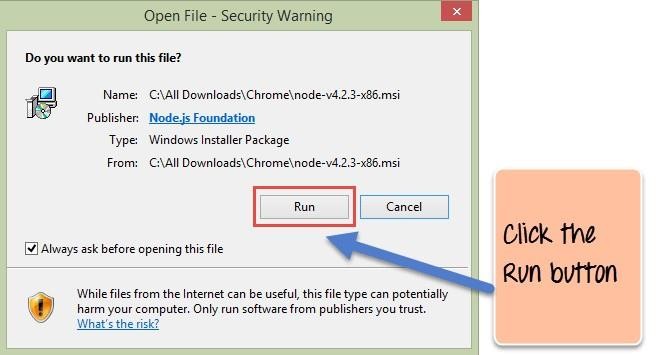
# Practical 1

**Steps to download node.js**

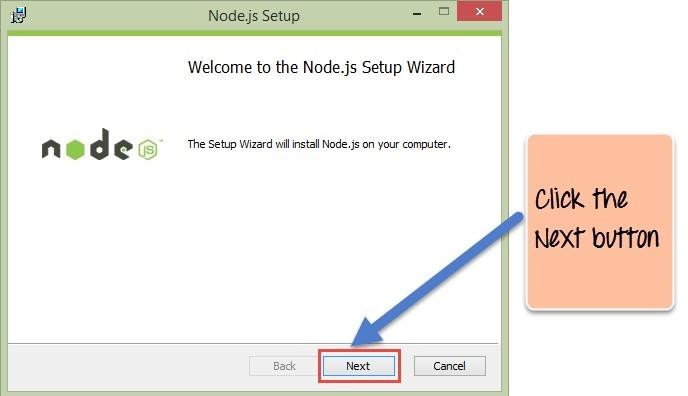
**Step 1)** Download Node.js Installer for Windows Go to the site

<https://nodejs.org/en/download/>and download the necessary binary files.

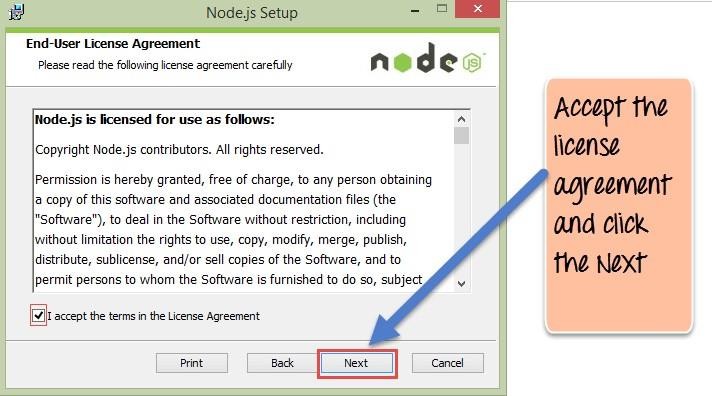
**Step 2)** Run the installation Double click on the downloaded .msi file to start the installation. Click the Run button on the first screen to begin the installation.

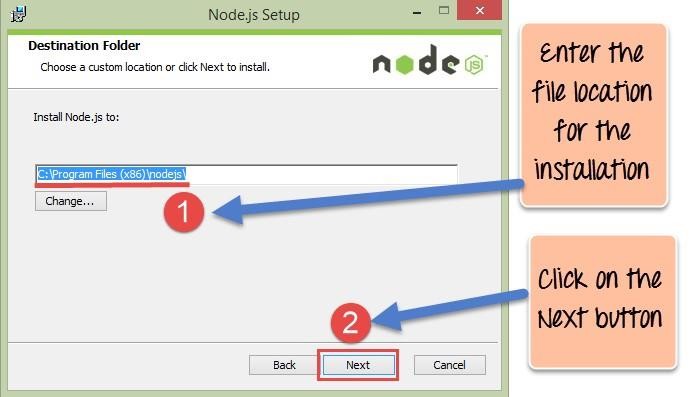


**Step 3)** Continue with the installation steps In the next screen, click the “Next” button to continue with the Node.js download and installation

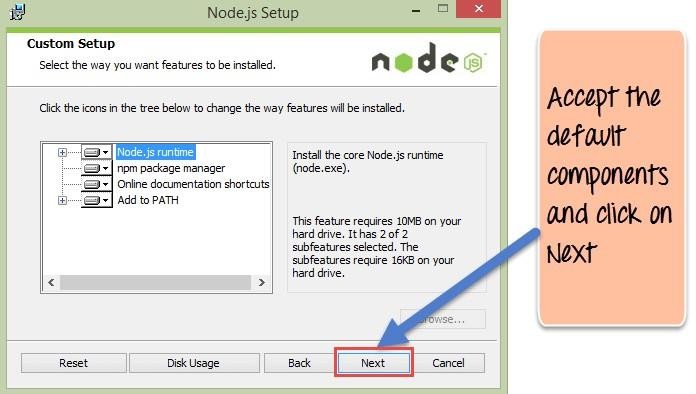


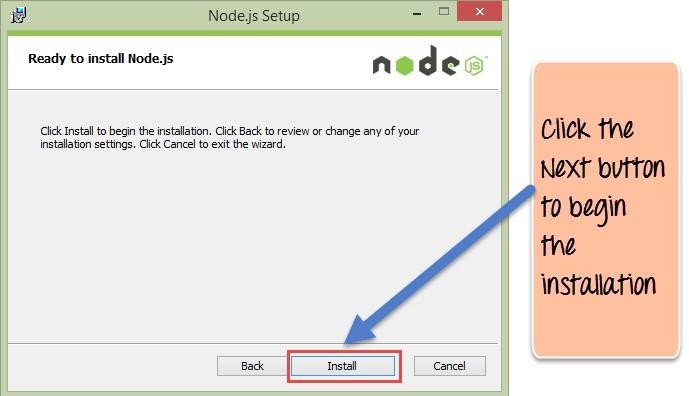
**Step 4)** Accept the terms and conditions In the next screen, Accept the license agreement and click on the Next button



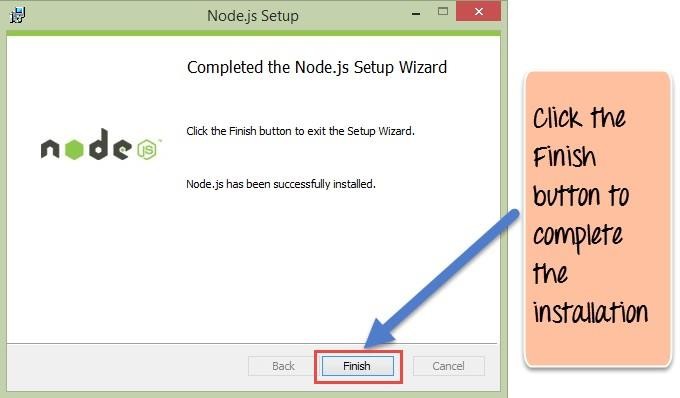
**Step 5)** Set up the path In the next screen, choose the location where Node.js needs to be installed and then click on the Next button.

**Step 6**) Select the default components to be installed Accept the default components and click on the Next button.

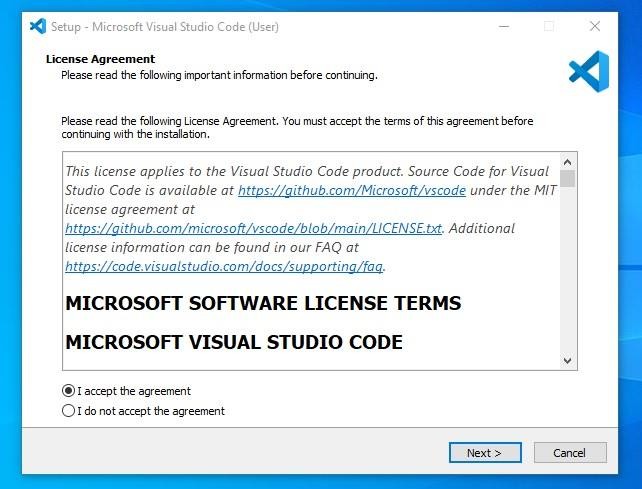
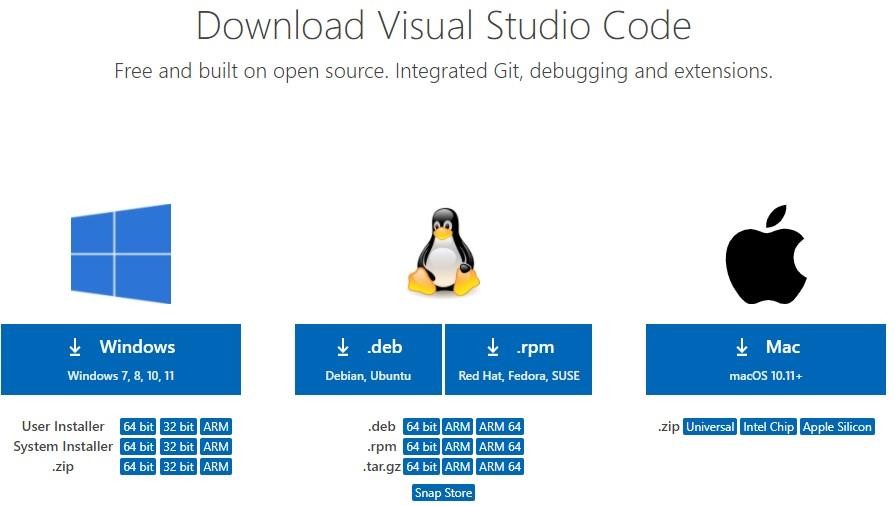


**Step 7)** Start the installation In the next screen, click the Node.js install button to start installing on Windows

**Step 8**) Complete the installation Click the Finish button to complete the installation. Complete the installation Click the Finish button to complete the installation.



**Practical 2**

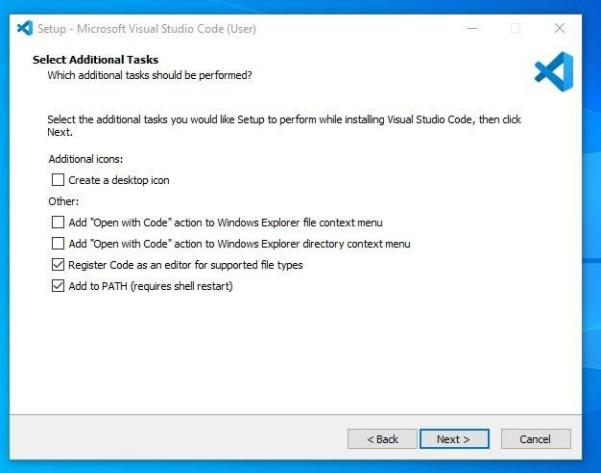


**Steps to download visual studio**

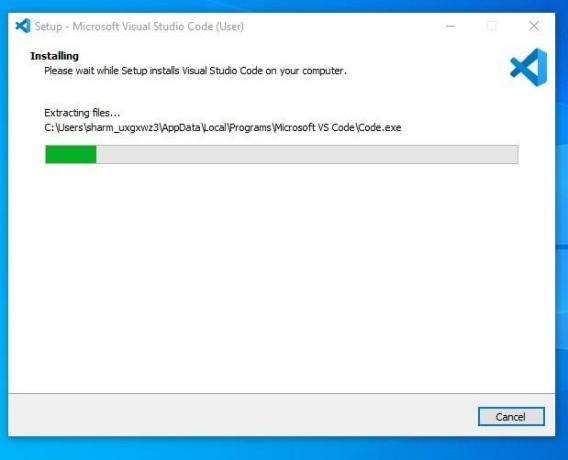
Step 1: **Visit the [official website](https://code.visualstudio.com/docs/?dv=win) of the** Visual Studio Code **using any web browser like Google Chrome, Microsoft Edge, etc. and Press the “**Download for Windows**” button**

Step 2:When the download finishes, then the Visual Studio Code icon appears in the downloads folder. Click on the installer icon to start the installation process of the Visual Studio Code. After the Installer opens, it will ask you for accepting the terms and conditions of the Visual Studio Code. Click on and then click the button.

Step 3: **Choose the location data for running the Visual Studio Code. It will then ask you for browsing the location. Then click on** Next **button.**



Step 4: **Then it will ask for beginning the installing setup. Click on the** Install **button. After clicking on Install, it will take about 1 minute to install the Visual Studio Code on your device.**



Step 5: **After the Installation setup for Visual Studio Code is finished, it will show a window like this below. Tick the “**Launch Visual Studio Code**” checkbox and then click** Next**.**



**Practical 3**

**Demonstrate the basic arithmetic operations in Node.js**

## function sum(a, b) {

## return a + b;

}

function sub(a, b) {

return a - b;

## }

## function mul(a, b) {

## return a \* b;

}

function div(a, b) {

return a / b;

## }

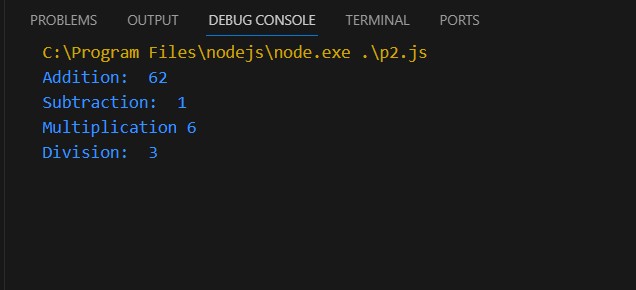
## console.log("Addition: ", sum(56, 6));

## console.log("Subtraction: ", sub(4, 3));

## console.log("Multiplication", mul(2, 3));

console.log("Division: ", div(6, 2));

# Output:



**Practical 4**

# To determine whether a given number is even or odd in Node.js

function displayresult(a)

{

console.log(a);

}

function check(num) {

let sum= num;

if (num % 2 == 0) {

console.log("Number is Even")

}

else

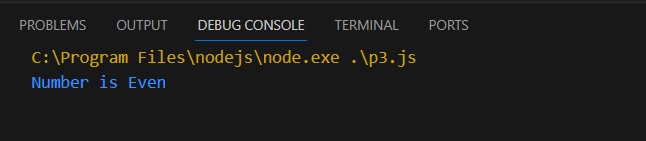
{

console.log("Number is odd")

} }

check(18,displayreslt)

# Output:



**Practical 5**

# To print all prime numbers up to a given number in Node.js

Function isPrime(n)

{

if(n==1||n==0)

return false;

for(var i=2;i<n;i++)

{

if(n%i==0)

return false;

}

return true;

}

var num =30;

for(vari=1;i<=num;i++){if(isPrime(i))

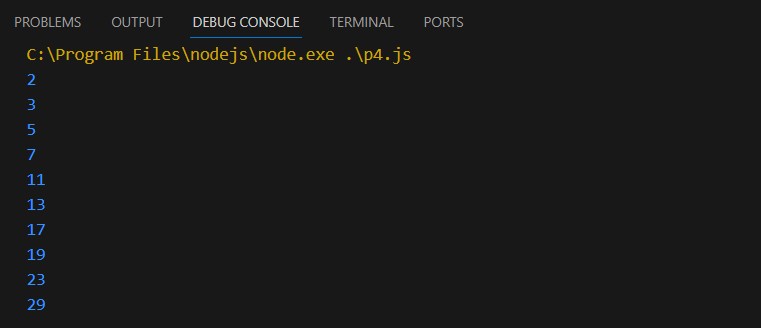
{

console.log(i);

}

}

# Output:



**Practical 6**

**Create an application in NodeJS to reverse the given number and display it (Note: 5 digit number)**

var number = 12345;

var reversedNumber = number.toString().split('').reverse().join('');

console.log('Reversed number is: ' + reversedNumber);

**Output:**



# Practical 7

**Create an application in Node.js to display Armstrong number 15**

function isArmstrongNumber(num) {

let sum = 0;

const strNum = String(num);

const len = strNum.length;

for (let i = 0; i < len; i++) {

sum += Math.pow(Number(strNum[i]), len);

}

return sum === num;

}

function printFirstNArmstrongNumbers(n) {

let count = 0;

let num = 1;

while (count < n) {

if (isArmstrongNumber(num)) {

console.log(num);

count++;

}

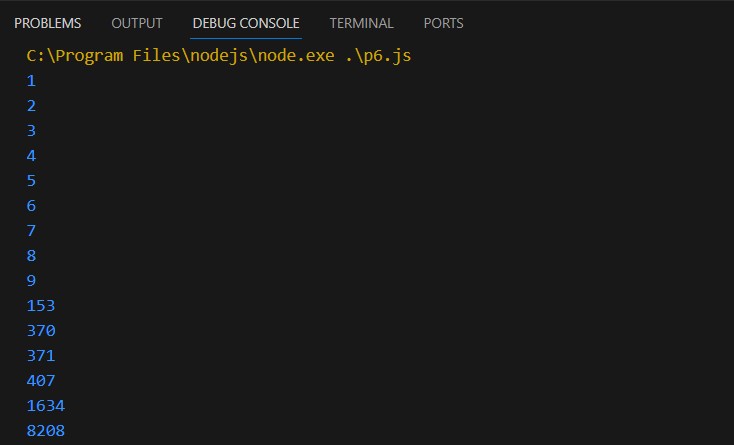
num++;

}

}

printFirstNArmstrongNumbers(15);

**Output:**



# Practical 8

**To generate the first 10 numbers in the Fibonacci sequence in Node.js**

var a=0;

var b=1;

var c;

console.log(a); console.log(b); for(i=0;i<8;i++)

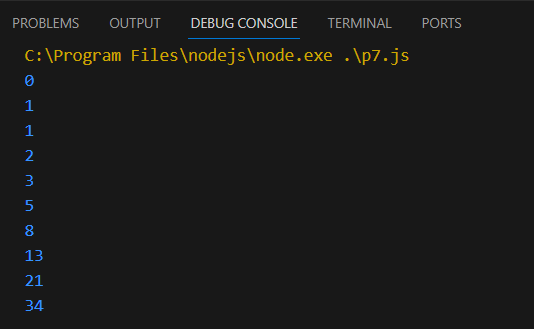
{

c=a+b; console.log(c); a=b;

b=c;

}

# Output:



**Practical 9**

# To demonstrate the use of setTimeout and arrow functions in Node.js

const message = function()

{

console.log("Hello NodeJS, Welcome");

}

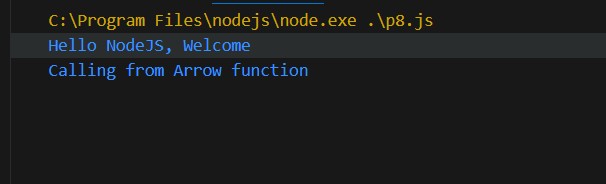
setTimeout(message,5000);

setTimeout(()=> {

console.log("Calling from Arrow function");

},8000);

# Output:



**Practical 10**

# To demonstrate module exports in Node.js

Function

add(a,b){

return a+b;

}

exports.add=add;

var req = require('./p6');

var res = req.add(15,9); console.log(res);

# Output:

**Practical 11**

# write an application to find area of circle, square, rectangle using module in Node.js

//square

function square(s){ return s\*s;

}

//rectangle

function rectangle(l,b){ return 1\*b;

}

//circle

function circle(r){ return 3014\*r\*r;

}

//export exports.square=square; exports.rectangle=rectangle; exports.circle=circle;

//import file

var req = require('./p7\_area.js'); var sRes,rRes,cRes;

//import module sRes=req.square(5); rRes=req.rectangle(4,6); cRes=req.circle(4);

//display result console.log("square:",sRes); console.log("rectangle:",rRes); console.log("circle:",cRes);

# Output:

**Practical 12**

# Write an application to demonstrate events module in Node.js

const EventEmitter = require('events');

const emitter = new EventEmitter();

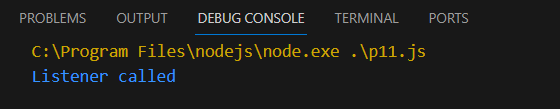
//REGISTER

emitter.on('messageLogged',function(){

console.log('Listener called');

}); //Raise emitter.emit('messageLogged')

# Output:



**Practical 13**

# write an application to demonstrate function (removeListner, listnerCount) in Node.js

const events = require("events"); const eventEmitter = new events.EventEmitter(); function listner1(){

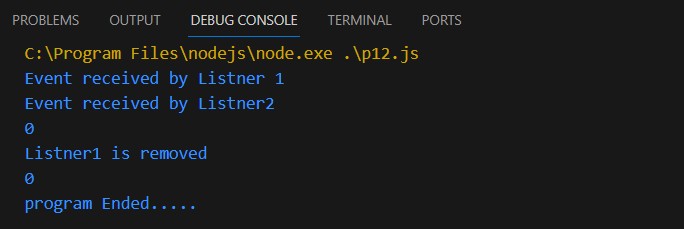
console.log("Event received by Listner 1");

} function listner2(){

console.log("Event received by Listner2");

} eventEmitter.addListener("Write",listner1); eventEmitter.on("Write",listner2); eventEmitter.emit("Write"); console.log(eventEmitter.listenerCount("write")); eventEmitter.removeListener("write",listner1); console.log("Listner1 is removed"); eventEmitter.emit("write"); console.log(eventEmitter.listenerCount("write")); console.log("program Ended ")

# Output:



**Practical 14**

# create an application in node.js to Return Event Emitter

var emitter = require('events').EventEmitter; function LoopProcessor(num){

var e = new emitter();

setTimeout(function(){

for(var i=1;i<=num;i++)

{

e.emit('BeforeProcess ',i); console.log('processing number: '+i);

e.emit('AfterProcess ',i);

}

}, 2000)

return e;

}

Var lp = LoopProcessor(3);

lp.on('BeforeProcess', function(data){

console.log('About to start the process for '+data);

});

lp.on('AfterProcess', function(data){

console.log('Completed processing '+data);

});

# Output:



**Practical 15**

create an application in node js to create Extend Event Emitter in Node.js

var emitter=require('events').EventEmitter;

var util = require('util');

function LoopProcessor (num) {

var me = this; setTimeout(function()

{

for (var i=1;i<=num;i++){

me.emit ('BeforeProcess',i);

console.log('processing number: '+i);

me.emit ('After Process',i);

}

},

2000)

Return this;

}

util.inherits (LoopProcessor, emitter)

var lp = new LoopProcessor (3);

lp.on('BeforceProcess', function(data)

{

console.log('About to start the process for' + data);

});

lp.on('AfterProcess', function(data) {

console.log('completed processing '+ data);

});

# Output:



**Practical 16**

# Write an event emitter code to design an event called as “calculate Salary” which is used to calculate the salary of an employee by passing some arguments like Basic Salary, HRA (20% of Basic), DA(100% of Basic), TA, and deductions like Income Tax (30% of Basic) and Professional Tax of 200.

const EventEmitter = require('events');

class SalaryCalculator extends EventEmitter { calculateSalary(basic, ta)

{

const hra = 0.2 \* basic; // HRA is 20% of Basic

const da = basic; // DA is 100% of Basic

const incomeTax = 0.3 \* basic; // Income Tax is 30% of Basic

const professionalTax = 200; // Professional Tax is 200

const salary = basic + hra + da + ta - incomeTax -professionalTax;

this.emit('calculateSalary', salary);

}

}

const salaryCalculator = new SalaryCalculator();

salaryCalculator.on('calculateSalary', (salary) => {

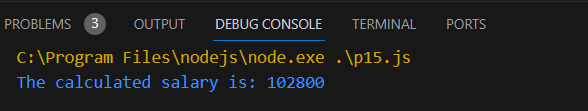
console.log(`The calculated salary is: ${salary}`);

});

// Example usage:

salaryCalculator.calculateSalary(50000, 8000); // Basic Salary is 50000 and TA is 8000

**Output**



# Practical 17

**create an application in node js to display message after 5 second &10 second**

const myfun = delay => {

console.log('Hello After ' +delay+ ' second');

};

setTimeout(myfun,5000,'five');

setTimeout(myfun,10000,'Ten');

# Output:

**Practical 18**

# create an application in node js to demonstrate set interval function

setInterval(

() => console.log('Hello After 4 Second'),4000

);

# Output:

**Practical 19**

# create an application in node js to display factorial of a number

function factorial(n){

let i=n;

let res=1;

while(i>+1){

res = res\*i;i--

}

return res;

}

const num = 6;

const result = factorial(num); console.log(result);

# Output:

**Practical 20**

# Write as application to create http Server and Display message in Node.js

var http = require('http');

var server = http.createServer(function(req,res){

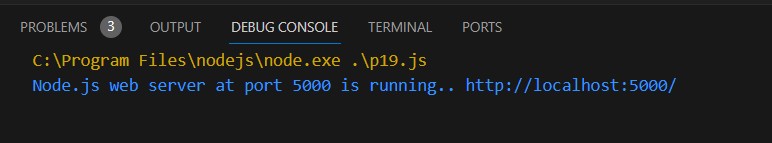
res.write("Hello Mr.Dhawal");

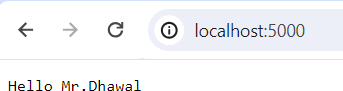
res.end();

}); server.listen(5000);

console.log('Node.js web server at port 5000 is running.. http://localhost:5000/');

# Output:





**Practical 21**

# Write a Node.js code to display Employee Job Registration

**Form saved in an HTML file in response to the client’s access request to the server.**

const http = require('http');

const fs = require('fs');

http.createServer((req, res) => {

fs.readFile('form.html', (err, data) => {

if (data) {

res.writeHead(200, { 'Content-Type': 'text/html' });

res.end(data);

}

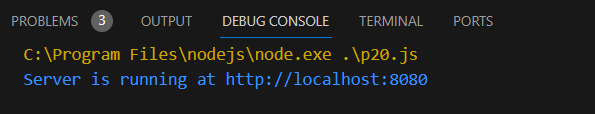
});

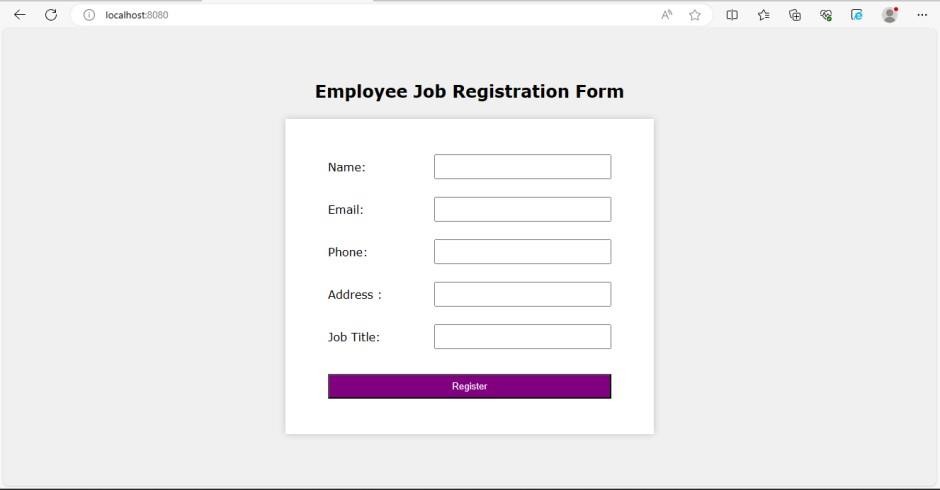
}).listen(8080, () => {

console.log('Server is running at http://localhost:8080');

});

**Output :**





# Practical 22

**Write as application to create Home page, Admin page and Student page using http server in Node.js.**

var http = require('http')

const{text} = require('stream/consumers');

var server = http.createServer(function (req,res){ if (req.url=='/'){

res.writeHead(200,{'content-type':'text/html'}); res.write('<html></head><body>');

res.write('<style> ul li{display : inline-block; float : right; height : 40px;} ul li a{padding : 20px; background : orange; color : white;}</style>');

res.write('<div><h1>My First Website</h1></div><div><ul><li><a href="/admin">Contact Admin</a></li><li><a href="/student">Student</a></li><li><a href="/home">Home</a></li></ul></div></div>');

res.write('<div style="background : white; padding : 20px;"><h2>Start Page</h2><p>This is my first webpage hehe!</p><p>Hi Everyone</p></div></body></html>');

res.end();

}

else if (req.url=='/home'){ res.writeHead(200,{'content-type':'text/html'});

res.write('<html><head><style>body{padding-left : 43px; padding-right : 43px; background-color : lightyellow;} </style></head><body><p><h1>This is home page</h1></p><h1>Sagar Parab</h1><h3>This page is a brief insight to who I am.</h3>');

res.write('<nav style="background-color:black; text-align:center;"><ul><li><a href="/">Start Page</a></li><li><a href="/student">Student</a></li><li><a href="/admin">Admin</a></li></ul></nav></body></html>');

res.end();

}

else if (req.url=='/student'){ res.writeHead(200,{'content-type':'text/html'});

res.write('<div style="display: inline-block; float: right; height: 40px; padding: 20px;"><ul><li><a href="/home">Home</a></li><li><a href="/">Start Page</a></li> <li><a href="/admin">Contact Admin</a></li></ul></div>');

res.write('<html><head><style>body{background- color:pink;}</style><title>Form</title></head><body bgcolor="White" ><h1 align="center">Student Page Form</h1>');

res.write('<form action="url" method="post"><fieldset><legend>Personal Imformation</legend>');

res.write('<lable><Strong>Student Name</strong></lable><br/><input type="text" name="Student Name" placeholder="Enter Your Name" /><br/>');

res.write('<lable><Strong>Email</strong></lable><br/><input type="email" name="eamil" placeholder="Enter Your Email Address" /></br>');

res.write('<lable><Strong>Password</strong></lable><br/>'); res.write('<input type="password" name="Password" placeholder="Enter Your

Password" /></br><lable><Strong>Gender</strong></lable><br/>'); res.write('<input type="Radio" name="Gender" value="Male" />Male <input

type="Radio" name="Gender" value="FeMale" />FeMale<br/>'); res.write('<lable><Strong>Hobbies</strong></lable><br/>'); res.write('<input type="checkbox" name="Hobbies" value="Playing Sports"

/>Playing Sports<br/>');

res.write('<input type="checkbox" name="Hobbies" value="Listening Music"

/>Listening Music<br/>');

res.write(' <input type="checkbox" name="Hobbies" value="Traveling"

/>Traveling<br/><input type="checkbox" name="Hobbies" value="Reading Books" />Reading Books<br/>');

res.write('<lable><Strong>Select Your City</strong></lable><select name="City">');

res.write('<option value="Ahemdabad">Ahemdabad</option><option value="Kalol">Kalol</option><option value="Surat">Surat</option>');

res.write(' <option value="Rajkot">Rajkot</option></select></br><input type="submit" onclick=alert("Thanks!") name="submit" value="Submit"/></form>');

res.end();

}

else if (req.url=='/admin'){ res.writeHead(200,{'content-type':'text/html'});

res.write('<style>ul li{display: inline-block; float: right; height: 40px;} ul li a{padding: 20px; background:orange; color: white;}</style>');

res.write('<div><ul><li><a href="/admin">Contact Admin</a></li><li><a href="/student">Student</a></li><li><a href="/home">Home</a></li></ul></div></div><br><br>');

res.write('<html><head><style>legend{text-align:center;} body{background- color:faf89a;border: 5px solid darkred;} form{display: inline-block; float: center; padding: 20px;} ');

res.write('border-radius:4px; padding:40px 5px; max- width:100%;}</style></head>');

res.write('<legend><h1><u>Admin Login</u></h1></legend>'); res.write('<form action="#" method="POST" autocomplete="off">'); res.write('<div class="input\_field"><h3>Username</h3></div><div

class="input\_field"><input type="text" ');

res.write('name="userid" placeholder="Username" required/></div>'); res.write('<div class="input\_field"><h3>Password</h3></div><div

class="input\_field"><input type="Password"');

res.write('name="pword" placeholder="Password" required/></div><p>'); res.write('<style>button{border:none; border-radius:5px; text-align:center;

padding:15px 15px; background-color:lavender;<div></div></style>'); res.write('<button onclick=alert("SUCESS")>LOGIN

NOW</button></form>');

res.end();

}

else{

res.end('Invalid request');

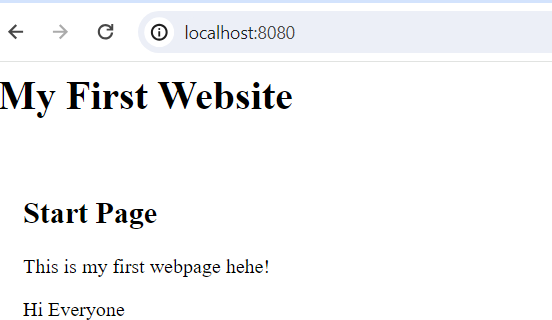
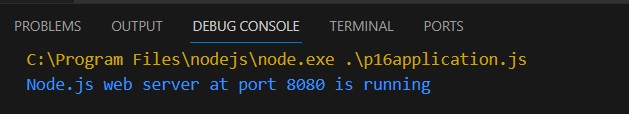
}

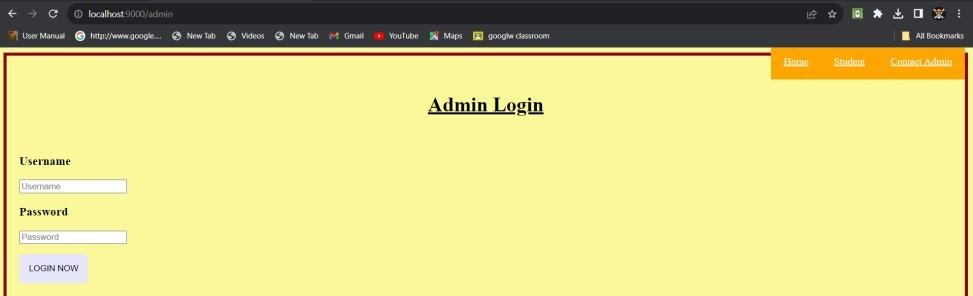
});

server.listen(8080);

console.log('Node.js web server at port 8080 is running');

# Output:





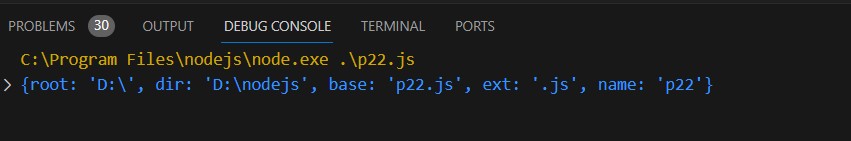
**Practical 23**

# Write in application to display details of the current file path in Node.js.

const location = require("path");

const localobj = location.parse( filename); console.log(localobj);

# Output:



**Practical 24**

# Write an application to read file in Node.js.

const fs = require('fs');

fs.readFile("p24.js",'utf8',function(err,data)

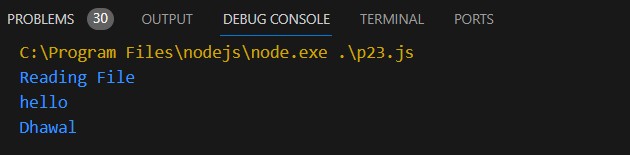
{

console.log("Reading File");

console.log(data);

});

# Output:



**Practical 25**

# Write an application to write in file in Node.js.

const fs = require("fs");

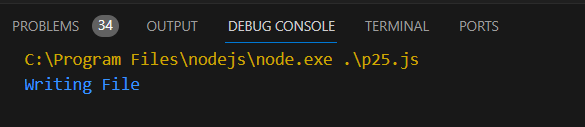
fs.writeFile("\_txt.txt",'Welcome to the live stream',function (err,data)

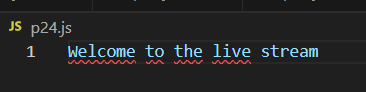
{

console.log("Writing File");

});

# Output:





**Practical 26**

# Write an application to add data in file in Node.js.

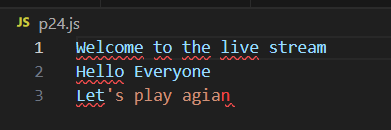
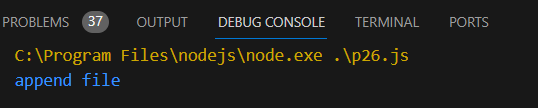
const fs = require("fs");

fs.appendFile("\_txt.txt","\nHello Everyone \nLet's play agian", function (err,data){

console.log("append file");

});

# Output:



**Practical 27**

# Write an application to delete a file in Node.js.

const fs = require("fs");

fs.unlink("p24.js",function(err,data)

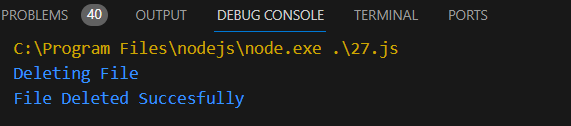
{

console.log("Deleting File");

console.log("File Deleted Succesfully");

});

# Output:



**Practical 28**

# Combine Read, Write, Append, Delete file in one program in Node.js

const fs = require("fs");

fs.writeFile("\_com.txt",'Hello world',function (err,data)

{

console.log("Writing File");

});

fs.appendFile("\_com.txt","\nHello Everyone \nGive ThumbsUp",function

(err,data) {

console.log("append file");

});

fs.readFile("\_com.txt",'utf8',function(err,data)

{

console.log("Reading File");

console.log(data);

});

fs.unlink("\_com.txt",function(err,data)

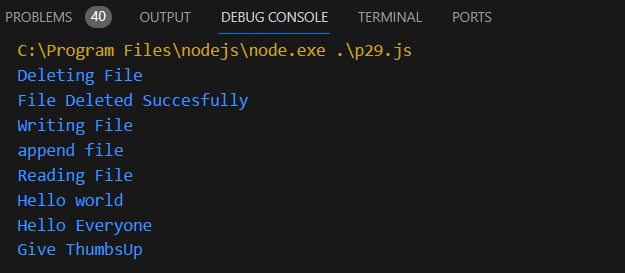
{

console.log("Deleting File");

console.log("File Deleted Succesfully");

});

# Output:



**Practical 29**

# Write and application to rename a file in Node.js

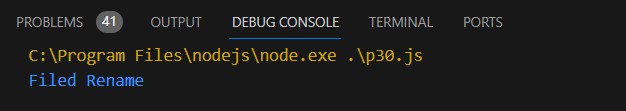
var fs = require('fs')

fs.rename('p31.js','p33.js',function(err){

if(err) throw err; console.log('Filed Rename')

});

# Output:



**Practical 30**

# Create an Application to create Database in Node.js

var mysql = require('mysql')

var con = mysql.createConnection({

host:'localhost',

user:'root',

password:'root'

});

con.connect(function(err){

if(err){ throw err;}

else{ console.log("connected");}

con.query("CREATE DATABASE STUDENTS4", function(err,result){

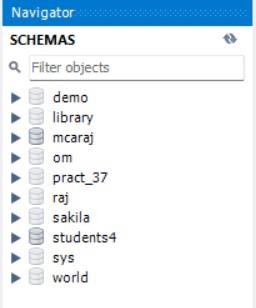
if(err) throw err;

console.log("Database Created");

});

});

# Output:



**Practical 31**

# Create an Application to create Student table with columns as id, name, address, course, contact in Node.js

var mysql=require('mysql');

Var con=mysql.createConnection

({

host:'localhost', user:'root', password:'root', database:'students4'

});

con.connect(function(err)

{

if(err) throw err; console.log("connected...");

var sql = "CREATE TABLE student1(id INT(10) PRIMARY KEY AUTO\_INCREMENT,name VARCHAR(255), address VARCHAR(255),course VARCHAR(20), contact INT(15))";

con.query(sql,function(err,result)

{

if(err)

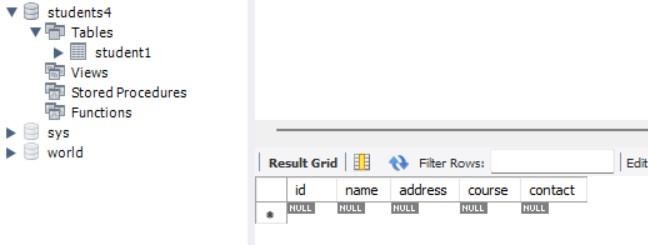
throw err;

console.log("table created...");

});

});

# Output:



**Practical 32**

# Create an Application to insert rows into Student table in Node.js

var mysql=require('mysql');

Var con=mysql.createConnection ({

host:'localhost', user:'root', password:'root', database:'students4'

});

con.connect(function(err)

{

if(err) throw err; console.log("connected...");

var sql2 = "INSERT INTO student1(id ,name , address,course , contact) VALUES('1','raj','thane','MCA','1234567890')";

con.query(sql2,function(err,result)

{

if(err) throw err;

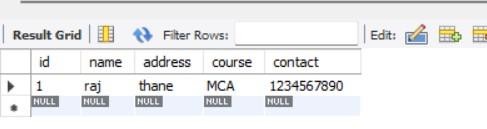
console.log();

console.log("row inserted successfuly...");

});

});

# Output:



**Practical 33**

# Create an Application to display rows into Student table in Node.js

var mysql=require('mysql');

Var con=mysql.createConnection ( {

host:'localhost', user:'root', password:'root', database:'students4'

});

con.connect(function(err)

{

if(err) throw err; console.log("connected...");

var sql2="select \* from student1"; con.query(sql2,function(err,result)

{

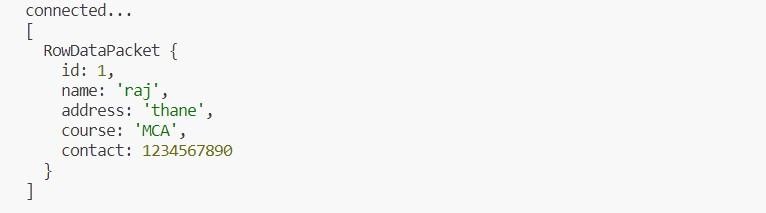
if(err) throw err;

console.log(result);

});

});

# Output:



**Practical 34**

# Create an Application to Update rows in Student table in Node.js

var mysql=require('mysql'); var

con=mysql.createConnection ({

host:'localhost', user:'root', password:'root', database:'students4'

});

con.connect(function(err)

{

if(err) throw err;

console.log("connected...");

var sql2= "UPDATE student1 SET course ='MMS' WHERE ID='1'"; con.query(sql2,function(err,result)

{

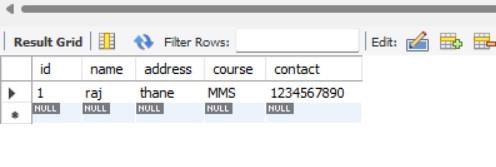
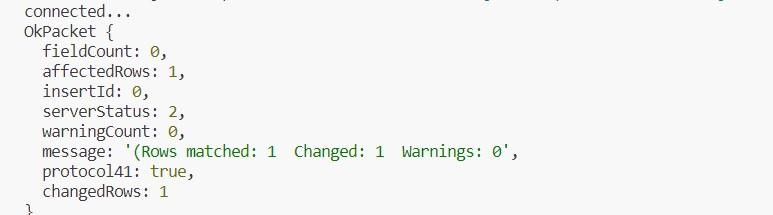
if(err) throw err;

console.log(result);

});

});

# Output:



**Practical 35**

# Write a Node.js application to retrieve and update the record related to the entries received for the conference participation. Update the mobile number of participant whose name is “Sharma

var mysql = require('mysql');

var con = mysql.createConnection({

host: "localhost",

user: "root",

password: "12345", database:"pract\_37"

});

con.connect(function(err) {

if (err) throw err;

console.log("Connected successfully to server");

var sql = "SELECT \* FROM participants WHERE name = 'Sharma'"; con.query(sql, function(err, result) {

if (err) throw err;

console.log("Participant found: ", result);

var newMobileNumber = '1234567890';

var updateSql = `UPDATE participants SET mobile = '${newMobileNumber}' WHERE name = 'sharma'`;

con.query(updateSql, function(err, result) {

if (err) throw err;

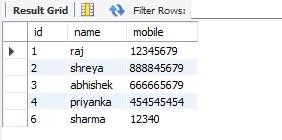
console.log("Number of records updated: " + result.affectedRows);

});

}); });

**Output :**





# Practical 36

**Create an Application to add column to Student table in Node.js**

var mysql=require('mysql'); var

con=mysql.createConnection ({

host:'localhost', user:'root', password:'root', database:'students4'

});

con.connect(function(err)

{

if(err) throw err; console.log("connected...");

var sql = "ALTER TABLE student1 ADD age INT(5)"; con.query(sql,function(err,result)

{

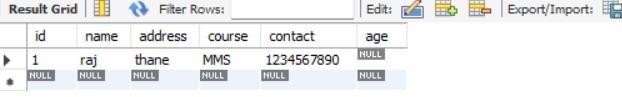
if(err) throw err;

console.log("column inserted successfuly...");

});

});

# Output:



**Practical 37**

# Create an Application to delete records in Student table in Node.js

var mysql=require('mysql'); var

con=mysql.createConnection ({

host:'localhost', user:'root', password:'root', database:'students4'

} );

con.connect(function(err)

{

if(err) throw err; console.log("connected...");

var sql= "DELETE FROM student1 WHERE ID='1'";

con.query(sql,function(err,result)

{

if(err) throw err;

console.log("row deleted successfuly...");

});

});

Output:

