

</body>

</html> Output:

# Write a program to draw the rectangle shape using SVG and apply animation to it.

Code:

<!DOCTYPEhtml>

<htmllang="en">

<head>

<metacharset="UTF-8">

<metahttp-equiv="X-UA-Compatible"content="IE=edge">

<metaname="viewport"content="width=device-width, initial-scale=1.0">

<title>Question-6</title>

</head>

<body>

<svgwidth="500"height="500">

<rectid="myRect"x="50"y="50"width="200"height="100"fill="blu e">

<animateattributeName="width"from="200"to="400"dur="2s"repea tCount="indefinite"/>

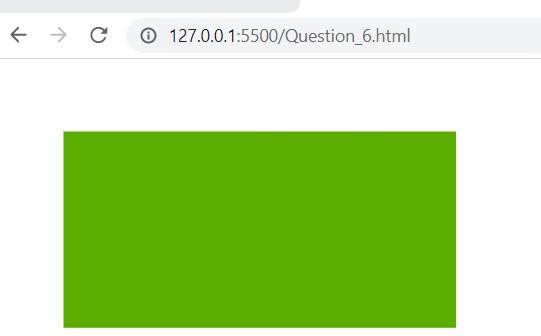
<animateattributeName="height"from="100"to="200"dur="2s"repe atCount="indefinite"/>

<animateattributeName="fill"values="blue; red; green; yellow"dur="4s"repeatCount="indefinite"/>

</rect>

</svg>

</body>

</html> Output:

# Apply pseudo selectors as per instructions to set the background color for visited and unvisited links to ‘lightblue’, and the background color for the hover and active

**link states to 'green’.**

Code:

<!DOCTYPEhtml>

<html>

<head>

<title>Question-8</title>

<style>

a:link { background-color: lightblue;

}

a:visited { background-color: lightblue;

}

a:hover { background-color: green;

}

a:active { background-color: green;

}

</style>

</head>

<body>

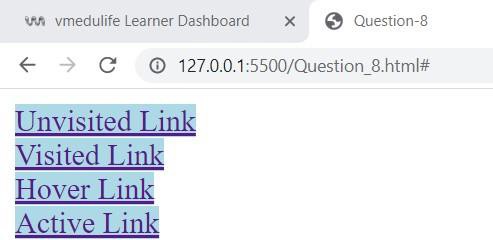
<a href="#">Unvisited Link</a><br>

<a href="#">Visited Link</a><br>

<a href="#">Hover Link</a><br>

<a href="#">Active Link</a>

</body>

</html> Output:

# Consider the heart shape as an image and apply Animation on hover effect where image will increase the size and revert back to original size with shadow effect.

Code:

<!DOCTYPEhtml>

<html>

<head>

<title>Question-11</title>

<style> .heart { width: 100px; height:60px; background- image: url("./heart2.jpg"); background-size:contain; transition: transform 0.3s, box-shadow 0.3s;

}

.heart:hover { transform: scale(1.2); box-shadow: 0010pxrgba(0, 0, 0, 0.3);

}

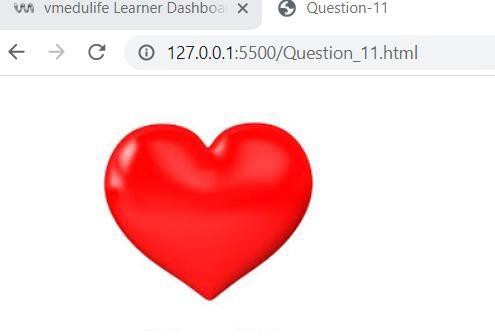
</style>

</head>

<body>

<divclass="heart"></div>

</body>

</html> Output:

* 1. **Apply transition and transformation to perform following task:**
     1. **On hover of an image, it should rotate in 360 degrees.**
     2. **ii. On hover of a div, text should increase the size and removing cursor from div, size of the text should be restored.**

Code:

<!DOCTYPEhtml>

<html>

<head>

<style>

.rotate-image { width: 100px; height: 100px;

transition: transform 0.3s;

}

.rotate-image:hover { transform: rotate(360deg);

}

.enlarge-text { display: inline-block; transition: font-size 0.3s; cursor: pointer;

}

.enlarge-text:hover { font-size: 1.2em; cursor: default;

}

</style>

<title>Question-12</title>

</head>

<body>

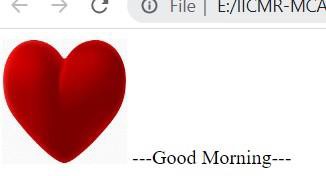
<imgclass="rotate-image"src="./heart.jpg"alt="Image">

<divclass="enlarge-text">

<p>---Good Morning---</p>

</div>

</body>

</html> Output:

# Write a PHP script to generate simple random password of 7 characters from a given string. Assume a string contains characters from A-Z, a-z, 0- 9. Generated password must contain at least 1 number, 1 capital letter and 1 small letter.

Code:

<?php

Function generateRandomPassword($length) {

$characters = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456 789';

$password = '';

$charLength = strlen($characters); for ($i = 0; $i< $length;

$i++) {

$randomIndex = mt\_rand(0, $charLength - 1);

$password .= $characters[$randomIndex];

} return $password;

}

$randomPassword = generateRandomPassword(10); echo "Random Password: " . $randomPassword;

?>

Output:

# Write a PHP program to print Fibonacci Series using both ways: loops, recursive function.

Code:

<?php function fibonacciLoop($nl) {

$fib = [];

$fib[0] = 0;

$fib[1] = 1;

for ($i = 2; $i< $nl; $i++) {

$fib[$i] = $fib[$i - 1] + $fib[$i - 2];

}

return $fib;

}

$nl = 10; // Number of Fibonacci numbers to generate

$fibonacciSeries = fibonacciLoop($nl);

echo "Fibonacci Series using Loops: "; foreach ($fibonacciSeries as $number) { echo $number ." ";

} echo "<br>"; echo "Fibonacci Series Using Recursion:"; functionfiboncacciRecursive($a,$b,$nr)

{

if($nr==0)

{ return; } $c=$a+$b;

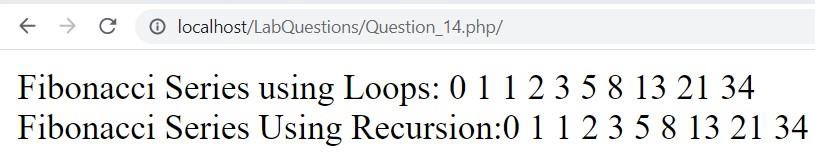
echo $c." "; $nr=$nr-1; fiboncacciRecursive($b,$c,$nr);

}

$nr = 10; //Number of Fibonacci numbers to generate $a=0; $b=1; echo $a." ".$b." ";

$nr=$nr-2; fiboncacciRecursive($a,$b,$nr);

?>

Output:

* 1. **Create a table Emp with fileds: empno, name, address, salary, deptno. Design HTML & PHP pages for following:**

1. **Insert a record**
2. **Delete record**
3. **View record**
4. **Display department number wise group report.**

Code:

Display.php

<?php

$servername='localhost';

$username='root'; $password='';

$dbname ='nitin';

$conn = mysqli\_connect($servername,$username,$password,$dbname); if(!$conn) { die('Could not connect mysql'.mysql\_error());

} if(isset($\_POST['display']))

{

$sql = "select \* from emp"; $result

=mysqli\_query($conn,$sql); if(mysqli\_num\_rows($result)>0){

?>

<table border=2>

<tr>

<th>EmpNo</th>

<th>Name</th>

<th>Address</th>

<th>Salary</th>

<th>DeptNo</th>

</tr>

<?php while($row=mysqli\_fetch\_assoc($result)){

echo

"<tr><td>".$row["empno"]."</td><td>".$row["name"]."</td><td> ".$row["address"]."</td><td>".$row["salary"]."</td><td>".$ro w["deptno"]."</td></tr>";

} } else { echo "No Record Found";

}

//mysqli\_close($conn);

} if(isset($\_POST['insert']))

{

$empno=$\_POST["id"];

$empname=$\_POST["name"]; $addr=$\_POST["address"];

$sal=$\_POST["salary"];

$deptno=$\_POST["deptno"];

$sql1="insert into emp values($empno,'$empname','$addr',$sal,'$deptno')"; if(mysqli\_query($conn,$sql1))

{

$sql = "select \* from emp"; $result

=mysqli\_query($conn,$sql); if(mysqli\_num\_rows($result)>0){

?>

<table border=2>

<tr>

<th>EmpNo</th>

<th>Name</th>

<th>Address</th>

<th>Salary</th>

<th>DeptNo</th>

</tr>

<?php while($row=mysqli\_fetch\_assoc($result)){

echo

"<tr><td>".$row["empno"]."</td><td>".$row["name"]."</td><td> ".$row["address"]."</td><td>".$row["salary"]."</td><td>".$ro w["deptno"]."</td></tr>";

} } else { echo "No Record Found";

} } else { echo"Error:".mysqli\_error($conn); }

}

//mysqli\_close($conn); if(isset($\_POST['Del']))

{

$no=$\_POST['no'];

$sql2="delete from emp where empno=$no"; if(mysqli\_query($conn,$sql2))

{

$sql = "select \* from emp"; $result

=mysqli\_query($conn,$sql); if(mysqli\_num\_rows($result)>0){

?>

<table border=2>

<tr>

<th>EmpNo</th>

<th>Name</th>

<th>Address</th>

<th>Salary</th>

<th>DeptNo</th>

echo

</tr>

<?php while($row=mysqli\_fetch\_assoc($result)){

"<tr><td>".$row["empno"]."</td><td>".$row["name"]."</td><td> ".$row["address"]."</td><td>".$row["salary"]."</td><td>".$ro w["deptno"]."</td></tr>";

} } else { echo "No Record Found";

} } else { echo"Error:".mysqli\_error($conn); }

}

//mysqli\_close($conn); if(isset($\_POST['display']))

{

$sql = "select \* from emp"; $result

=mysqli\_query($conn,$sql); if(mysqli\_num\_rows($result)>0){

?>

<table border=2>

<tr>

<th>EmpNo</th>

<th>Name</th>

<th>Address</th>

<th>Salary</th>

<th>DeptNo</th>

</tr>

<?php while($row=mysqli\_fetch\_assoc($result)){

echo

"<tr><td>".$row["empno"]."</td><td>".$row["name"]."</td><td> ".$row["address"]."</td><td>".$row["salary"]."</td><td>".$ro w["deptno"]."</td></tr>";

} } else { echo "No Record Found";

}

//mysqli\_close($conn);

}

?>

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>PHP Crud Operation</title>

<link rel="stylesheet"

href="[https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dis](https://cdn.jsdelivr.net/npm/bootstrap%405.2.3/dis) t/css/bootstrap.min.css"> <link rel="stylesheet" href="[https://cdn.jsdelivr.net/npm/bootstrapicons@1.10.2/font/bootstrap-icons.css"](https://cdn.jsdelivr.net/npm/bootstrapicons%401.10.2/font/bootstrap-icons.css)

</head>

<body>

<div class="m-5">

<button class="btn btn-primary">New Record</button>

<div class="d-flex justify-content-center">

<div>

<h1>Student Information</h1>

<table>

<thead>

<tr>

<th>Name</th>

<th>Mobile</th>

<th>Email</th>

<th>Password</th>

<th>Operation</th>

</tr>

</thead> <tbody>

<tr>

<td></td>

</tr>

</tbody>

</table>

</div>

</div>

</div>

<form method="post" action="Display.php"> Enter id <input type="number" name="id"><br> enter name<input type="text" name="name"><br> enter address<input type="text" name="address"><br> enter salary<input type="number" name="salary"><br> enter deptNo<input type="number" name="deptno"><br>

<button name="display" value="display">Display</button>

<input type="submit" value="Insert" name="insert">

<button name="Delete" value="Delete"><a href="./delete.html">Delete</a></button>

<input type="submit" name="View Dept Wise" value="display2">

</form>

</body>

</html>

Delete.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Delete</title>

</head>

<body>

<h2>Insert the Employee N0/h2>

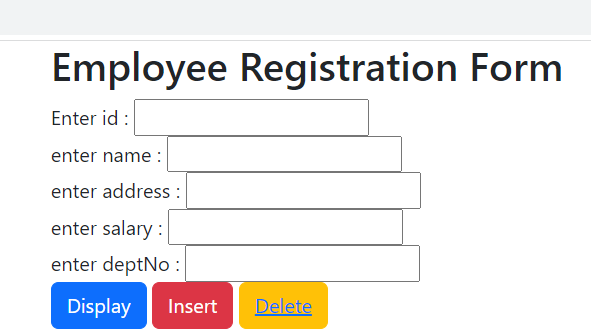
<form action="Display.php" method="post">

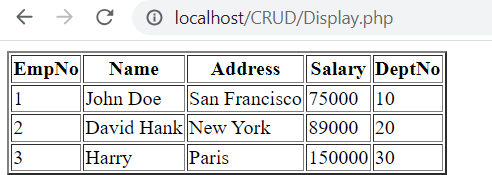
<input type="number" name="no">

<button name="Del">Delete</button>

</form>

</body>

</html> Output:



# Create book shopping cart application in PHP. Display all books details on page using Book table.

Code:

HTML:

<html>

<form method="post" action="main2.php">

<label>Welcome to E-Book store</label><br><br>

<input type="submit" name="show" value="Show Books">

</form>

</html>

PHP:

<?php

$servername="localhost";

$username="root";

$password="";

$dbname="Books";

$conn=mysqli\_connect("$servername","$username","$password",$dbname); if(!$conn)

{

die("Error:".mysqli\_error($conn));

}

if(isset($\_POST['show']))

{

$sql = "select \* from bk";

$result =mysqli\_query($conn,$sql); if(mysqli\_num\_rows($result)>0){

?>

<table border=2>

<tr>

<th>BookNo</th>

<th>Name</th>

<th>Author</th>

<th>BookType</th>

</tr>

<?php while($row=mysqli\_fetch\_array($result)){

echo "<tr><td>".$row["BookNo"]."</td><td>".$row["Name"]."</td><td>".$row["Author"]."</td><td>".

$row["Type"]."</td></tr>";

}

}

else {

echo "No Record Found";

}

}

else

{

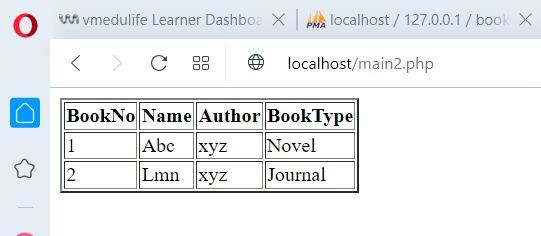
}

echo"Error:".mysqli\_error($conn);

mysqli\_close($conn);

?>

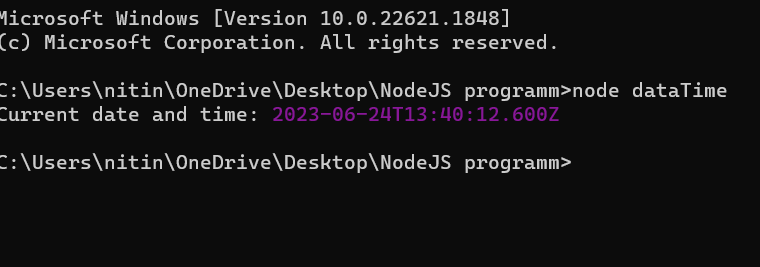
Output:



# Program to show current date and time using user defined module.

Code:

const currentDate = new Date(); console.log("Current date and time:", currentDate);

Output:

# Program using built-in modules to split the query string into readable parts.

Code:

const url = require('url');

// Sample query string

const queryString = 'https://portal.vmedulife.com//student/Assignments/AssignmentList.php';

// Parsing the query string using the url module const parsedUrl = new URL(queryString);

// Extracting readable parts from the parsed URL const protocol = parsedUrl.protocol;

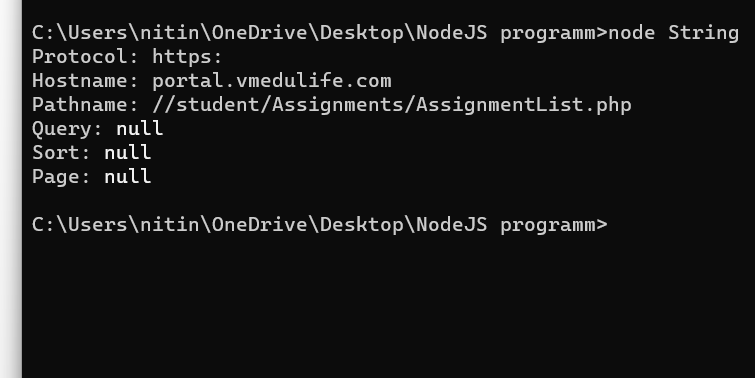
const hostname = parsedUrl.hostname; const pathname = parsedUrl.pathname;

const searchParams = parsedUrl.searchParams;

// Getting specific query parameters const query = searchParams.get('query'); const sort = searchParams.get('sort'); const page = searchParams.get('page');

// Outputting the results console.log('Protocol:', protocol); console.log('Hostname:', hostname); console.log('Pathname:', pathname); console.log('Query:', query); console.log('Sort:', sort);

console.log('Page:', page);

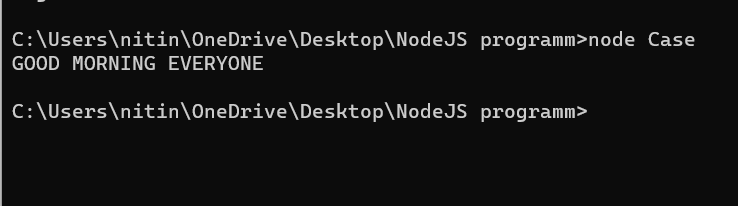
Output:

# Program using NPM which will convert entered string into either case.

Code:

var uc = require('upper-case'); console.log(uc.upperCase("Good Morning Everyone"));

Output:



# Create angular project which will demonstrate the usage of component directive, structural directive and attribute directives.

Code: Show.Component.html

<h1>Inside the Component Directive</h1>

<h2 appColor>Structural Directive</h2>

<ul \*ngFor="let value of Names">

<li>{{value}}</li>

</ul>

<div \*ngIf = 'myBool'>Boolean is set to true</div>

<div [ngSwitch]="num">

<div \*ngSwitchCase="'1'">One</div>

<div \*ngSwitchCase="'2'">Two</div>

<div \*ngSwitchCase="'3'">Three</div>

<div \*ngSwitchCase="'4'">Four</div>

<div \*ngSwitchCase="'5'">Five</div>

</div>

Show.component.ts

import { Component } from '@angular/core';

@Component({ selector: 'app-show',

templateUrl: './show.component.html', styleUrls: ['./show.component.css']

})

export class ShowComponent {

Names = ["John","Harry","David","Peter"]; myBool = true;

num = 4;

}

Color.directive.ts

import { Directive,ElementRef,HostListener } from '@angular/core';

@Directive({ selector: '[appColor]'

})

export class ColorDirective {

constructor(public eobj : ElementRef)

{

}

@HostListener('mouseenter')onmouseenter()

{

this.eobj.nativeElement.style.background = 'blue'

}

@HostListener('mouseleave')onmouseleave()

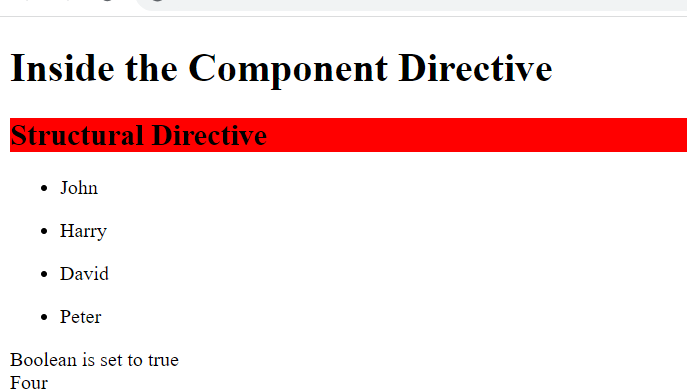
{

this.eobj.nativeElement.style.background = 'red'

}

}

Output:



# Create angular project which has HTML template and handle the click event on click of the button.

Code: App.component.html

<button (click)="Message()">Show</button>

App.component.ts

import { Component } from '@angular/core';

@Component({ selector: 'app-root',

templateUrl: './app.component.html', styleUrls: ['./app.component.css']

})

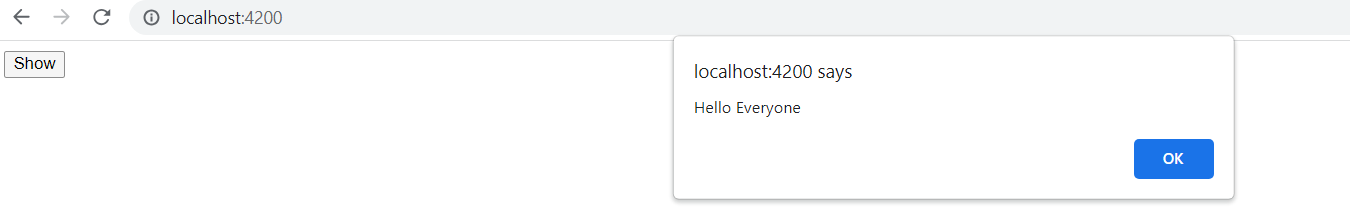
export class AppComponent { title = 'Assignment'; Message()

{

alert("Hello Everyone");

}

}

Output:

# Write Program for Form validation in Angular.

Code:

App.component.html

<div class = "container-fluid">

<h2>Marvellous Login Form</h2>

<!--Add FormGroup woth our model-->

<form [formGroup] = "MarvellousForm">

<div class="form-group">

<label>User Name </label>

<!--Use class binding for validation-->

<input [class.is-invalid] = "MarvellousForm.get('username')?.invalid" type="text" class="form-control" formControlName="username">

<small [class.d-none]="MarvellousForm.get('username')?.valid" class="text- danger">Username is required</small>

</div>

<div class="form-group">

<label>Password </label>

<input [class.is-invalid] = "MarvellousForm.get('passowrd')?.invalid && MarvellousForm.get('passowrd')?.touched" type="password" class="form-control" formControlName="passowrd">

<small [class.d-none]="MarvellousForm.get('passowrd')?.valid || MarvellousForm.get('passowrd')?.untouched" class="text-danger">Password is required</small>

</div>

<div class="form-group">

<label>Confirm Password </label>

<input type="password" class="form-control" formControlName="ConfirmPass">

</div>

<!--Add nested Form group-->

<div formGroupName = "MarvellousClass">

<div class = "form-group">

<label>Batch Name</label>

<input type="text" class = "form-control" formControlName="batch">

</div>

<div class = "form-group">

<label>Fees</label>

<input type="text" class = "form-control" formControlName="fees">

</div>

</div>

<button type="submit" class="btn btn-primary">Login</button>

<button type="button" class="btn btn-primary ml-3" (click)="SetData()">Set Values</button>

</form>

{{MarvellousForm.value | json}}

</div>

App.component.ts

import { Component } from '@angular/core';

// import classes which are required for reactive forms

import {FormBuilder,FormGroup,Validators, FormControl, MinLengthValidator} from '@angular/forms'

@Component({ selector: 'app-root',

templateUrl: './app.component.html', styleUrls: ['./app.component.css']

})

export class AppComponent

{

// Inject FormBuilder service constructor(public fbobj : FormBuilder)

{

}

MarvellousForm = this.fbobj.group(

{

// Add Multiple validations

username :['', [Validators.required, Validators.minLength(5)] ], passowrd : ['',Validators.required],

ConfirmPass : [''], MarvellousClass : this.fbobj.group(

{

batch :['Python'],

fees : ['5000']

}

)

}

);

// Method to set FormControl fields through program SetData()

{

this.MarvellousForm.setValue(

{

username : 'Piyush', passowrd : 'abcd', ConfirmPass : 'abcd', MarvellousClass :

{

batch : 'Python', fees : '5000'

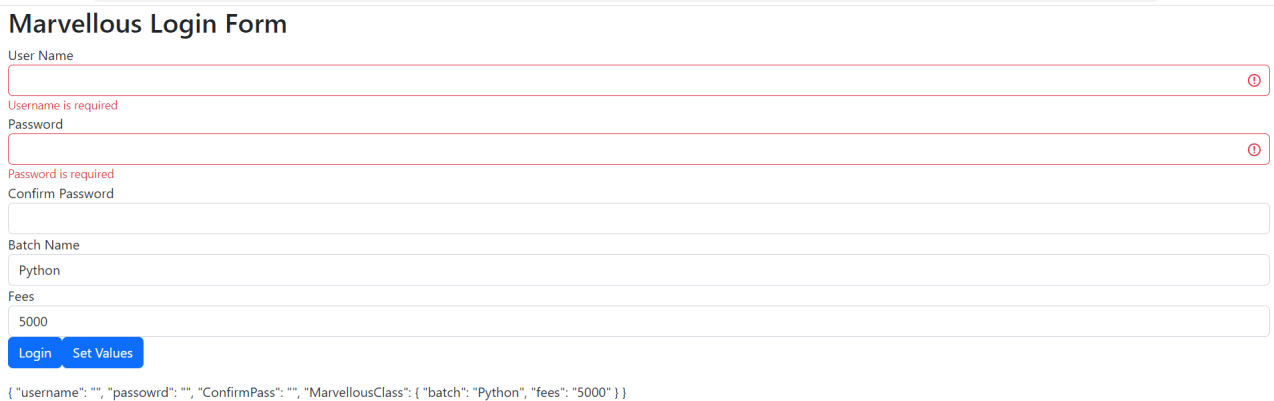
}

}

)

}

}

Output: