**Node.js Practical**

|  |  |  |
| --- | --- | --- |
| Sr. No. | Practical Name | Date |
| 1 | Steps to download node.js |  |
| 2 | Steps to download visual studio |  |
| 3 | Demonstrate the basic arithmetic operations in Node.js |  |
| 4 | To determine whether a given number is even or odd in Node.js |  |
| 5 | To print all prime numbers up to a given number in Node.js |  |
| 6 | Create an application in Node.JS to reverse the given number and display it |  |
| 7 | Create an application in Node.js to display Armstrong number 15 |  |
| 8 | To generate the first 10 numbers in the Fibonacci sequence in Node.js |  |
| 9 | To demonstrate the use of setTimeout and arrow functions in Node.js |  |
| 10 | To demonstrate module exports in Node.js |  |
| 11 | write an application to find area of circle, square, rectangle using module in Node.js |  |
| 12 | Write an application to demonstrate events module in Node.js |  |
| 13 | write an application to demonstrate function (removeListner, listnerCount) in Node.js |  |
| 14 | create an application in node.js to Return Event Emitter |  |
| 15 | create an application in node.js to create Extend Event Emitter in Node.js |  |
| 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 |  |
| 17 | create an application in node.js to display message after 5 second &10 second |  |
| 18 | create an application in node.js to demonstrate set interval function |  |
| 19 | create an application in node.js to display factorial of a number |  |
| 20 | Write as application to create http Server and Display message in Node.js |  |
| 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. |  |
| 22 | Write as application to create Home page, Admin page and Student page using http server in Node.js. | 16/10/2023 |
| 23 | Write in application to display details of the current file path in Node.js. |  |
| 24 | Write an application to read file in Node.js |  |
| 25 | Write an application to write in file in Node.js. |  |
| 26 | Write an application to add data in file in Node.js. |  |
| 27 | Write an application to delete a file in Node.js |  |
| 28 | Combine Read, Write, Append, Delete file in one program in Node.js |  |
| 29 | Write and application to rename a file in Node.js |  |
| 30 | Create an Application to create Database in Node.js |  |
| 31 | Create an Application to create Student table with columns as id, name, address, course, contact in Node.js |  |
| 32 | Create an Application to insert rows into Student table in Node.js |  |
| 33 | Create an Application to display rows into Student table in Node.js |  |
| 34 | Create an Application to Update rows in Student table in Node.js |  |
| 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 |  |
| 36 | Create an Application to add column to Student table in Node.js |  |
| 37 | Create an Application to delete records in Student table in Node.js |  |

**Angular Practical**

|  |  |  |
| --- | --- | --- |
| Sr. No. | Practical Name | Date |
| 1 | Create an application in angular.js to demonstrate arithmetic operations and list. |  |
| 2 | Create an application in angular.js to calculate registration fees if the number of people and registration amount is given by the user |  |
| 3 | Create an application in angular.js to calculate simple interest take appropriate input from the user |  |
| 4 | Write an application in angular.js to create an array of names and display all the names which has letter “i” using controller |  |
| 5 | Create an application in angular.js to demonstrate the use of filters |  |
| 6 | Create an application in angular.js to change the background color as the user changes input in the textbox |  |
| 7 | Create an application in angular.js to demonstrate to display text in alert box |  |
| 8 | Create an application in angular.js to demonstrate the use of ng-if, ng-disabled and ng-readonly |  |
| 9 | Create an application in angular.js to demonstrate use of mouse enter and mouse-leave even |  |
| 10 | Write an application in angular js to display options using select tag as user chooses the color option the respective color and content should change |  |
| 11 | Write an Angular JS code to display a Registration form for Student applying for a new Course. Display all the values entered by the students. |  |
| 12 | To demonstrate the use of regular expressions for validating input fields in a form |  |
| 13 | To demonstrate use of validation directives. |  |
| 14 | To demonstrate the state properties of form fields |  |
| 15 | To demonstrate the use of a Single Page Application (SPA) |  |
| 16 | Create an application with Login page and Registration Page using Single Page Application(SPA) |  |

**Practical 1-Steps to download node.js**

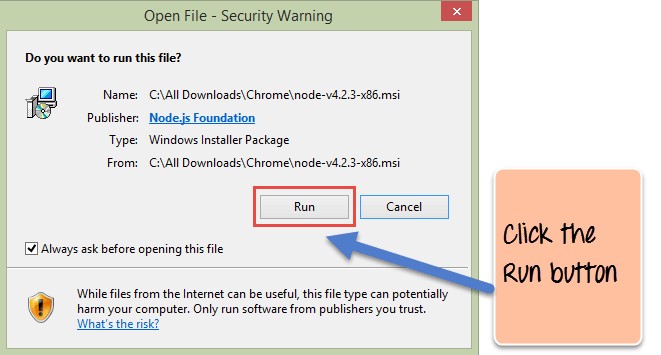
Step 1) Download Node.js Installer for Windows

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

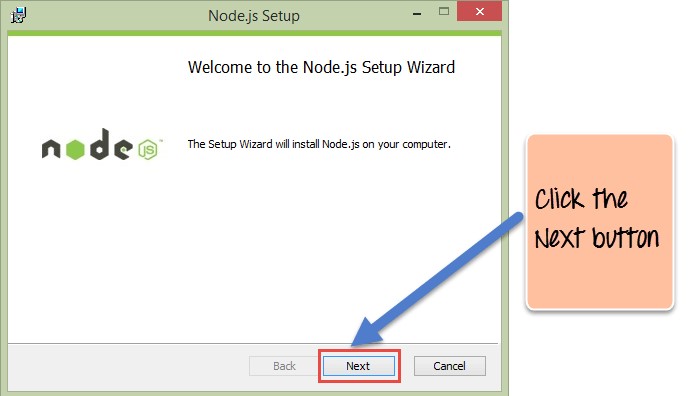


**Step 2)** Run the installation Double click on the downloaded .msi file to start the installation.

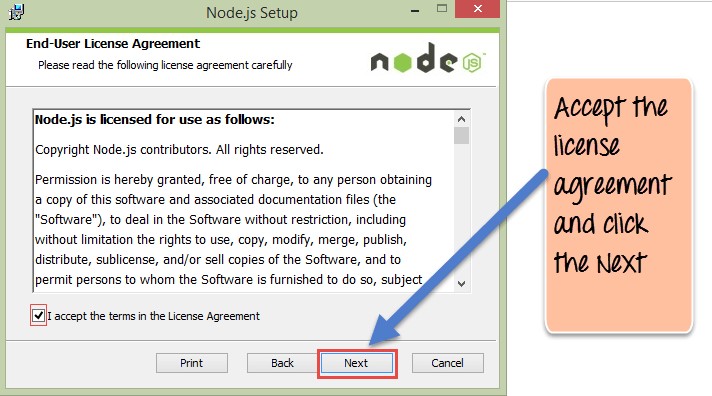
Click the Run button on the first screen to begin the installation.



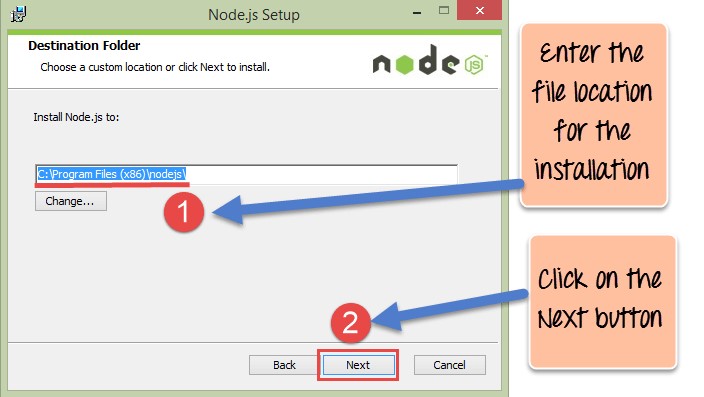
**Step 3)** Continue with the installation steps In the next screen, click the “Next” button to continue with the Node.js download and installation



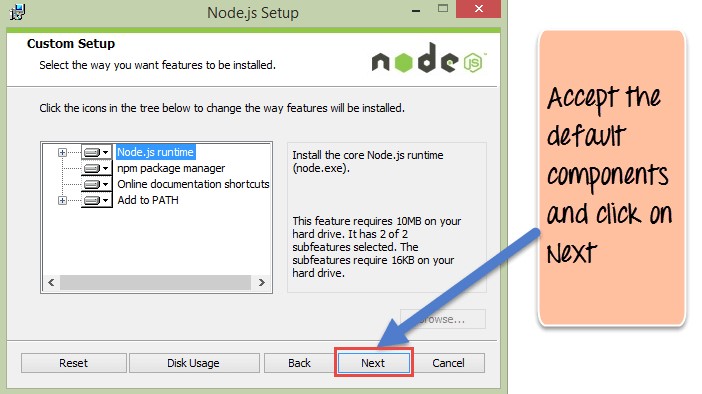
**Step 4)** Accept the terms and conditions In the next screen, Accept the license agreement and click on the Next button



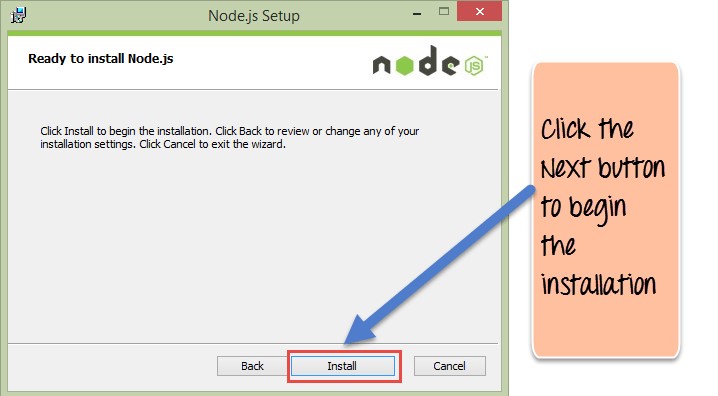
**Step 5)** Set up the path In the next screen, choose the location where Node.js needs to be installed and then click on the Next button.



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

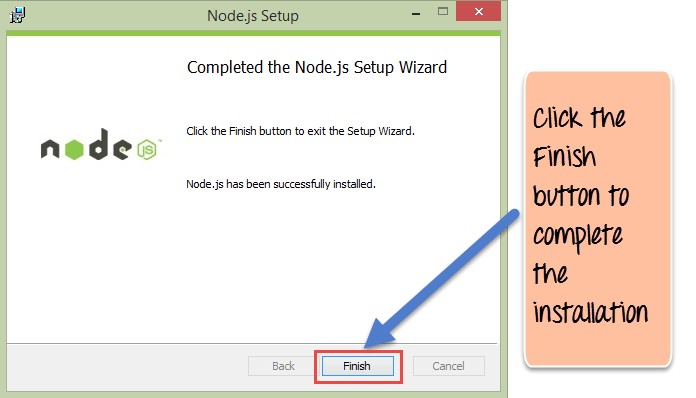


**Step 7)** Start the installation In the next screen, click the Node.js install button to start installing on Windows



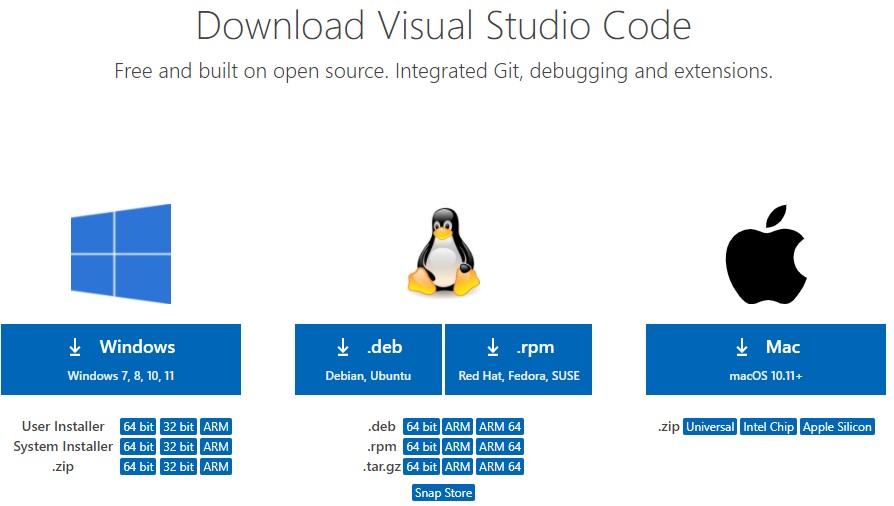
**Step 8**) Complete the installation Click the Finish button to complete the installation.

Complete the installation Click the Finish button to complete the installation.

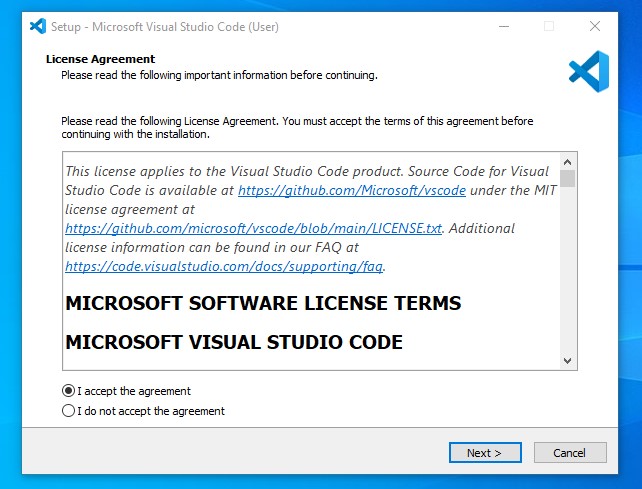


**Practical 2-Steps to download visual studio**

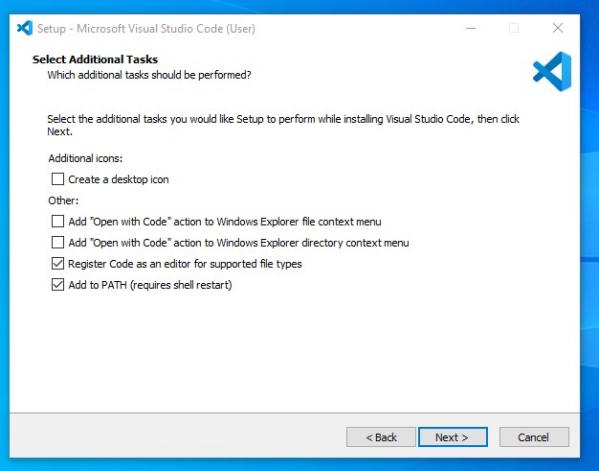
Step 1: **Visit the [official website](https://code.visualstudio.com/docs/?dv=win) of the** Visual Studio Code **using any web browser like Google Chrome, Microsoft Edge, etc. and Press the “**Download for Windows**” button**



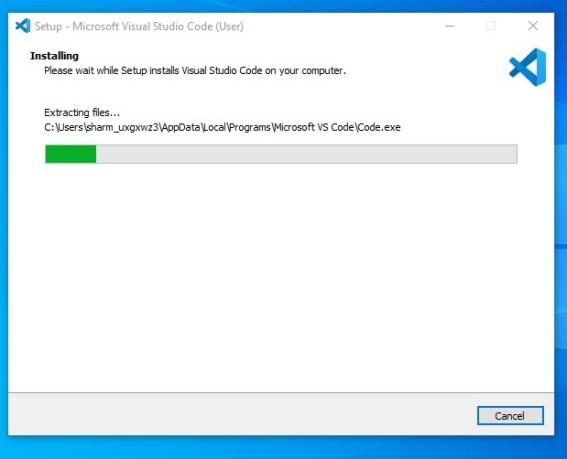
Step 2:When the download finishes, then the Visual Studio Code icon appears in the downloads folder. Click on the installer icon to start the installation process of the Visual Studio Code. After the Installer opens, it will ask you for accepting the terms and conditions of the Visual Studio Code. Click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



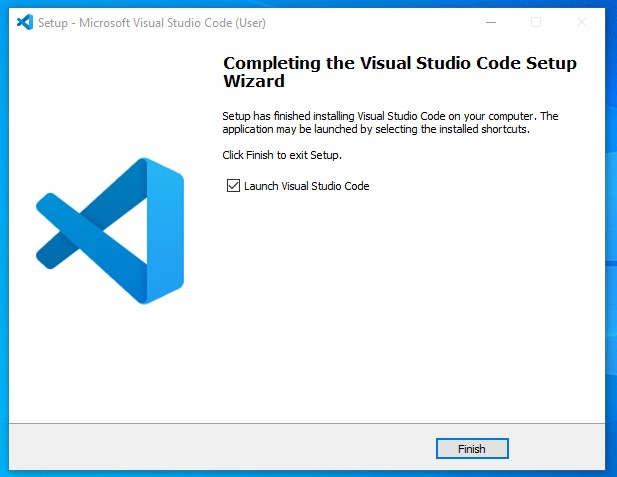
Step 3: **Choose the location data for running the Visual Studio Code. It will then ask you for browsing the location. Then click on** Next **button.**



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- Performing Arithmetic Operations(+,-,\*,/) in node.js?**

Code-

function add(a,b)

{

return a+b;

}

function sub(a,b)

{

return a-b;

}

function mult(a,b)

{

return a\*b;

}

function div(a,b)

{

return a/b;

}

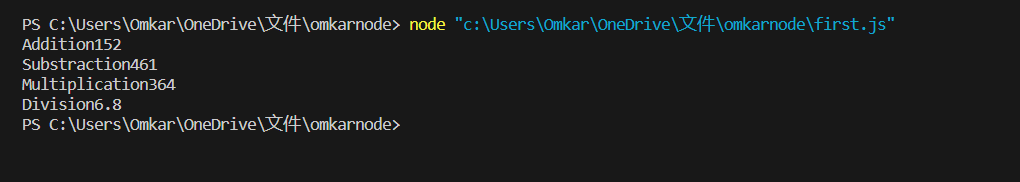
console.log("Addition"+add(100,52));

console.log("Substraction"+sub(512,51));

console.log("Multiplication"+mult(7,52));

console.log("Division"+div(34,5));

Output-



**Practical 4- Even/Odd number in node.js?**

Code-

function EvenOdd(a)

{

if (a%2 == 0) {

return true;

}

else {

return false;

}

}

a = 100;

//a = 101; //for odd number

result = EvenOdd(a);

if (result == true) {

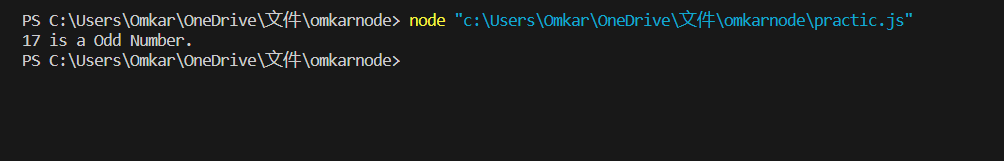
console.log(a + " is an Even Number.");

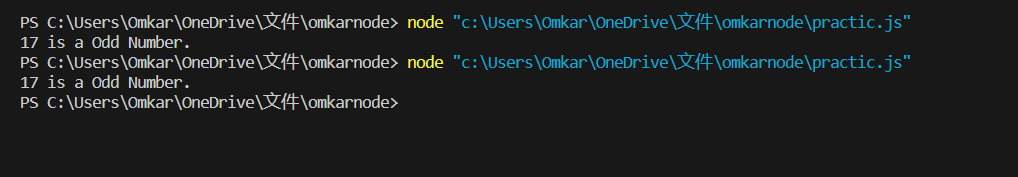
} else {

console.log(a + " is a Odd Number.");

}

Output-





**Practical 5-Create an application in node.js to Print Prime numbers .**

Code-

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 =100;

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

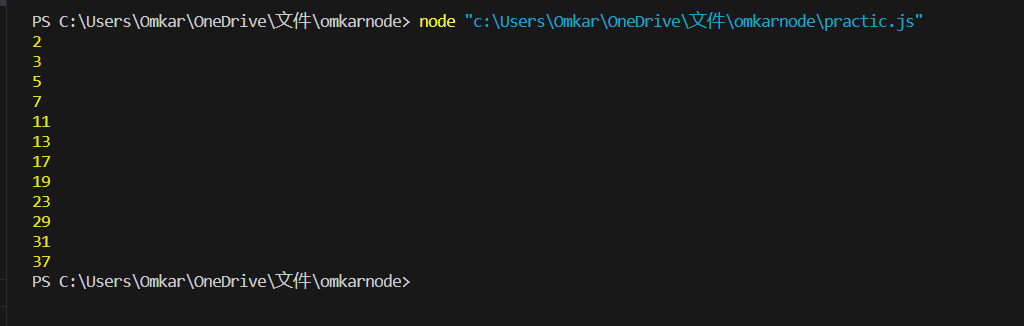
if(isPrime(i)){

console.log(i);

}

}

Output-



**Q-Determine whether given number is Prime number using Node js?**

Code-

function isPrime(a){

if(a <= 1)

return false;

for(i = 2; i <= a/2; i++)

if(a%i == 0)

return false;

return true;

}

a = 4546;

result = isPrime(a);

if (result == true) {

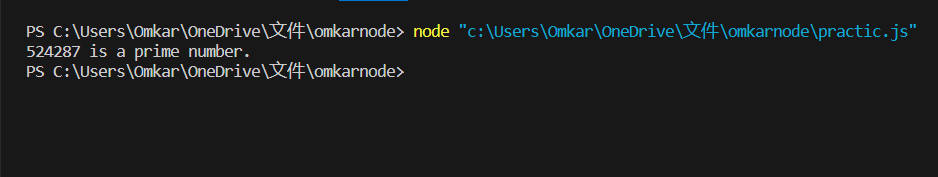
console.log(a + " is a prime number.");

} else {

console.log(a + " is not a prime number.")

}

Output-



**Practical 6-Create an application in NodeJS to reverse the given number and display it (Note: 5 digit number)**

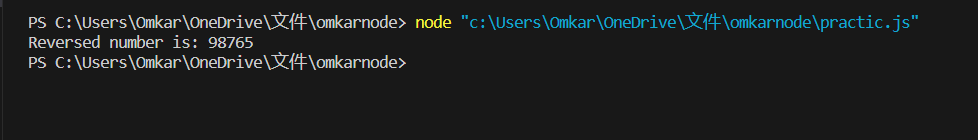
Code-

var number = 56789;

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

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

output-



**Practical 7- Create An Application Node.js to display Armstrong number.**

Code-

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++;

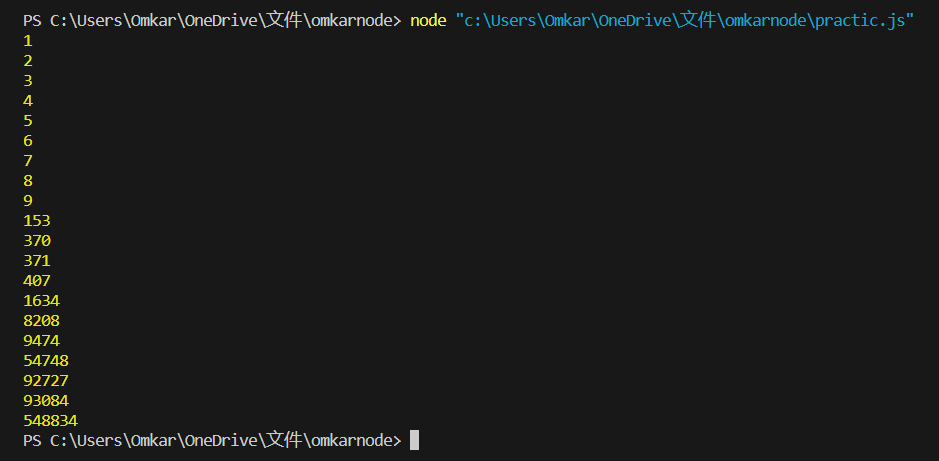
}

num++;

} }

printFirstNArmstrongNumbers(20);

output-



**Practical 8-Create an application to Display or Print Fibonacci series.**

Code-

var a=0;

var b=1;

var c;

console.log(a);

console.log(b);

for(i=0;i<10;i++)

{

c=a+b;

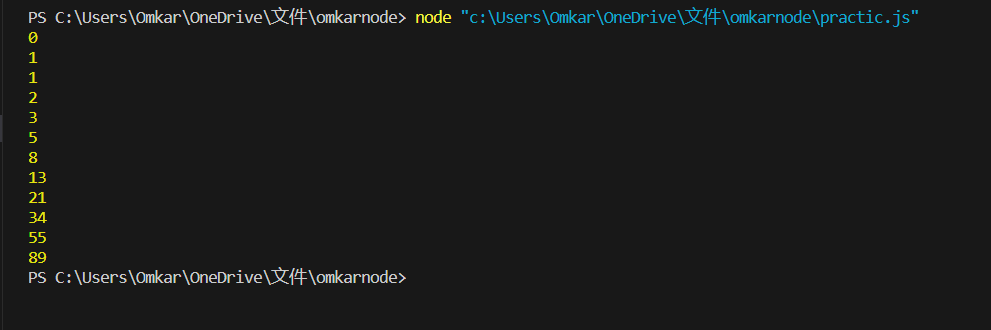
console.log(c);

a=b;

b=c;

}

Output-



**Practical 9- Write An Application To demonstrate arrow function.**

Code-

const message= function()

{

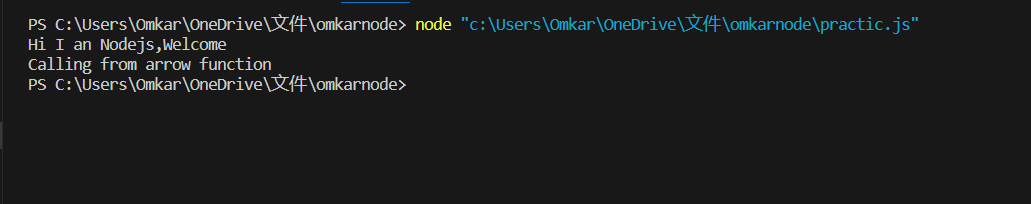
console.log("Hi I an Nodejs,Welcome");

}

setTimeout(message,5000);

setTimeout(()=>{console.log("Calling from arrow function");},8000);

Output –



**Practical 10- Write An application to demonstrate module.**

Code-

function add(a,b){

return a+b;

}

exports.add=add;

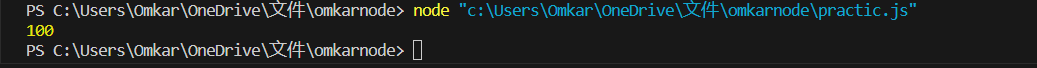
filename – p6\_firstmod.js

var req = require('./p6\_first.js');

var res = req.add(49,51);

console.log(res);

Output-



**Practical 11- Write An application to find area of circle ,square,rectangle,using module.**

Code-

//SQUARE

function square(s){

return s\*s;

}

//RECTANGLE

function rectangle(l,b)

{

return l\*b;

}

//CIRCLE

function circle(r)

{

return 3.14\*r\*r;

}

//Exports

exports.square=square;

exports.rectangle=rectangle;

exports.circle=circle;

step2-create a file - p7\_area.js

program-

var req=require('./p7\_area.js');

var sRes,rRes,cRes

sRes=req.square(5);

rRes=req.rectangle(15,4)

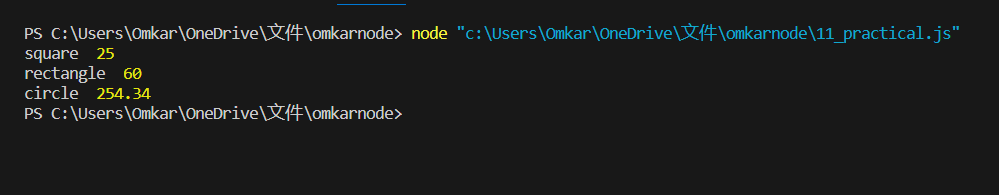
cRes=req.circle(9);

console.log("square ",sRes);

console.log("rectangle ",rRes);

console.log("circle ",cRes);

Output-



**Practical 12- Write an application to demonstrate events module.**

Code-

const EventEmitter = require('events');

const emitter=new EventEmitter();

//Register

emitter.on('messaglogged',function()

{

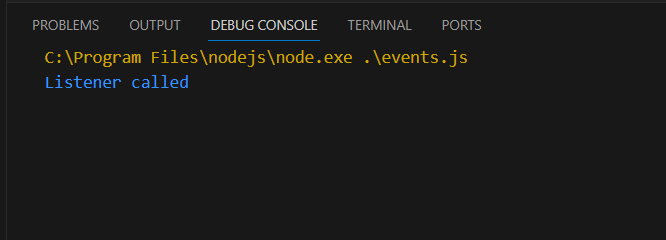
console.log('Listener called');

});

//Raise

emitter.emit('messagelogged');

Output-



**Practical 13- Write An application to demonstrate function removeListener,listenerCount.**

Code-

//Write an application to demonstarte function(remove listener,listenerCount)

const events = require("events");

const eventEmitter =new events.EventEmitter();

function listener1()

{

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

}

function listener2()

{

console.log("Event received by listener 2");

}

eventEmitter.addListener("Write",listener1);

eventEmitter.on("write",listener2);

eventEmitter.emit("write");

console.log(eventEmitter.listenerCount("write"));

eventEmitter.removeListener("write",listener1);

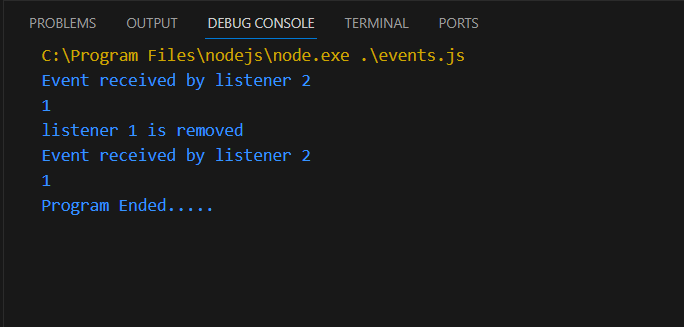
console.log("listener 1 is removed");

eventEmitter.emit("write");

console.log(eventEmitter.listenerCount("write"))

console.log("Program Ended.....");

Output-



**Practical 14- Create an Application in nodejs to return event emitter**.

Code-

// Create an application in nodejs to create 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:'+1);

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);

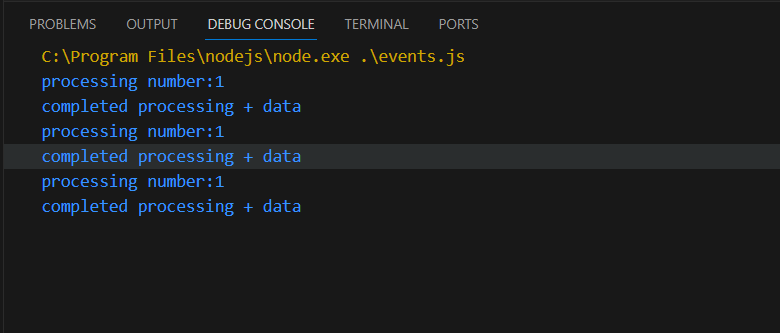
});

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

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

});

Output-



**Practical 15- Write a Program in NodeJS to create Extent Event Emitter?**

Code-

// Create an application in nodejs tp create extent event Emitter

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('AfterProcess',i);

}

},2000)

return this;

}

util.inherits(LoopProcessor,emitter)

var lp=new LoopProcessor(3);

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

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

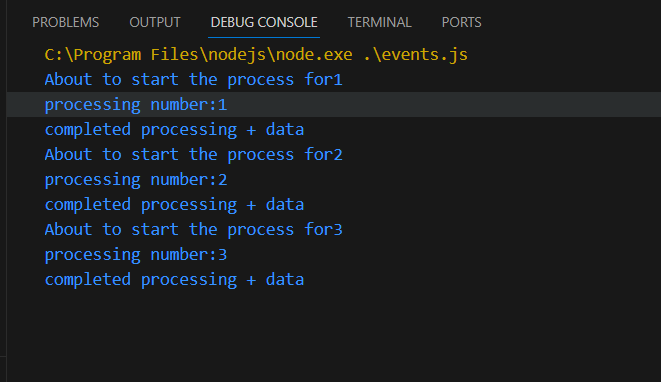
});

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

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

});

Output-



**Practical 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.**

Code-

const EventEmitter = require('events');

class SalaryCalculator extends EventEmitter { calculateSalary(basic, ta) { const hra = 0.2 \* basic;

const da = basic;

const incomeTax = 0.3 \* basic;

const professionalTax = 200;

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

this.emit('calculateSalary', salary);

}

} const salaryCalculator = new SalaryCalculator();

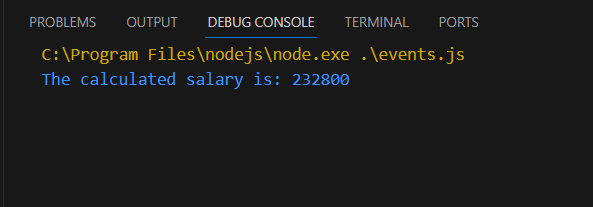
salaryCalculator.on('calculateSalary', (salary) => { console.log(`The calculated salary is: ${salary}`);

});

// Example usage:

salaryCalculator.calculateSalary(70000, 100000); // Basic Salary is 70000 and TA is 100000

Output-



**Practical 17- Create an application in node js to display message after 5 second and 10 second**.

Code-

const myfun = delay => {

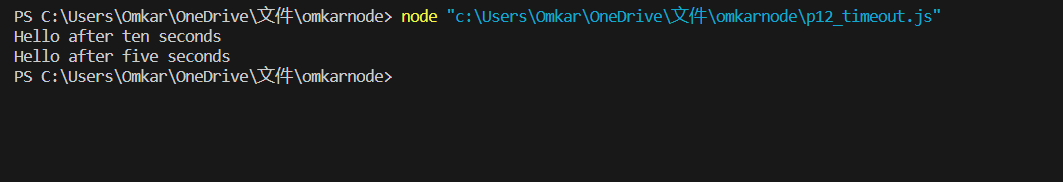
console.log('Hello after '+ delay +' seconds');

};

setTimeout(myfun,5000,'five');

setTimeout(myfun,1000,'ten');

Output-



**Practical 18- Create An application to demonstrate Set Interval function in nodejs.**

Code-

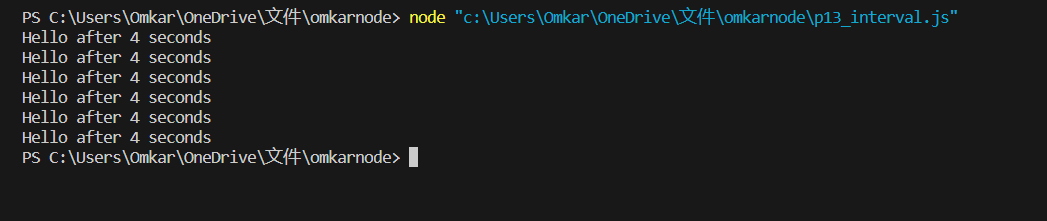
// Create an application in nodejs to demonstarte set interval function

setInterval(

()=>console.log('Hello after 4 seconds'),4000

)

Output-



**Practical 19- Create an application in nodejs to display factorial of number**

Code-

function factorial(n){

let i=n;

let res =1;

while(i>+1)

{

res=res\*i;

i--

}

return res;

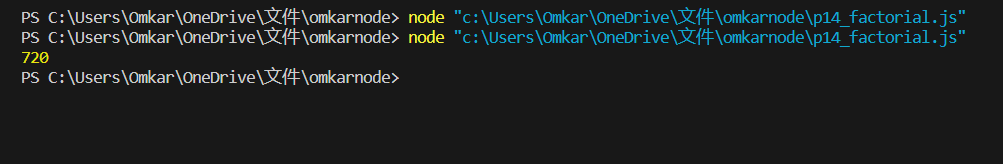
}

const num=10;

const result=factorial(num);

console.log(result);

Output-



**Practical 20- Creating server in node js.**

Code-

var http = require('http');

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

res.write("fymca satyam yadav");

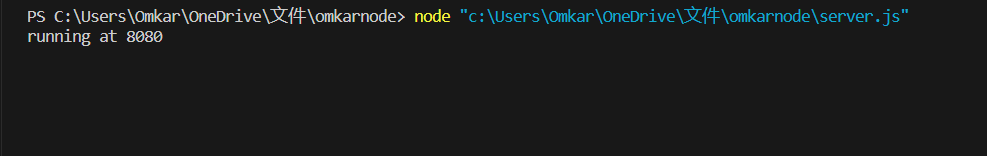
res.end();

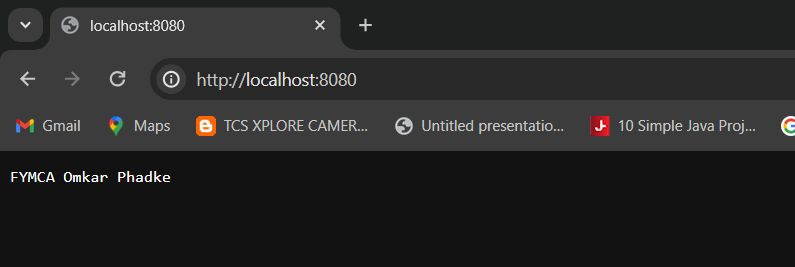
});

server.listen(8080);

console.log('running at 8080')

Output-





**Practical 21- Write an application to create home page ,admin page and student page using http server.**

Code-

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');

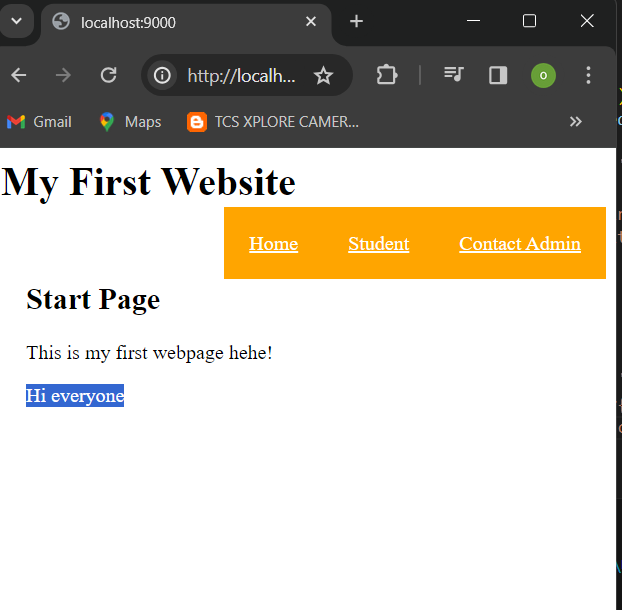
}

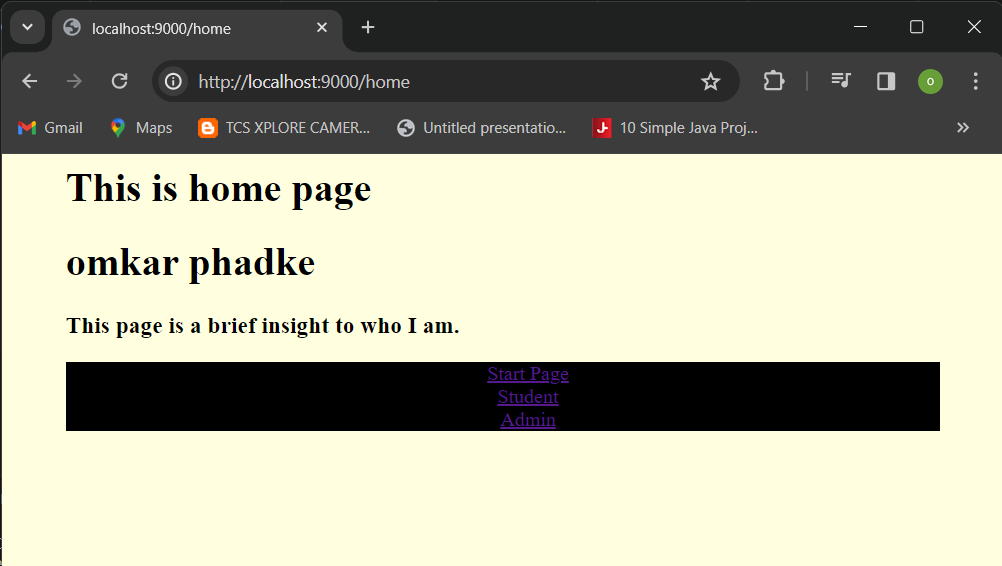
});

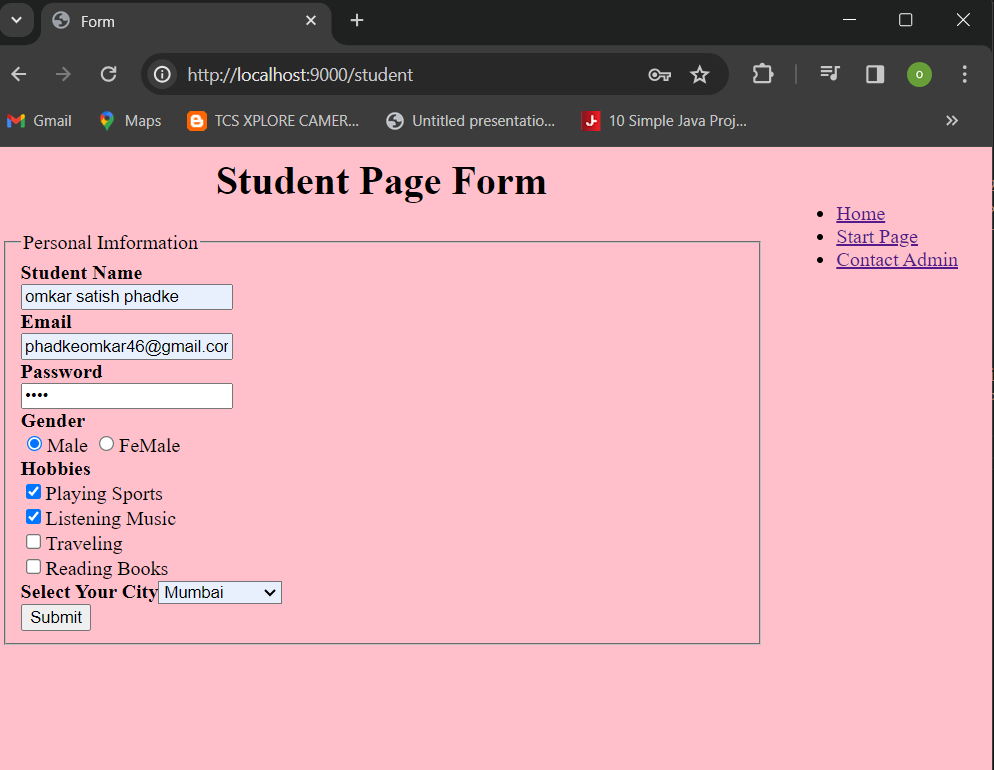
server.listen(9000);

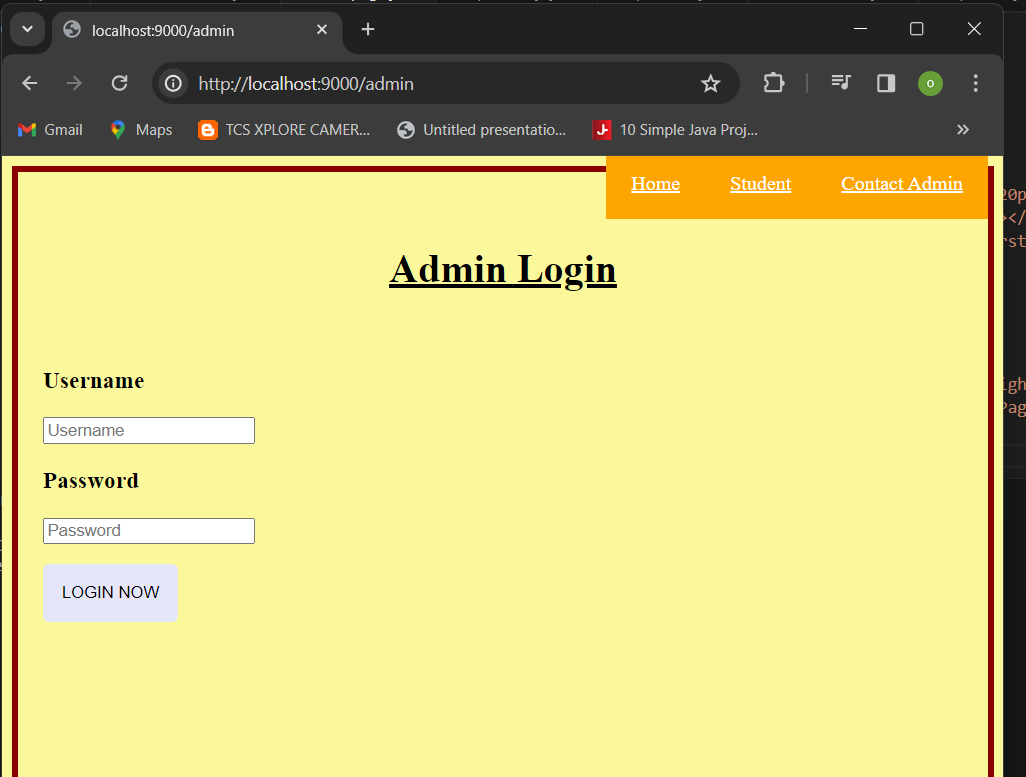
console.log('Node.js web server at port 9000 is running');

**Output-**









**Practical 22- 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.**

**Code-**

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);

}

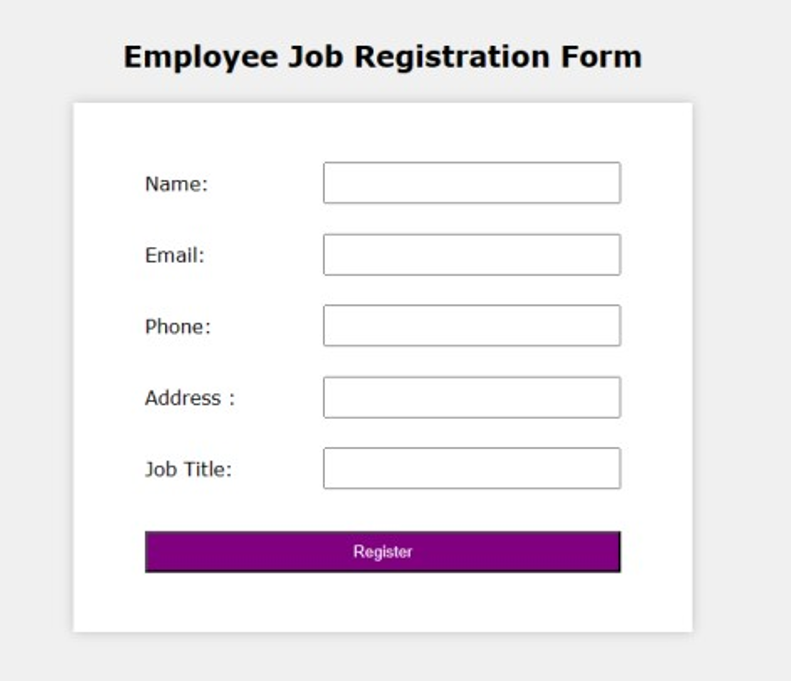
});

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

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

});

Output-



**Practical 23-Write An application to display details of current path.**

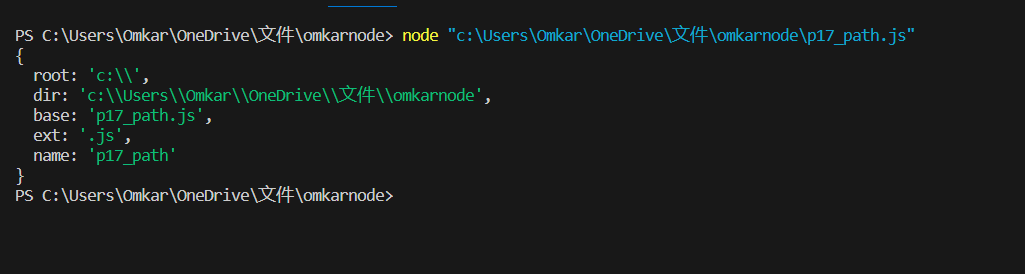
Code-

const location =require("path")

const localobj = location.parse(\_\_filename);

console.log(localobj);

Output-



**Practical 24- Write an application to read file.**

Code-

const fs=require('fs');

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

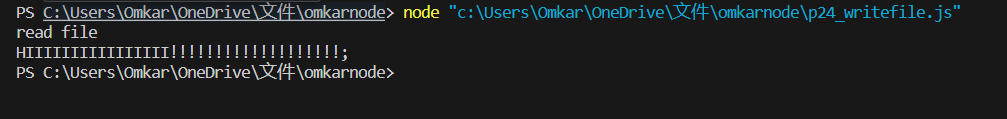
{

console.log("read file");

console.log(data);

});

Output-



**Practical 25- Write an application to write file.**

Code-

const fs = require("fs");

fs.writeFile("welcome.js;",' Welcome Back!!!',function(err,data)

{

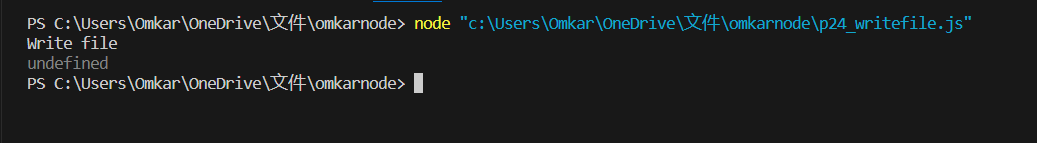
console.log("Write file");

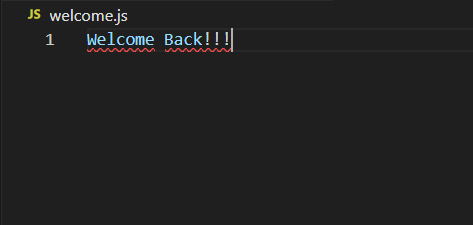
console.log(data);

}

);

**Output-**





**Practical 26- Write an application to Append data in file.**

Code-

const fs=require("fs");

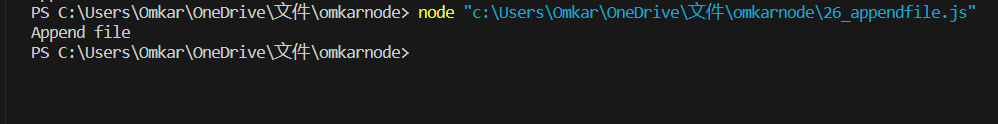
fs.appendFile("welcome.js","\n omkar \n phadke ", function(err,data)

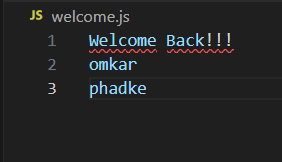
{

console.log("Append file");

});

Output-





**Practical 27- Write An application to delete file.**

Code-

const fs=require("fs");

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

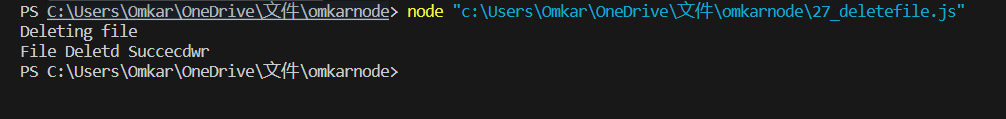
{

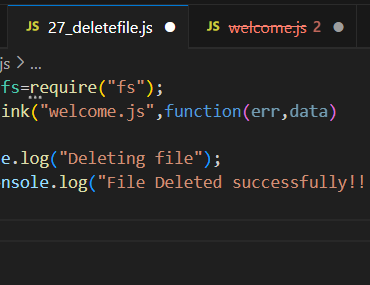
console.log("Deleting file");

console.log("File Deleted successfully!!!!!");

});

Output-





**Practical 28-write an application to rename a file.**

Code-

var fs = require('fs');

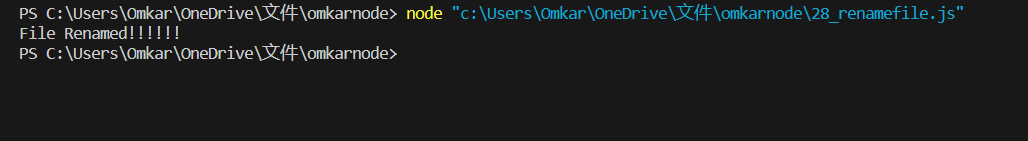
fs.rename('welcome12.js','welcome.js', function (err) {

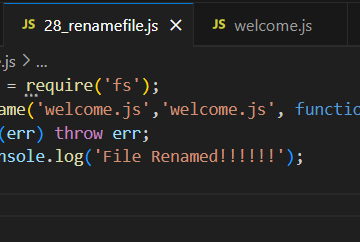
if(err) throw err;

console.log('File Renamed.');

});

Output-







**Practical 29- Combine Read, Write, Append and Delete files in one node js program.**

Code-

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)

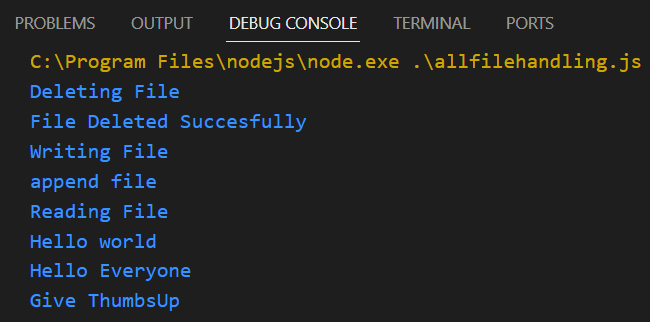
{ console.log("Reading File"); console.log(data);

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

{ console.log("Deleting File");

console.log("File Deleted Succesfully");

});

Output-

**Practical 30- Create an application to create database in nodejs.**

Code-

my=require("mysql");

con=my.createConnection({

host:"localhost",

user:"root",

password:"omkarsql"

});

con.connect(function(err)

{

if (err)throw err;

con.query("create database omkar",function(err)

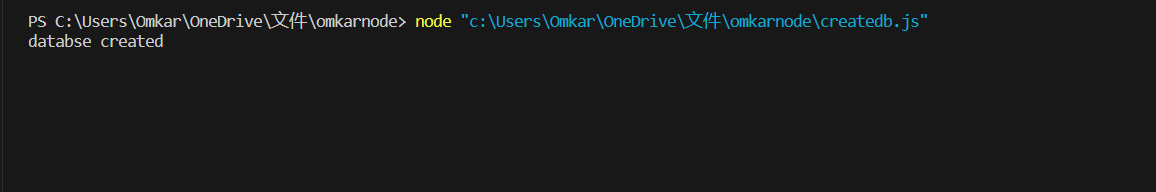
{

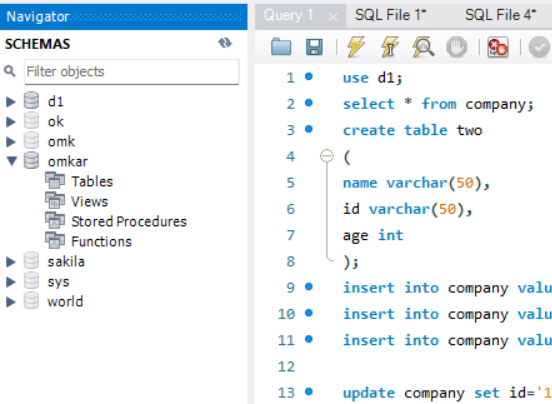
console.log("databse created");

});

});

Output-





**Practical 31- Create an Application to create Student table with columns name, rollno, class, contact in Node.js.**

Code-

my=require("mysql");

con=my.createConnection({

host:"localhost",

user:"root",

password:"omkarsql",

database:"omkar"

});

con.connect(function(err)

{

if(err)throw err

{

con.query("create table student(name varchar(20),roll int ,class varchar(30),contact varchar(10))",function(err)

{

if (err){

console.log(err);

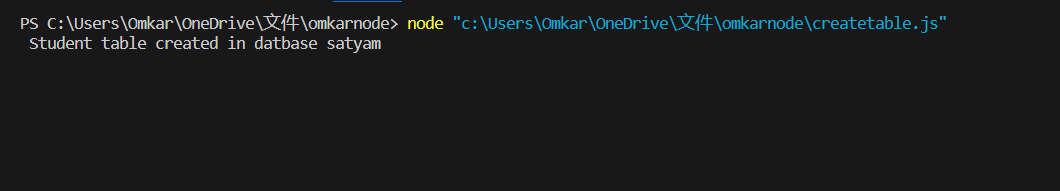
}

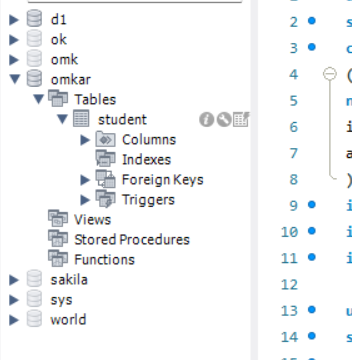
console.log(" Student table created in datbase omkar" );

});

}

Output-





**Practical 32- Create an application to insert rows into student table in Nodejs.**

Code-

my=require("mysql");

con=my.createConnection({

host:"localhost",

user:"root",

password:"omkarsql",

database:"omkar"

});

con.connect(function(err)

{

if(err)throw err

{

con.query("insert into student values('omkar',1oo,'mcaB')",function(err)

{

console.log("data inserted");

});

con.query("insert into student values('satish',1o1,'mcaB')",function(err)

{

console.log("data inserted 2");

});

con.query("insert into student values('phadke',1o2,'mcaB')",function(err)

{

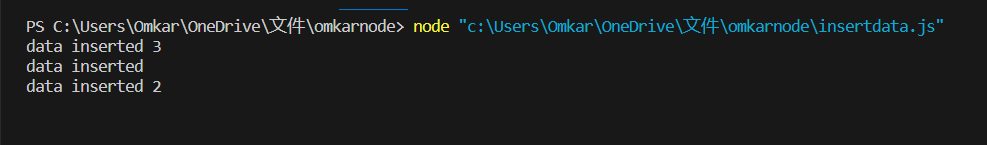
console.log("data inserted 3");

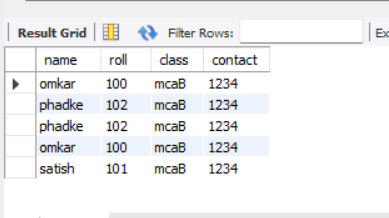
});

}

});

Output-





**Practical 33- Create an application to display rows from student table in Nodejs.**

Code-

my=require("mysql");

con=my.createConnection({

host:"localhost",

user:"root",

password:"omkarsql",

database:"omkar"

});

con.connect(function(err)

{

if(err)throw err

{

con.query("select \* from student",function(err,result)

{

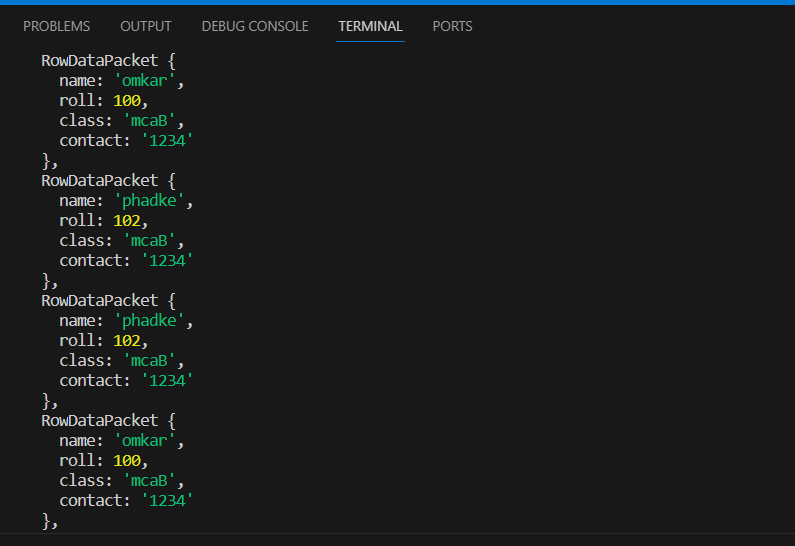
console.log(result);

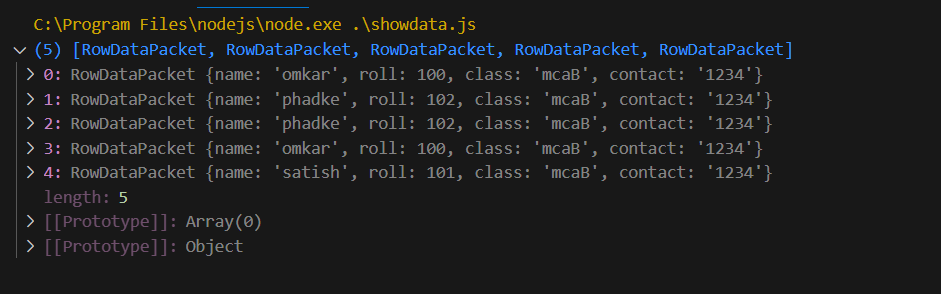
});

}

});

Output-





**Practical 34- Create an application to Update rows into student table in Nodejs.**

Code-

my=require("mysql");

con=my.createConnection({

host:"localhost",

user:"root",

password:"omkarsql",

database:"omkar"

});

con.connect(function(err)

{

if(err)throw err

{

con.query("Update student set roll=59 where name='satyam'",function(err,result)

{

console.log("Data Updated");

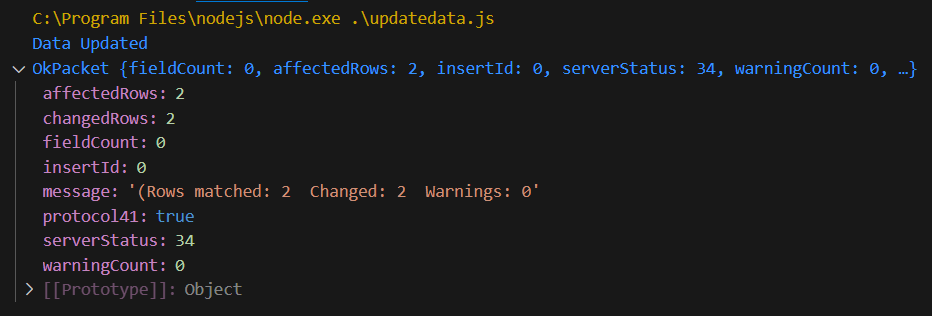
console.log(result);

});

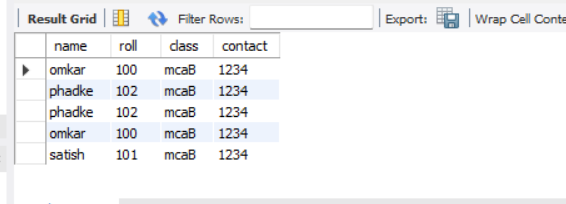
}

});

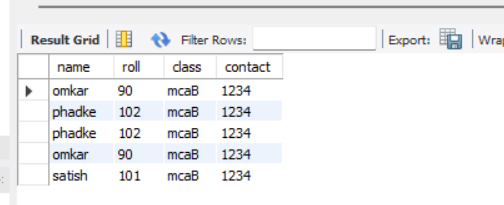
Output-

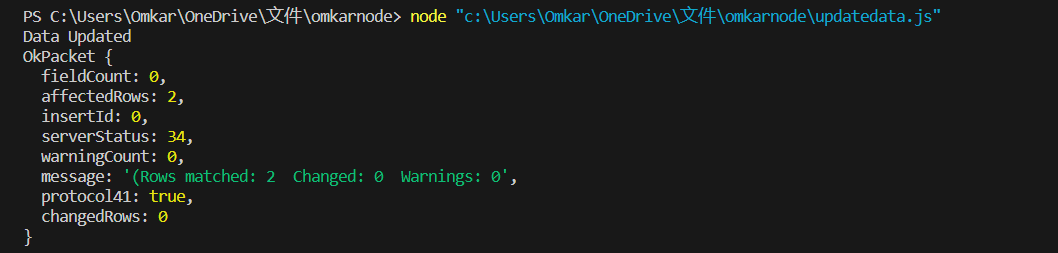


Previous:-



After Updated:-





**Practical 35- Create an application to update mobile number of student name =”omkar” into student table in Nodejs.**

Code-

my=require("mysql");

con=my.createConnection({

host:"localhost",

user:"root",

password:"omkarsql",

database:"omkar"

});

con.connect(function(err)

{

if(err)throw err

{

con.query("Update student set contact=8104809271 where name='omkar'",function(err)

{

console.log("Data Updated");

});

con.query("select \* from student",function(err,result)

{

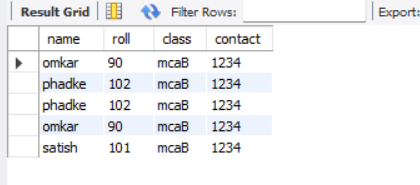
console.log(result);

});

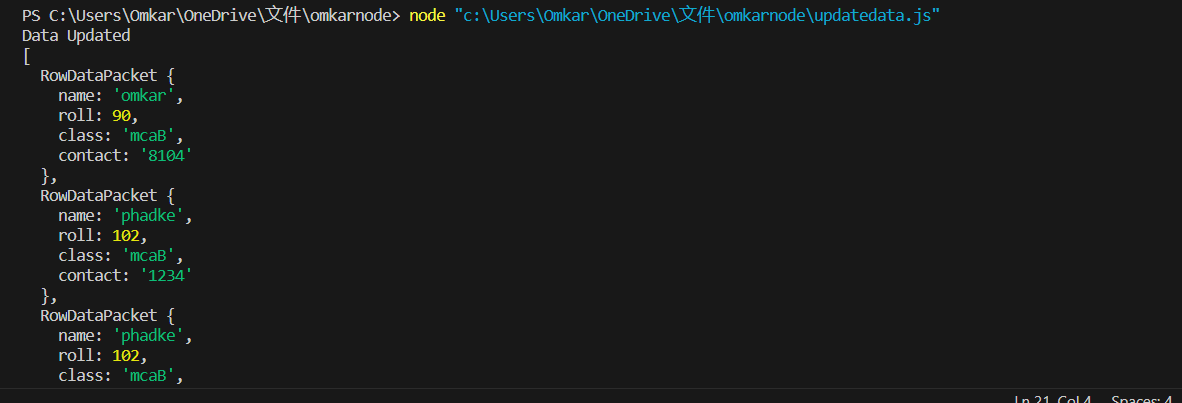
}

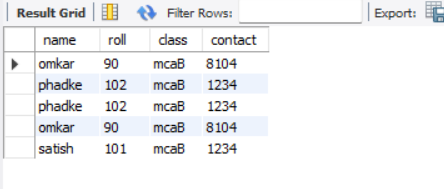
Output-

Before update



After update





**Practical 36- Create an application to add columns into student table in Nodejs**.

Code-

my=require("mysql");

con=my.createConnection({

host:"localhost",

user:"root",

password:"omkarsql",

database:"omkar"

});

con.connect(function(err)

{

if(err)throw err

{

con.query("Alters table student add address varchar(10)",function(err)

{

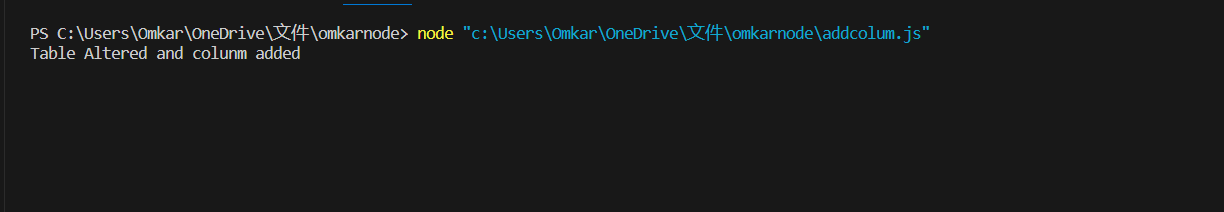
console.log("Table Altered and colunm added");

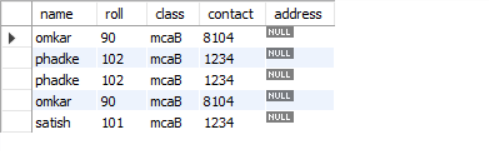
});

}

});

Output-





**Q37- Create an application to Delete record from student table in Nodejs.**

Code-

my=require("mysql");

con=my.createConnection({

host:"localhost",

user:"root",

password:"student",

database:"satyam"

});

con.connect(function(err)

{

if(err)throw err

{

con.query("Delete from student where roll",function(err)

{

console.log("Record delted from table !!!");

});

}

});

Output-

