Program No	21
Roll No	1560
Title	
Objective	

SOURCE CODE -

TodoList.js

```
import React, {useState} from 'react'

function TodoList() {
    const [tasks, setTasks] = useState([]);
    const [input, setInput] = useState('');

    const addTask = () => {
        if(input.trim()) {
            setTasks([...tasks, {text : input, completed: false}]);
            setInput('');
        }
    }

    const toggleTask = (index) => {
        const newTasks = tasks.map((task, i) =>
            i==index ? {...task, completed : !tasks.completed} : task
        );
        setTasks(newTasks);
    }
    const deleteTask = (index) => {
        const newTasks = tasks.filter((_, i) => i !== index);
        setTasks(newTasks);
    };
    return (
        <div className="app">
            <header className="header">
                <h1>To-Do List</h1>
            </header>
            <div className="input-container">
```

```
                <input
                type="text"
                value={input}
                onChange={(e) => setInput(e.target.value)}
                placeholder="Add a new task"
                />
                <button onClick={addTask}>Add</button>
            </div>
            <ul className="task-list">
            {tasks.map((task, index) => (
                <li key={index} className={task.completed ? 'task
completed' :
            'task'}>
                        <span onClick={() =>
toggleTask(index)}>{task.text}</span>
                        <button onClick={() =>
deleteTask(index)}>Delete</button>
                </li>
            ))}
            </ul>
        </div>
        );
    }
    export default TodoList;
```

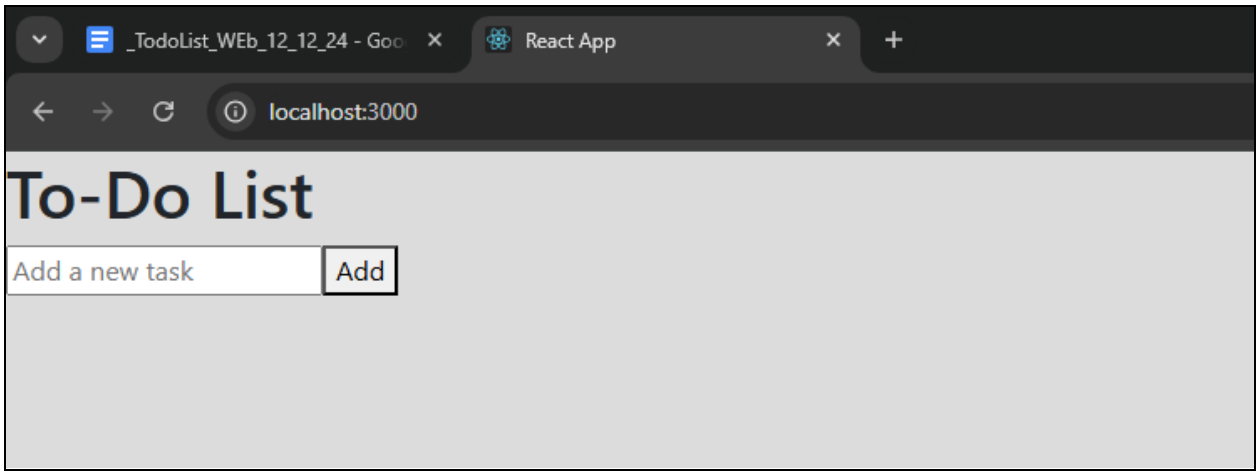
App.js

```
import React from 'react'
import TodoList from './components/TodoList'

function App5() {
  return (
    <div>
        <TodoList />
    </div>
  )
}
export default App5
```

App.css

```
.task-list {
  list-style: none;
  padding: 0;
  }
  .task {
  background-color: white;
  padding: 15px;
  margin-bottom: 10px;
  border-radius: 5px;
  display: flex;
  justify-content: space-between;
  align-items: center;
  box-shadow: 0 0 5px rgba(0, 0, 0, 0.1);
  cursor: pointer;
  }
  .task.completed {
  text-decoration: line-through;
  color: #888;
  }
  .task button {
  background-color: #dc3545;
  border: none;
  color: white;
  padding: 5px 10px;
  border-radius: 5px;
  cursor: pointer;
  }
  .task button:hover {
  background-color: #c82333;
  }
```

OUTPUT -


_TodoList_WEb_12_12_24 - Goo
React App
localhost:3000
To-Do List
Add a new task
Add


_TodoList_WEb_12_12_24 - Goo
React App
localhost:3000
Guest
To-Do List
Add a new task
Add
Study for practical
Delete


_TodoList_WEb_12_12_24 - Goo
React App
localhost:3000
Guest
To-Do List
Add a new task
Add
Study for practical
Delete

Program no	18
Roll no	1545
Title	React hooks
Objective	Creating form and using usestate hook and implementing form validation

Source code:

Form1.js

```
import React, { useState } from 'react'
import '../App.css'
function Form1() {
   const [errors, setErrors] = useState({
      username: '',
      email: '',
      password: '',
      confirmPassword: ''
   });
   const [formData, setFormData] = useState({
      username: '',
      email: '',
      password: '',
      confirmPassword: ''
   })
   const handleChange = e => {
      const { name, value } = e.target;
      setFormData({
         ...formData,
         [name]: value,
      })
   }

   const handleSubmit = e => {
      e.preventDefault();
      const newErrors = validateForm(formData);
      setErrors(newErrors);

      if (Object.keys(newErrors).length === 0) {
         alert('Form submitted successfully!')
      } else {
         alert('Form submission failed due to validation error')
      }
   }

   const validateForm = (data) => {
      const errors = {}
```

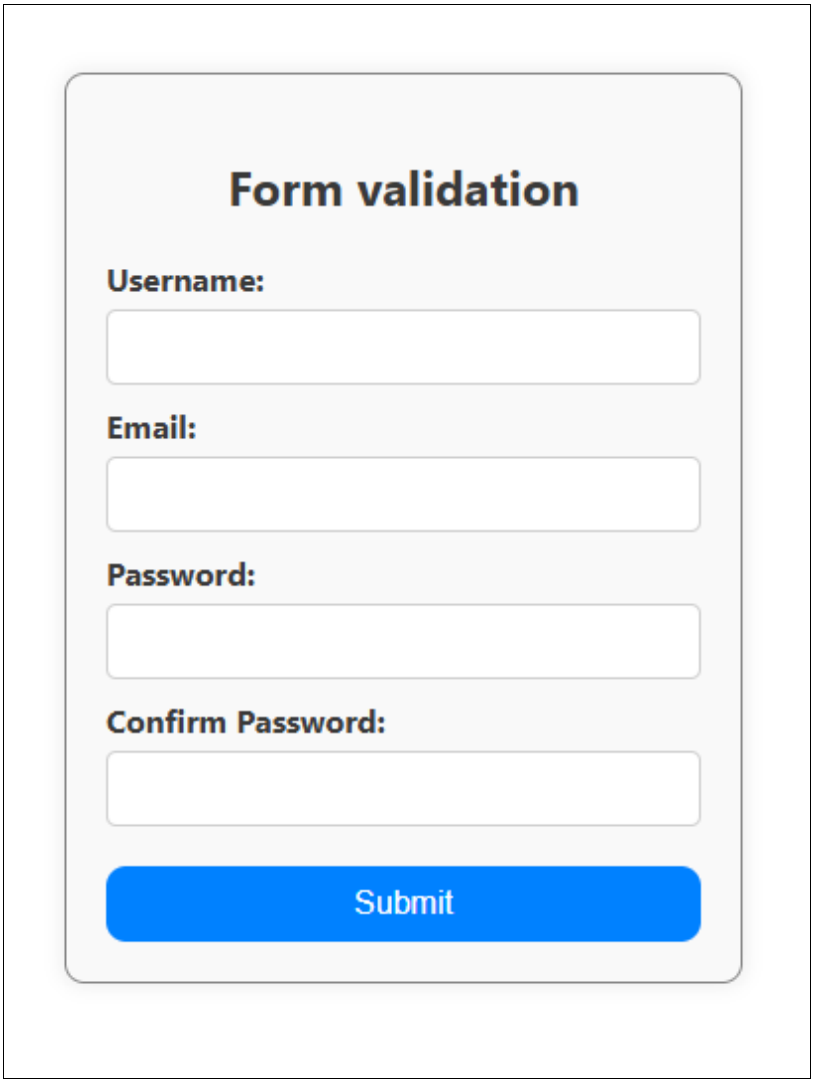
```
    if (!data.username.trim()) {
      errors.username = 'Username is required';
    } else if (!data.username.length < 4) {
      errors.username = "Username must be 4 characters";
    }

    if (!data.email.trim()) {
      errors.email = 'Email is required';
    } else if (!/^[^@]+@[^@]+\.[^@]+$/.test(data.email)) {
      errors.email = "Email is invalid";
    }

    if (!data.password.trim()) {
      errors.password = 'Password is required';
    } else if (!data.password.length < 8) {
      errors.password = "Username must be 8 characters";
    }

    if (!data.confirmPassword.trim()) {
      errors.confirmPassword = 'Confirm password is required';
    } else if (data.password !== data.confirmPassword) {
      errors.confirmPassword = "Password and Confirm Password doesn't match";
    }

    return errors;

  }
  return (
    <div className='form-container'>
      <h2 className='form-title'>Form validation</h2>
      <form onSubmit={handleSubmit}>
        <div>
          <label className='form-label'>Username:</label>
          <input className='form-input' type='text' name='username' value={formData.username}
onChange={handleChange} />
          {errors.username && <p className='error-message'>{errors.username}</p>}
        </div>
        <div>
          <label className='form-label'>Email:</label>
          <input className='form-input' type='text' name='email' value={formData.email}
onChange={handleChange} />
          {errors.username && <p className='error-message'>{errors.username}</p>}
        </div>
        <div>
          <label className='form-label'>Password:</label>
          <input className='form-input' type='password' name='password'
value={formData.password} onChange={handleChange} />
          {errors.username && <p className='error-message'>{errors.username}</p>}
        </div>
```

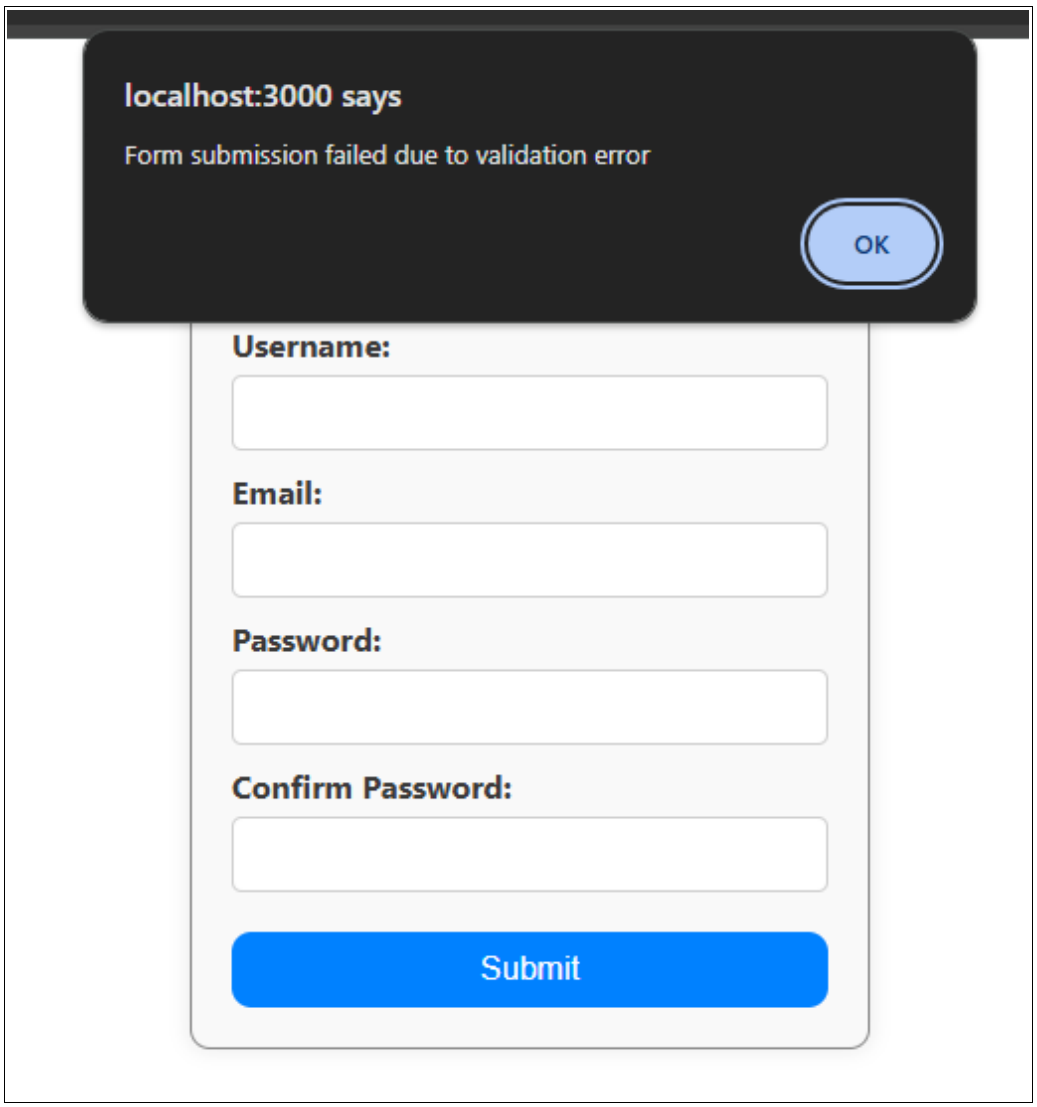
```
            <div>
               <label className='form-label'>Confirm Password:</label>
               <input className='form-input' type='password' name='confirmPassword'
value={formData.confirmPassword} onChange={handleChange} />
               {errors.username && <p className='error-message'>{errors.username}</p>}
            </div>
            <div>
               <input type='submit' value='Submit' className='submit-button' />
            </div>
         </form>
      </div>
   )
}

export default Form1
```

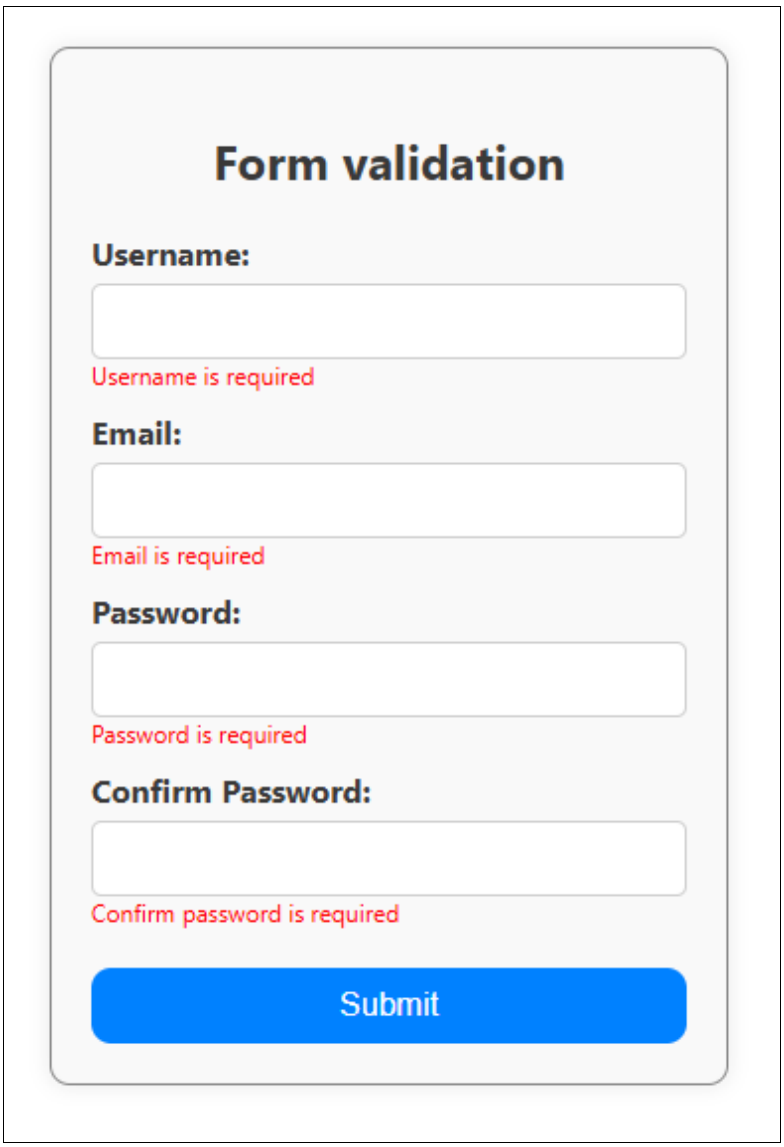
App.js

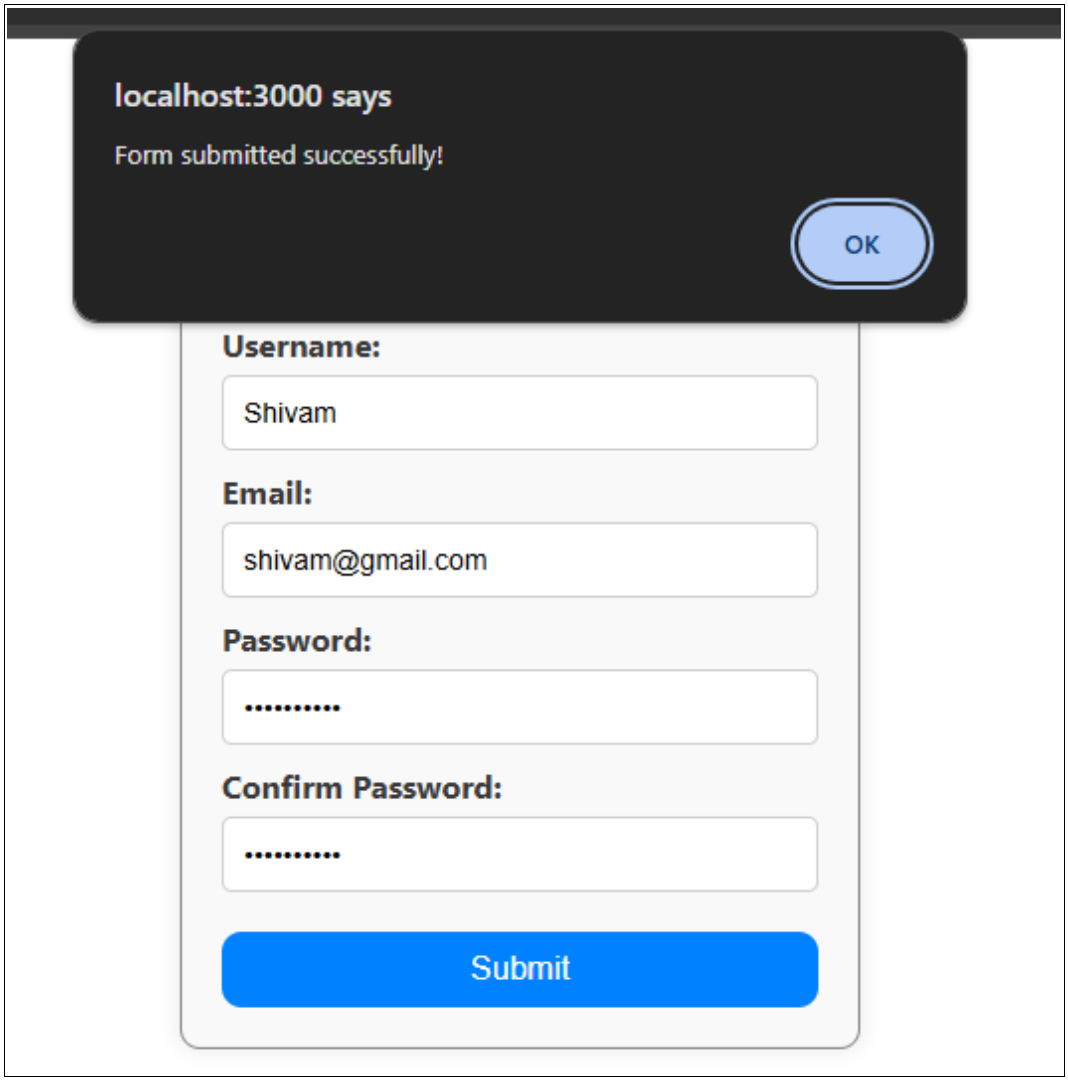
```
import './App.css';
import Form1 from './components/Form1';
function App() {

  return (
    <div className='container'>
      <Form1 />
    </div>
  )
}

export default App;
```

Output:


Form validation
Username:
Email:
Password:
Confirm Password:
Submit


localhost:3000 says
Form submission failed due to validation error
OK
Username:
Email:
Password:
Confirm Password:
Submit


Form validation
Username:
Username is required
Email:
Email is required
Password:
Password is required
Confirm Password:
Confirm password is required
Submit


localhost:3000 says
Form submitted successfully!
OK
Username:
Shivam
Email:
shivam@gmail.com
Password:
Confirm Password:
Submit

Program no	19
Roll no	1545
Title	React router dom
Objective	Single page application

Source code:

index.js

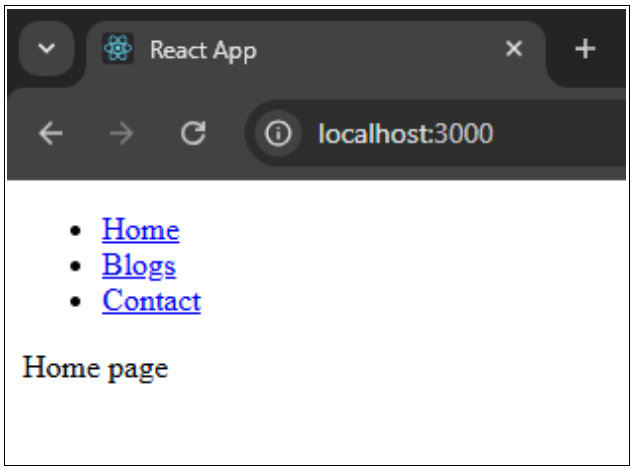
```
import ReactDOM from "react-dom/client";
import { BrowserRouter, Routes, Route } from "react-router-dom";
import Layout from "./pages/Layout";
import Home from "./pages/Home";
import Blogs from "./pages/Blogs";
import Contact from "./pages/Contact";
import NoPage from "./pages/NoPage";

export default function App() {
  return (
    <BrowserRouter>
      <Routes>
        <Route path="/" element={<Layout />}>
          <Route index element={<Home />} />
          <Route path="blogs" element={<Blogs />} />
          <Route path="contact" element={<Contact />} />
          <Route path="*" element={<NoPage />} />
        </Route>
      </Routes>
    </BrowserRouter>
  );
}

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(<App />);
```

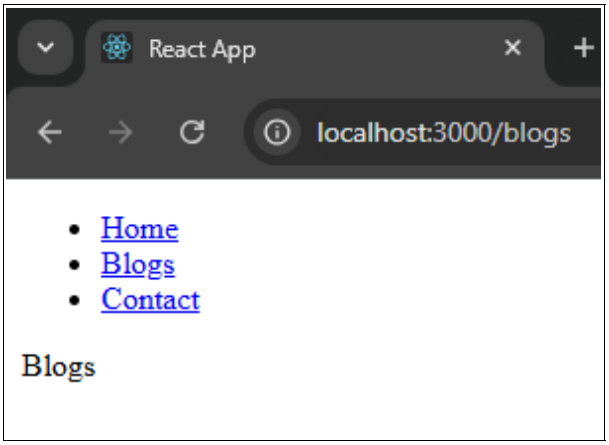
Layout.js

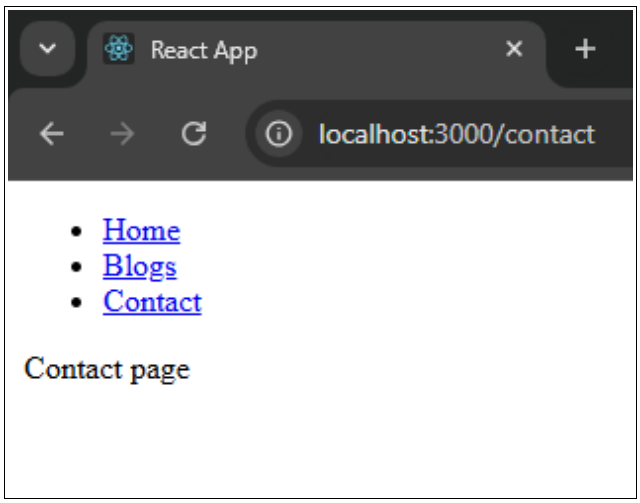
```
import { Outlet, Link } from "react-router-dom";

const Layout = () => {
  return (
    <>
      <nav>
        <ul>
          <li>
            <Link to="/">Home</Link>
          </li>
```

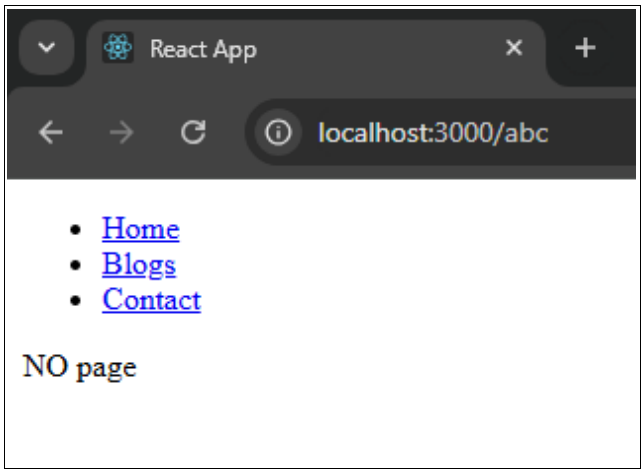
```
            <li>
              <Link to="/blogs">Blogs</Link>
            </li>
            <li>
              <Link to="/contact">Contact</Link>
            </li>
          </ul>
        </nav>

        <Outlet />
      </>
    )
};

export default Layout;
```

Output:






React App
localhost:3000/contact
Home
Blogs
Contact
Contact page


React App
localhost:3000/abc
Home
Blogs
Contact
NO page

Program	22
Roll no	1545
Title	Shopping Cart application
Objective	To design shopping cart using class component and state

Source code:

App.js

```
import './App.css';
import Product from './components/Product';

function App() {
  return <Product />;
}

export default App;
```

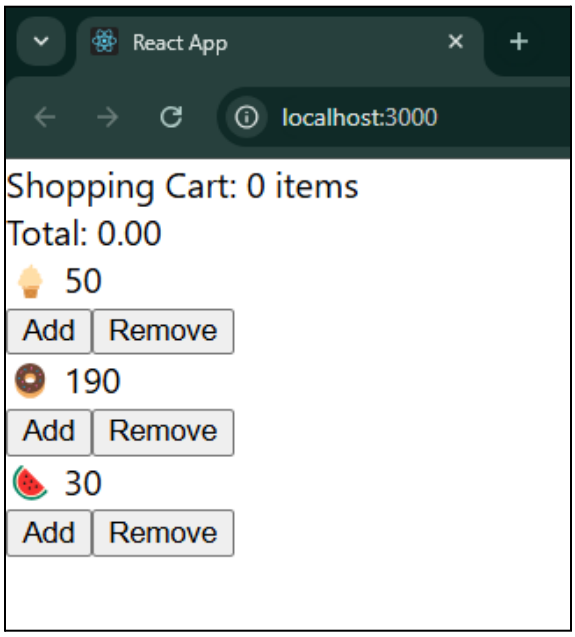
Product.js

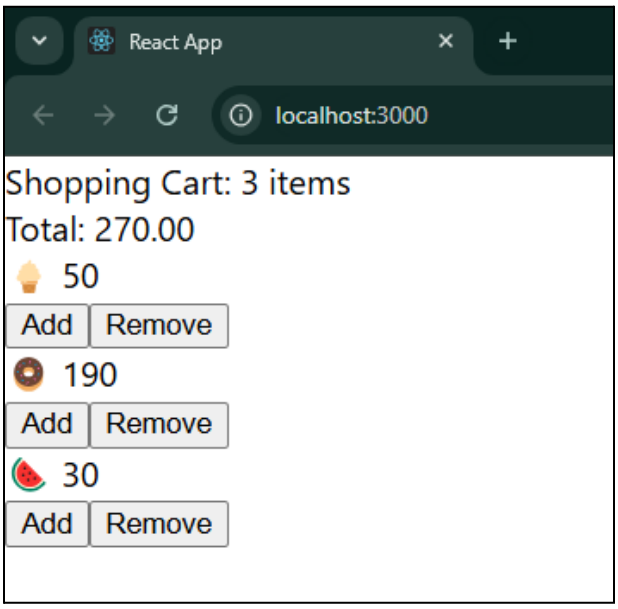
```
import React, { Component } from 'react';

const products = [
  {
    pr: ' 🍦',
    name: 'ice cream',
    price: 50,
  },
  {
    pr: '🍩',
    name: 'donuts',
    price: 190,
  },
  {
    pr: '🍉',
    name: 'watermelon',
    price: 30,
  },
];
```

```
export class Product extends Component {
  state = {
    cart: [],
    total: 0,
  };
  currencyOptions = {
    minimumFractionDigits: 2,
    maximumFractionDigits: 2,
  };
  getTotal = () => {
    return this.state.total.toLocaleString(undefined, this.currencyOptions);
  };
  add = (product) => {
    this.setState((state) => ({
      cart: [...state.cart, product.name],
      total: state.total + product.price,
    }));
  };
  remove = (product) => {
    this.setState((state) => {
      const cart = [...state.cart];
      cart.splice(cart.indexOf(product.name));
      return {
        cart,
        total: state.total - product.price,
      };
    });
  };
  render() {
    return (
      <div>
        <div>Shopping Cart: {this.state.cart.length} items</div>
        <div>Total: {this.getTotal()}</div>
        <div>
          {products.map((product) => (
            <div key={product.name}>
              <div>
                <span role='img' aria-label={product.name}>
                  {product.pr} {product.price}
                </span>
              </div>
              <button onClick={() => this.add(product)}>Add</button>
              <button onClick={() => this.remove(product)}>Remove</button>
            </div>
```

```
          ))}
        </div>
      </div>
    );
  }
}

export default Product;
```

Output:





Program No	11
Roll No	1560
Title	Interval Methods
Objective	Program to defer execution and cancel Execution of function

SOURCE CODE :

```
i = 0;
var inter = setInterval(function(){
  ++i;
  console.log("Hello There : "+i);
  if(i==10) clearInterval(inter);
}, 3000);

process.nextTick(function(){
  console.log("Urgent task");
});
setImmediate(() => {
  console.log("Running before the timeout => number 3");
});


var t = setTimeout(function(){
  console.log("Time out example!");
}, 2000);
clearTimeout(t);

console.log("Program is running!");


console.log("Hello => Number 1");


setTimeout(() => {
  console.log("The timeout running last => number 4");
}, 100);

process.nextTick(() =>{
  console.log("Running at next tick => number 2");
});
```

OUTPUT :

```
PS D:\Rohit\WEB\Program12_16_11> node __callback3.js
Program is running!
Hello => Number 1
Urgent task
Running at next tick => number 2
Running before the timeout => number 3
The timeout running last => number 4
Hello There : 1
Hello There : 2
Hello There : 3
Hello There : 4
Hello There : 5
Hello There : 6
Hello There : 7
Hello There : 8
Hello There : 9
Hello There : 10
PS D:\Rohit\WEB\Program12_16_11>
```

Program No	
Roll No	1560
Title	ProductCard
Objective	

SOURCE CODE -

Data.js

```
const data = [
  {
    pname: "Samsung Galaxy S22 5G",
    price: 72900,
    rating: 4.5
  },
  {
    pname: "SOnePlus Nord 2T 5G",
    price: 33999,
    rating: 4.2
  },
  {
    pname: "Realme C11 2021",
    price: 8900,
    rating: 3.2
  },
];
export default data;
```

ProductCard.js

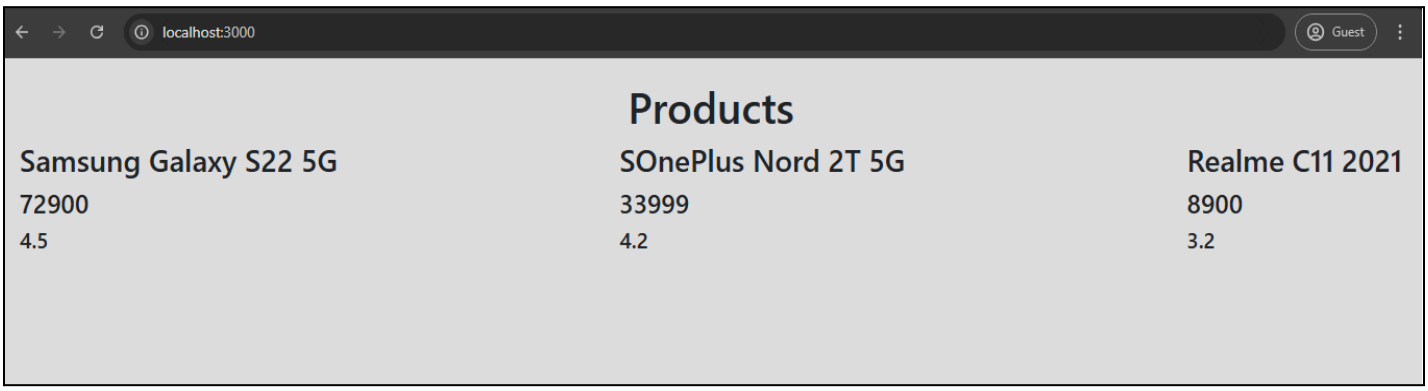
```
import React from "react";
export default function ProductCard({ additional, pname, price, rating })
{
  return (
    <div>
      <h3>{pname}</h3>
      <h4>{price}</h4>
      <h5>{rating}</h5>
    </div>
```

```
  );
}
```

App.js

```
import "./App.css";
import ProductCard from "./components/ProductCard";
import data from "./components/Data";
function App() {
  return (
    <div className="wrapper">
      <h1>Products</h1>
      {data.map((product) => (
        <ProductCard
          key={product.pname}
          pname={product.pname}
          price={product.price}
          rating={product.rating}
        />
      ))}
    </div>
  );
}
export default App;
```

OUTPUT -



Program No	2
Roll No	1560
Title of Program	Modules in NodeJs
Objective	Create a module with following functions A. To find whether given string is palindrome or not B. To find whether given number is Armstrong C. To find sum of digits of a given number

Source Code :

lab2node_module.js

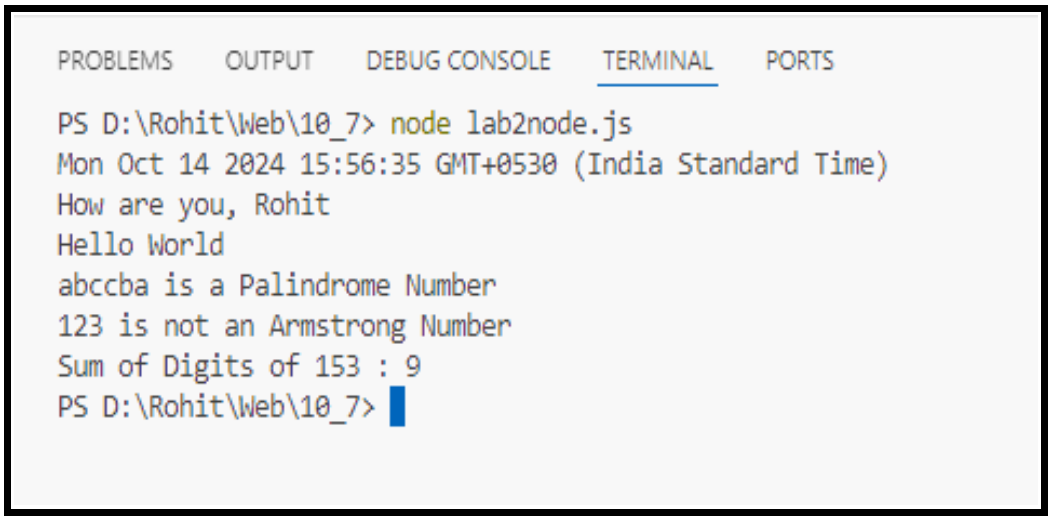
```
exports.myDateTime = function(){
   return Date();
}

exports.myMessage = function(){
   return "Hello World";
}

exports.greetingFriend = function(name){
   return "How are you, "+name;
}

exports.palindrome = function(str){
   var rev = "";
   var str1 = str.toString();
   for(i=str1.length-1;i>=0;i--){
      rev = rev + str1.charAt(i);
   }
   if(rev==str1)
      return str1+" is a Palindrome Number";
   else
      return str1+" is not a Palindrome Number"  ;
}

exports.armstrong = function(num){
   let sum = 0;
   let temp = num;
   while(temp>0){
      sum = sum + Math.pow(temp%10, 3);
      temp = parseInt(temp/10);
   }
   if(sum==num)
```

```
        return num+" is an Armstrong Number";
    else
        return num+" is not an Armstrong Number";
}

exports.sumOfDigits = function(num){
    let sum = 0;
    while(num>0){
        sum += num%10;
        num = parseInt(num/10);
    }
    return sum;
}
```

lab2node.js

```
var uselab2 = require('./lab2node_module.js');
console.log(uselab2.myDateTime());
console.log(uselab2.greetingFriend("Rohit"));
console.log(uselab2.myMessage());

var str = "abccba"
console.log(uselab2.palindrome(str));

var num = 153;
console.log(uselab2.armstrong(123));
console.log("Sum of Digits of "+sum+" : "+uselab2.sumOfDigits(num));
```

Output :



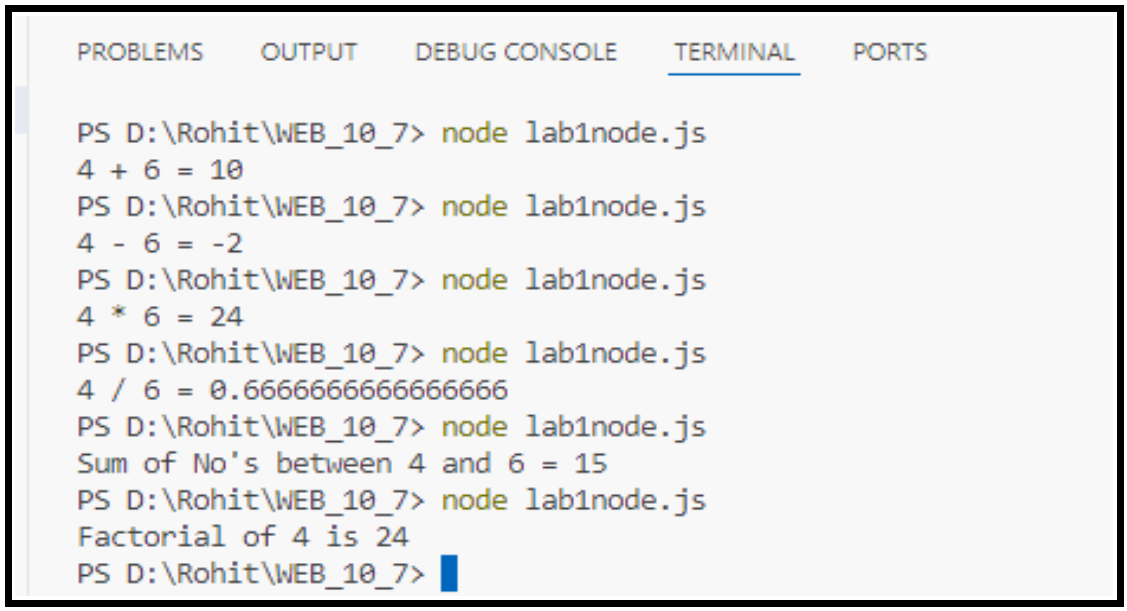
Program No	1
Roll No	1560
Topic	Nodejs
Title of Program	Program to function in node js
Objective	Basics of node js B1 1. To understand basic of Node.js using functions A. Function in Node js to sum two values B. Function in Node js to find sum of numbers between two values C. Function in Node js to find factorial of a number D. Function in Node js to mimic functions of calculator ADD,SUBTRACT,etc

Source Code :

```
function add(x, y){
  return x+y;
}

function sub(x, y){
  return x-y;
}

function mul(x, y){
  return x*y;
}

function div(x, y){
  return x/y;
}

function sumbtxy(x, y){
  sum = 0;
  for(i=x; i<=y; i++) sum = sum + i;
  return sum;
}

function fact(n){
  fact = 1;
  if(n==0 || n==1) return fact;
  else{
    for(i=2; i<=n; i++) fact = fact * i;
    return fact;
  }
```

```
}

num1 = 4;
num2 = 6;
choice = 6;
switch(choice){
  case 1 :
    console.log(num1+" + "+num2+" = "+add(num1, num2));
    break;
  case 2 :
    console.log(num1+" - "+num2+" = "+sub(num1, num2));
    break;
  case 3 :
    console.log(num1+" * "+num2+" = "+mul(num1, num2));
    break;
  case 4 :
    console.log(num1+" / "+num2+" = "+div(num1, num2));
    break;
  case 5 :
    console.log("Sum of No's between "+num1+" and "+num2+" = "+sumbtxy(num1, num2));
    break;
  case 6 :
    console.log("Factorial of "+num1+" is "+fact (num1));
}
```

Output :

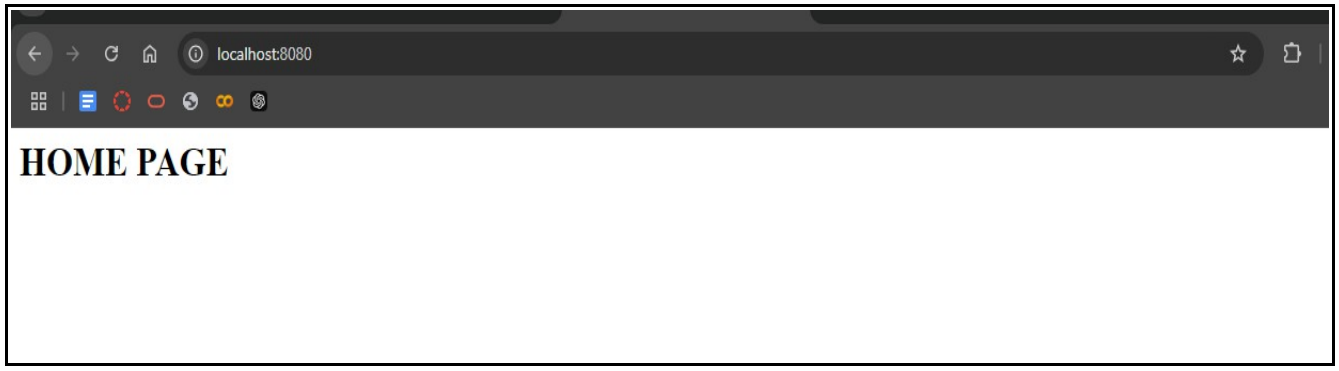


Program No	3A
Roll No	1560
Title	Create HTTP server.
Objective	Create HTTP server using http module.

Code :

```
var h = require('http');
h.createServer(function(req, res){
  res.writeHead(200,{'Content-Type':'text/html'});
  if(req.url=='/'){
    res.write('<html><body><h1>HOME PAGE</h1></body></html>');
  }
  if(req.url=='/student'){
    res.write('<html><body><h1>STUDENT PAGE</h1></body></html>');
  }
  res.end();
}).listen(8080);
```

OUTPUT :

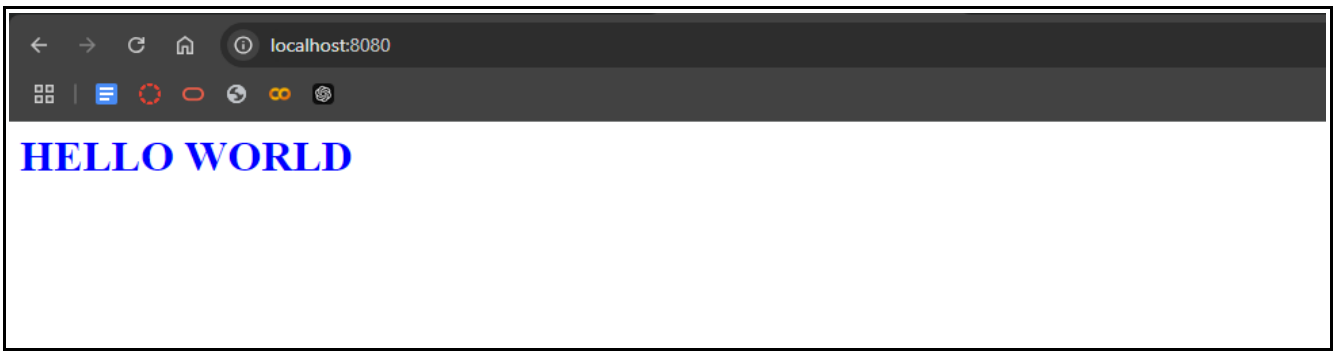


Program No	3B
Roll No	1560
Title	Using upper-case and http module
Objective	Using upper-case and http module to print text in uppercase.

Source Code :

```
var http = require('http');
// var uc = require('upper-case');
import('upper-case').then(uc =>{
  http.createServer(function(req, res){
    res.writeHead(200,{'Content-Type':'text/html'});
    res.write(uc.upperCase("<h1 style='color:blue'>Hello World</h1>"));
    res.end();
  }).listen(8080);
})
```

OUTPUT :



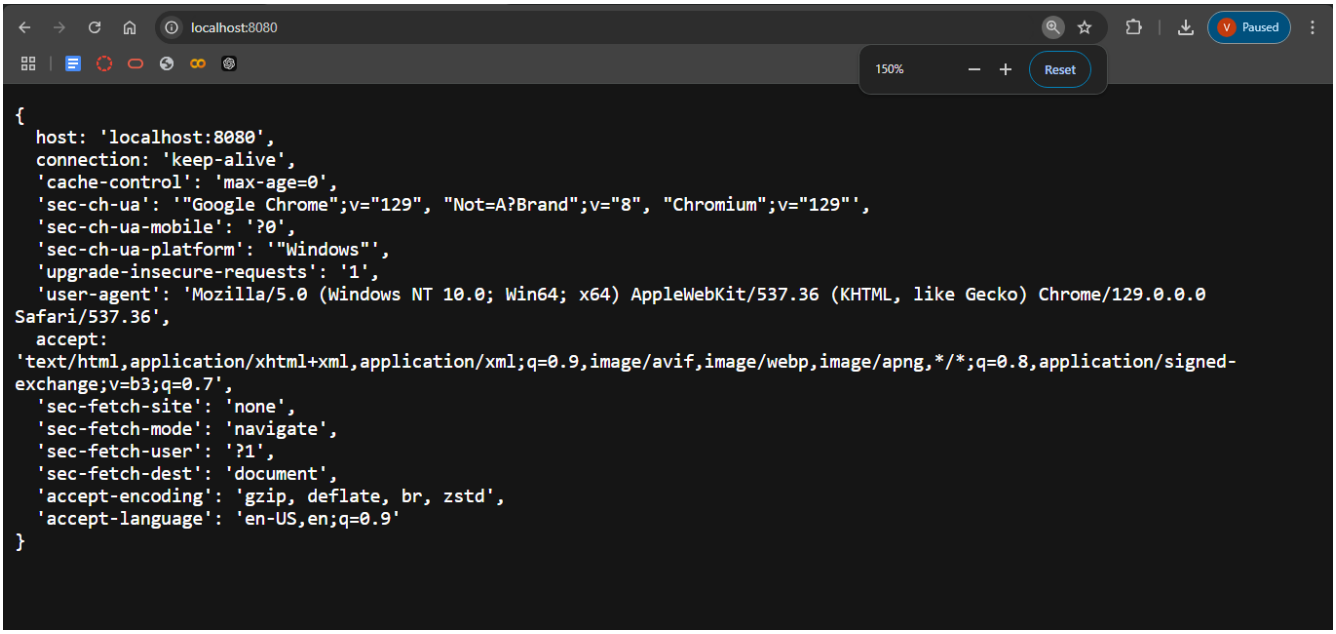
Program No	4
Roll No	1560
Title	To create http server and perform following operations
Objective	A Inspect http hdeader B Open file and print it on browser C Read CSV file and print on browser D Read pdf file and display on browser

SOURCE CODE :

Inspect http headers -

```
var h = require('http');
var u = require('util');

h.createServer(function(req, res){
  res.writeHead(200, {'Content-Type' : 'text/plain'});
  res.write(u.inspect(req.headers));
  res.end();
}).listen(8080);
```

OUTPUT :

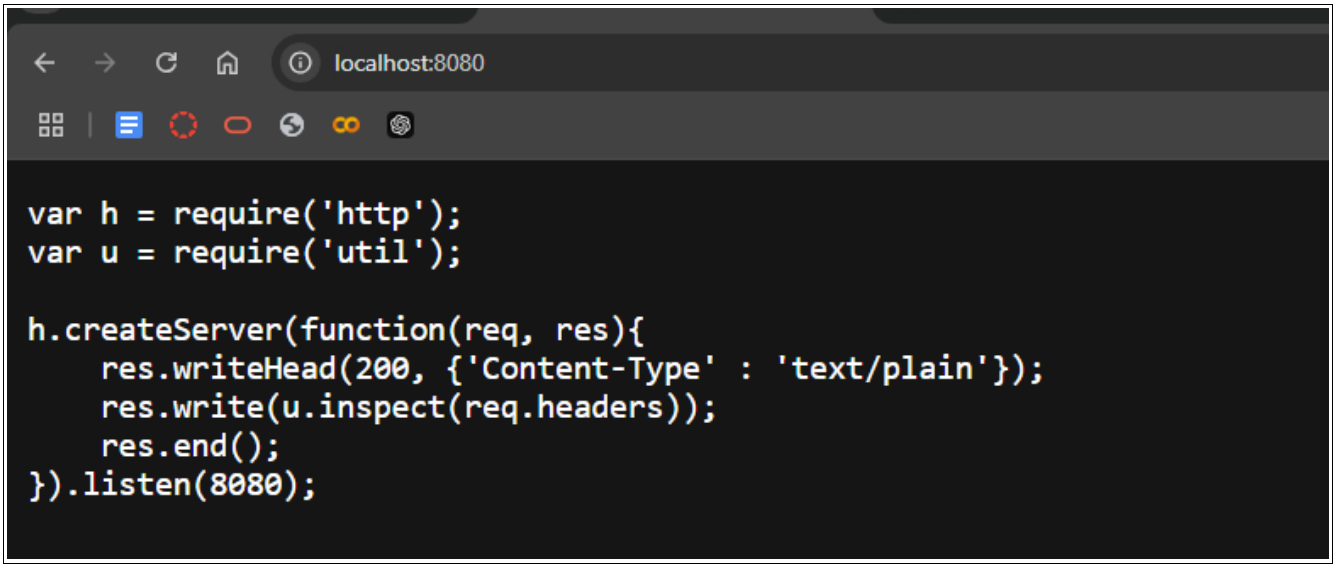


Open file and print it on browser -

```
var h = require('http')
var f = require('fs')
```

```
h.createServer(function(req, res){
  res.writeHead(200, {'Content-Type' : 'text/plain'});
  var rfs = f.createReadStream('httpserver2.js');
  rfs.pipe(res);
}).listen(8080);
```

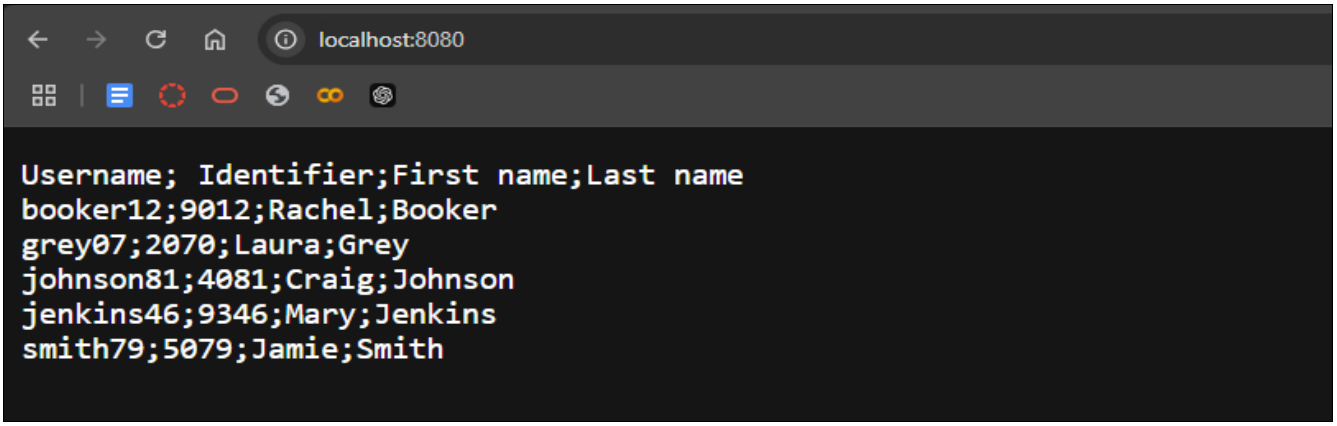
OUTPUT



C) Open CSV file and print it on browser

```
var h = require('http')
var f = require('fs')

h.createServer(function(req, res){
  res.writeHead(200, {'Content-Type' : 'text/plain'});
  var rfs = f.createReadStream('username.csv');
  rfs.pipe(res);
}).listen(8080);
```

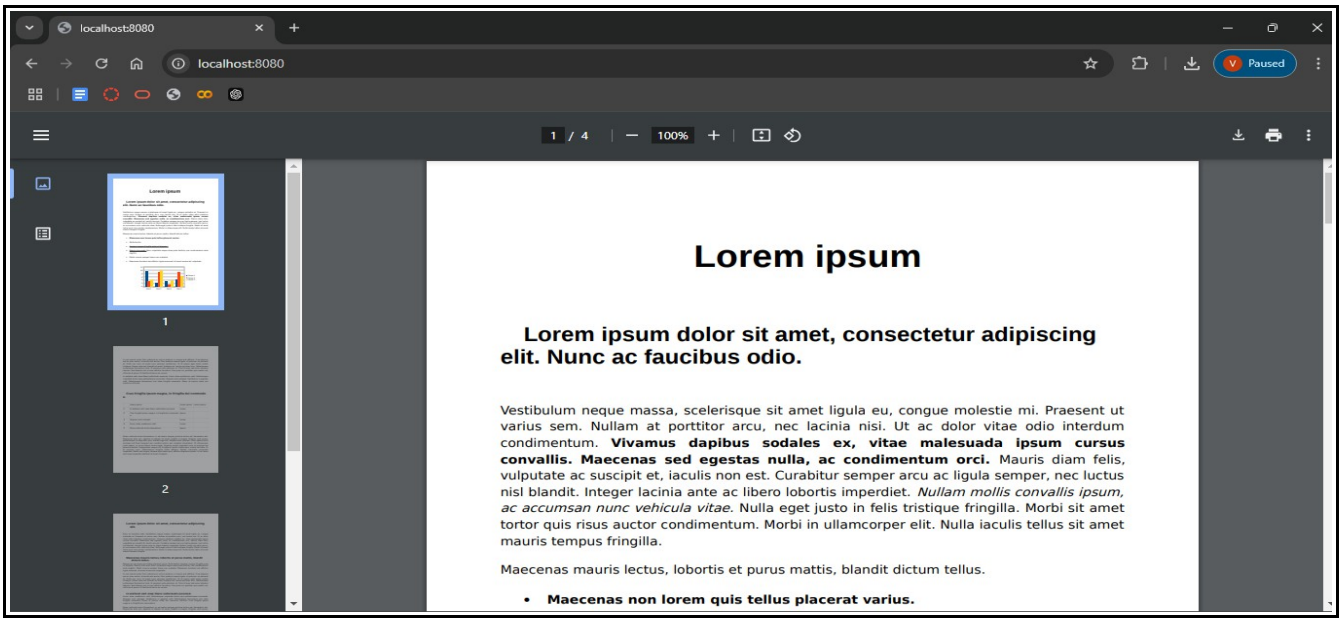
OUTPUT



D)Read Pdf file and display it on browser -

```
var h = require('http')
var fs = require('fs')

fs.readFile('./summer.html', function(err, html){
  if(err) throw err;

  h.createServer(function(req, res){
    res.writeHead(200, {'Content-Type' : 'text/html'});
    res.write(html);
    res.end();
  }).listen(8080);
});
```

OUTPUT -



Program No	5
Roll No	1560
Title	To demonstrate use of callback function
Objective	A)Program to use callback function for Asynchronous File Read B) Program using setTimeout to demonstrate callback function -

A) Program to use callback function for Asynchronous File Read -

```
var fs = require('fs');

fs.readFile('input.txt', function(ferr, data){
  if(ferr) return console.error(ferr);

  console.log(data.toString());
  console.log("End of Read");
});
console.log("Reading File!");
x = 5;
y = 5;
console.log("Sum : "+(x+y));
```

```
PS D:\Rohit\WEB\Program4_21_10> node callback.js
Reading File!
Sum : 10
This is an example file used to test the callback program.
End of Read
PS D:\Rohit\WEB\Program4_21_10>
```

B) Program using setTimeout to demonstrate callback function -

```
var calculate = function(x, y, display){
  sum = x+y;
  setTimeout(function(){
    display(sum);
  }, 20000);

  console.log("Leaving...");
}

calculate(5, 10, function(res){
  console.log("Sum : "+res);
  console.log("Program Ends");
```

```
});

console.log("Doing Something Else...");
console.log("Please Wait...");
```

```
PS D:\Rohit\WEB\Program4_21_10> node callback2.js
Leaving...
Doing Something Else...
Please Wait...
Sum : 15
Program Ends
PS D:\Rohit\WEB\Program4_21_10>
```

Program No	6
Roll No	1560
Title	Events and event emitters
Objective	**A**. Write a node js program to handle events using Event Emitter class **B**. Write a program using Event Emitter class to emit events for Sorting an Array, To find sum of odd elements of Array , to search element in the Array.

SOURCE CODE - (A)

```
var e = require('events');
var event = new e.EventEmitter();

function CreateFile(){
  console.log("Creating a File");
}
function Read(){
  console.log("Reading a File");
}
function Write(){
  console.log("Writing a File");
}

event.addListener('IOoperations', Write);
event.addListener('IOoperations', Read);
event.once('CreateIO', CreateFile);
console.log(event.listenerCount('CreateIO'));
event.emit('CreateIO');
event.emit('IOoperations');
event.emit('IOoperations');
console.log(event.rawListeners('IOoperations'));
console.log(event.listenerCount('IOoperations'));
event.removeListener('IOoperations', Read);
event.emit('IOoperations');
console.log(event.listenerCount('IOoperations'));
event.emit('CreateIO');
console.log(event.listenerCount('CreateIO'));
```

OUTPUT - (A)

```
PS D:\Rohit\WEB\Program5_26_10> node event1.js
1
Creating a File
Writing a File
Reading a File
Writing a File
Reading a File
[ [Function: Write], [Function: Read] ]
2
Writing a File
1
0
PS D:\Rohit\WEB\Program5_26_10>
```

SOURCE CODE – (B)

```
var e = require('events');
var evmt = new e.EventEmitter();
var A = [7,2,5,4,3,6];
var st = ['a','v','e','d'];

function sortarray(){
  console.log("Array before Sorting : ");
  console.log(st);
  st.sort();
  console.log("Array After Sorting : ");
  console.log(st);
}
function sumofevenelements(){
  var sum = 0;
  for(i=0;i<A.length;i++){
    if(i%2!=0) sum += A[i];
  }
  console.log("Sum of Even elements in array is "+sum);
}
function sumofoddelements(){
  var sum = 0;
  for(i=0;i<A.length;i++){
    if(i%2==0) sum += A[i];
  }
  console.log("Sum of odd elements in array is "+sum);
}

var searcharray = function(ch){
  var flag = false;
  for(i=0;i<st.length;i++){
    if(st[i]==ch){
      flag = true;
      break;
    }
  }
  if(flag)
    console.log(ch + " is found");
  else
    console.log(ch + " is not found");
}

evmt.addListener('sort', sortarray);
evmt.on('sumeven', sumofevenelements);
evmt.on('sumodd', sumofoddelements);
evmt.on('search', searcharray);
evmt.emit('sort');
```

```
evmt.emit('sumodd');
evmt.emit('sumeven');
evmt.emit('search', 'e');
```

OUTPUT – (B)

```
PS D:\Rohit\WEB\Program5_26_10> node .\eventsortarray.js
Array before Sorting :
[ 'a', 'v', 'e', 'd' ]
Array After Sorting :
[ 'a', 'd', 'e', 'v' ]
Sum of odd elements in array is 15
Sum of Even elements in array is 12
e is found
PS D:\Rohit\WEB\Program5_26_10>
```

Program No	8
Roll No	1560
Title	Event emitter with parameterized and arrow functions
Objective	**A**. Customer information passed as argument to traditional function attached to event emitter **B**. Salary calculation arrow/lambda function

SOURCE CODE - (A)

```
var e = require('events'); var event = new
e.EventEmitter(); var customerInfo = function(code,
cname, city, emailid){   console.log("Customer No : " +
code);   console.log("Customer Name :" + cname);
console.log("City : " + city);   console.log("Email Id
: " + emailid);
}

event.on('customerInfo', customerInfo); event.emit('customerInfo', 1,
'Rohit', 'Vasai', 'mca24_1560@met.edu');
```

OUTPUT - (A)

```
PS D:\Rohit\WEB\Program5_26_10> node .\eventhandlingcustomer2.js
Customer No : 1
Customer Name :Rohit
City : Vasai
Email Id : mca24_1560@met.edu
PS D:\Rohit\WEB\Program5_26_10>
```

SOURCE CODE – (B)

```
var e = require('events'); var
event = new e.EventEmitter();

event.on('calculateSalary', (basic) => {
var hra = basic * 60/100;   var da = basic;
var pt = 200;   var it = basic * 30/100;
var gross = basic + hra + da + pt + it;
var net = gross - (pt+it);
console.log(`Basic Salary : ${basic}`);
console.log(`Dearness Allowance : ${da}`);
console.log(`HRA : ${hra}`);
console.log(`Profession Tax : ${pt}`);
```

```
console.log(`Income Tax : ${it}`);
console.log(`Gross Salary : ${gross}`);
console.log(`Net Salary : ${net}`);
}); event.emit('calculateSalary',
50000);
```

OUTPUT – (B)

```
PS D:\Rohit\WEB\Program5_26_10> node .\eventhandlingsalary.js
Basic Salary : 50000
Dearness Allowance : 50000
HRA : 30000
Profession Tax : 200
Income Tax : 15000
Gross Salary : 145200
Net Salary : 130000
PS D:\Rohit\WEB\Program5_26_10>
```

Program No	9
Roll No	1560
Title	File Operations
Objective	To Perform following file operations : 1) File Read 2) File Write 3) File Append 4) File Delete 5) Rename

SOURCE CODE :

READ & WRITE -

```
var fs = require("fs");

// fs.writeFileSync('testfile.txt', "Hello World, Good Morning!"), function(ferr){
//     console.log("Write File Successful...");
//     if(ferr) return console.error(ferr);
//     else console.log("Write File Successful...");
// };

fs.writeFileSync('testfile.txt', "Hello World, Good Morning!");
console.log("Write File Successful...");

fs.readFile('testfile.txt', function(ferr, filedata){
  if(ferr) return console.error(ferr);
  else{
    console.log(filedata.toString());
    console.log("Reading File...");
  }
})
```

```
PS D:\Rohit\WEB\Program9_11_09> node filewrite1.js
Write File Successful...
Hello World, Good Morning!
Reading File...
PS D:\Rohit\WEB\Program9_11_09>
```

APPEND -

```
var f = require('fs');
data = f.readFileSync("testFile.txt", 'utf-8');
console.log("Before Appending : " + data.toString());
f.appendFileSync("testFile.txt", "Appending New Data");
console.log("Appending Complete");
f.readFile("testFile.txt", function(err, data){
```

```
  if(err) console.log(err);
  else console.log("\nAfter Appending : " + data.toString());
});
```

```
PS D:\Rohit\WEB\Program9_11_09> node .\filereadappend.js
Before Appending : Hello World, Good Morning!
Appending Complete

After Appending : Hello World, Good Morning!Appending New Data
PS D:\Rohit\WEB\Program9_11_09>
```

DELETE -

```
var f=require('fs');
f.unlink("testfile.txt", function(){
  console.log("Delete operation Completed!");
});

f.readFile("testfile.txt", "utf-8", function(error, data){
  if(error) console.log(error);
  else console.log(data);
});
```

```
PS D:\Rohit\WEB\Program9_11_09> node .\filedelete.js
Delete operation Completed!
Hello World, Good Morning!Appending New Data
PS D:\Rohit\WEB\Program9_11_09>
```

RENAME -

```
var f = require("fs");
f.mkdirSync('./test');
f.writeFileSync("testFile.txt", "Welcome to files");
f.renameSync('testFile.txt', 'newFile.txt');
f.readFile('newfile.txt', function(err, data){
  if(err) return console.error(err);
  else{
    console.log("Reading File");
    console.log(data.toString());
  }
});
```

```
PS D:\Rohit\WEB\Program9_11_09> node .\filereadwrite.js
Reading File
Welcome to files
PS D:\Rohit\WEB\Program9_11_09>
```

Program No	10
Roll No	1560
Title	Connecting to mySQL and performing various operatins on table
Objective	Perform following operations on table : 1) Connect 2) Create Database 3) Create Table 4) Insert values

Create database :

```
var mysql = require('mysql2');
var con = mysql.createConnection({
  host : "localhost",
  user : "root",
  password : "root",
  database : "student",
  port : 3308
});
con.connect(function(err){
  if(err) throw err;
  console.log("Connected to mySql");
});
con.query("CREATE DATABASE faculty", function(err){
  if(err) throw err;
  console.log("Faculty Database Created!");
});
```

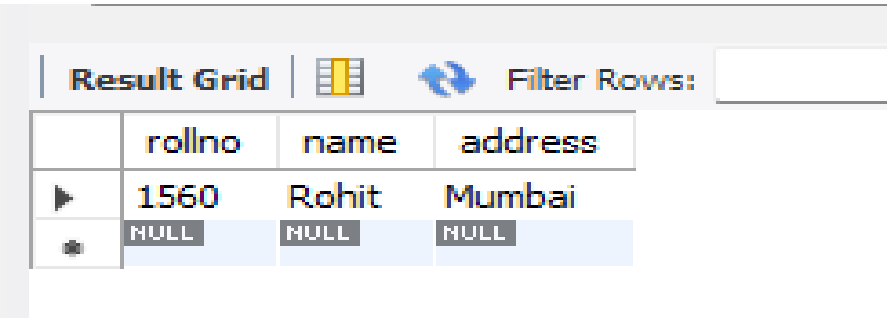
```
PS D:\Rohit\WEB\Program10_11_11> node .\dbconnect1.js
Connected to mySql
Faculty Database Created!
```

Create Table & Insert a Row :

```
var mysql = require('mysql2');
var con = mysql.createConnection({
  host : "localhost",
  user : "root",
  password : "root",
  database : "student",
  port : 3309
});
con.connect(function(err){
```

```
  if(err) throw err;
  console.log("Connected to mySql");
});

con.query("CREATE TABLE student(rollno int primary key, name VARCHAR(255), address
VARCHAR(255))", function(err){
  if(err) throw err;
  console.log("Student Table Created!");
});

con.query("INSERT INTO student VALUES(1535, 'Rohit', 'Mumbai')", function(err){
  if(err) throw err;
  console.log("Row Inserted");
});
```

```
PS D:\Rohit\WEB\Program10_11_11> node .\dbcreate1.js
Connected to mySql
Student Table Created!
Row Inserted
```



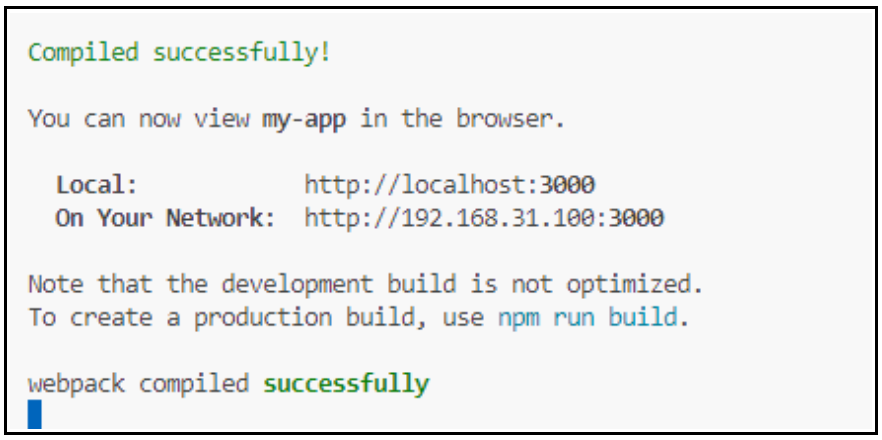
Program No	12
Roll No	1560
Title	React
Objective	To print “Hello Word” using ReactJs

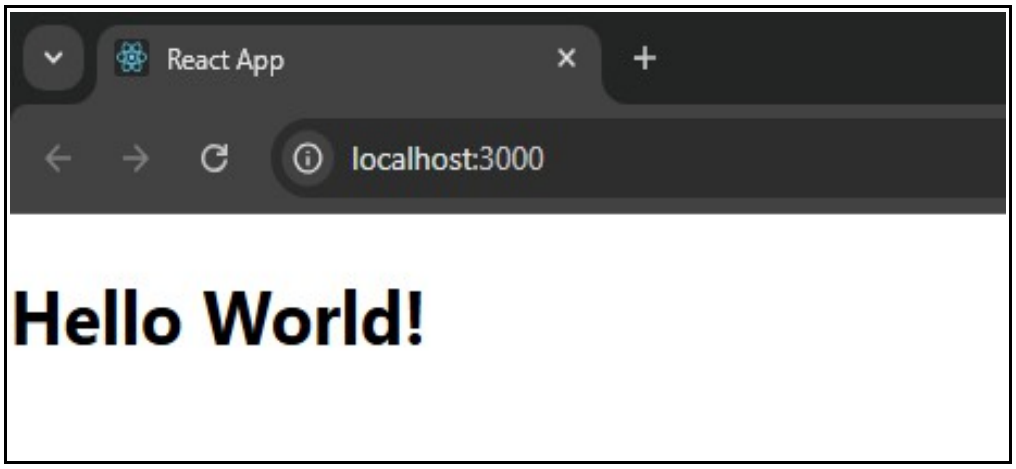
SOURCE CODE :

```
import React from 'react';
import './App.css';

function App() {
 return (
  <>
  <h1>Hello World!</h1>
  </>
 );
}
export default App;
```

OUTPUT :

Step-1 : npx create-react-app my-app
step 2: cd my-app
step 3 : npm start




React App
localhost:3000
Hello World!

Program No	13
Roll No	1560
Title	Exploring JSX
Objective	Adding multiple lines in return. Creating new App. Add new style in App.css. Making use of variables.

SOURCE CODE :

App.js

```
import React from 'react';
import './App.css';

function App() {
  return (
    <div>
      <h1>Hello World!</h1>
      <p>Welcome to MET</p>
      <hr />
      <table border={2}>
        <tbody>
          <tr>
            <th>Roll No</th>
            <th>Name</th>
          </tr>
          <tr>
            <td>1560</td>
            <td>Rohit</td>
          </tr>
          <tr>
            <td>1561</td>
            <td>Yash</td>
          </tr>
          <tr>
            <td>1563</td>
            <td>Siddharth</td>
          </tr>
          <tr>
            <td>1538</td>
            <td>Ritesh</td>
          </tr>
          <tr>
            <td>1545</td>
            <td>Shivam</td>
          </tr>
```

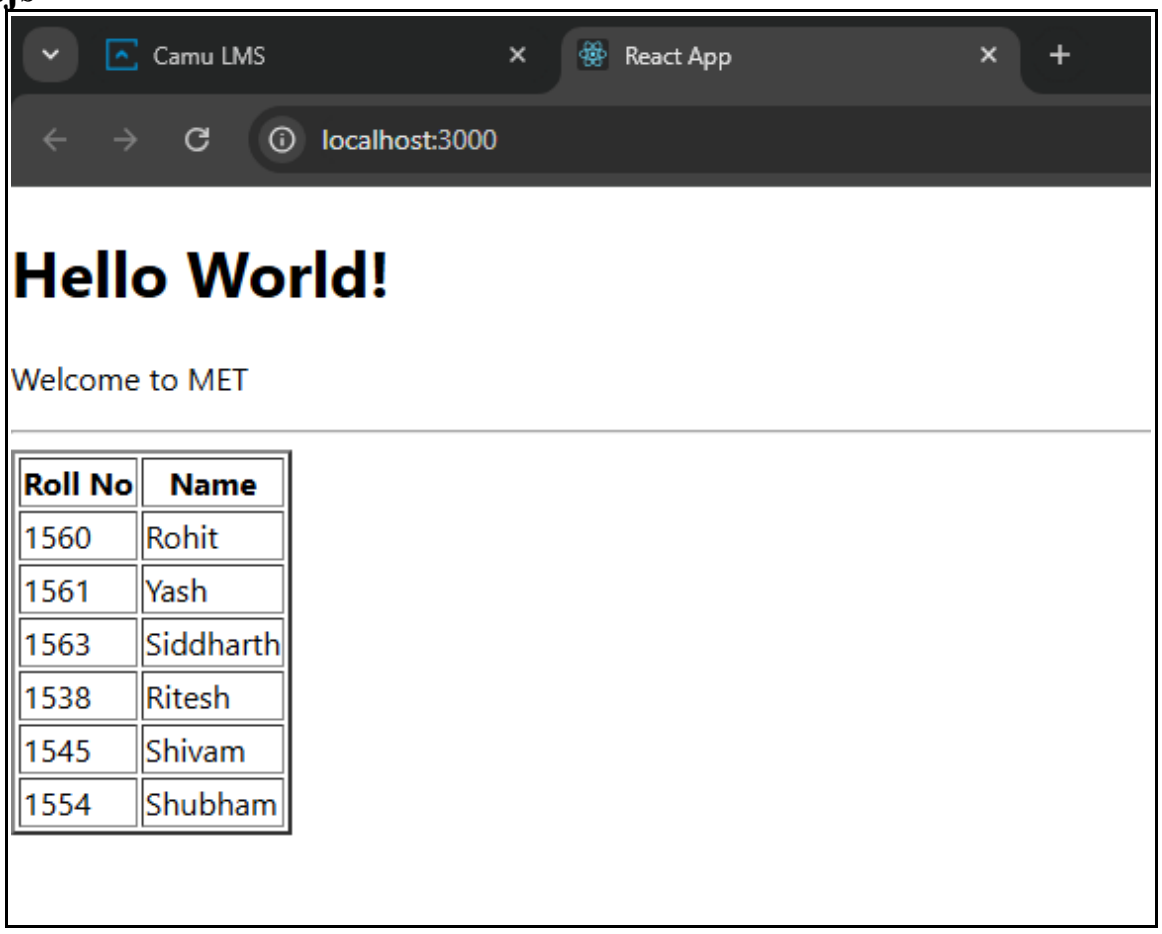
```
          <tr>
            <td>1554</td>
            <td>Shubham</td>
          </tr>
        </tbody>
      </table>
    </div>
  );
}

export default App;
```

App1.js

```
import React from 'react';
import './App.css';

function App1() {
    const color = "red";
  return (
    <div className='container'>
        <h1 style={{'color':color}}>Hello World</h1>
        <p>Rohit Singh (●'◡'●)</p>?□゜⍰⍰
    </div>
  );
}

export default App1;
```

OUTPUT :

App.js



App1.js

Program No	17
Roll No	1554
Title	Build a Calculator
Objective	Design a Calculator using useState

SOURCE CODE -

App.js

```
import React from 'react'
import Form from './components/form'
import Calculator from './components/Calculator'

function App5() {
 return (
  <div>
    <Calculator />
  </div>
 )
}

export default App5
```

Calculator.js

```
import React, {useState} from 'react'

function Calculator() {
  const [val, setVal] = useState("");
  return (
    <div className='container'>
      <div className='row'>
        <div className='col-12'>
          <h1 className='display-5 fw-bolder text-primary'>Calculator</h1>
        </div>
      </div>

      <div>
        <input type='text' value={val}></input>
      </div>

      <div>
        <button className='button' value='1' onClick={(e)=>setVal(val +
e.target.value)}>1</button>
        <button className='button' value='2' onClick={(e)=>setVal(val +
e.target.value)}>2</button>
        <button className='button' value='3' onClick={(e)=>setVal(val +
e.target.value)}>3</button>
```

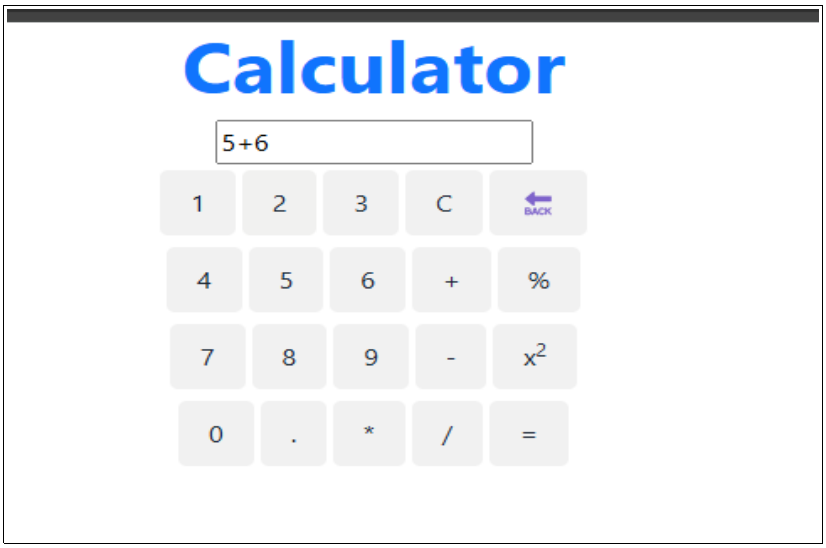
```
        <button className='button' value='C' onClick={(e)=>setVal(val.slice(0,
0))}>C</button>
        <button className='button' value='CE' onClick={(e)=>setVal(val.slice(0,
-1))}>?</button>
      </div>
      <div>
        <button className='button' value='4' onClick={(e)=>setVal(val +
e.target.value)}>4</button>
        <button className='button' value='5' onClick={(e)=>setVal(val +
e.target.value)}>5</button>
        <button className='button' value='6' onClick={(e)=>setVal(val +
e.target.value)}>6</button>
        <button className='button' value='+' onClick={(e)=>setVal(val +
e.target.value)}>+</button>
        <button className='button' value='%' onClick={(e)=>setVal(val +
 e.target.value)}>%</button>
      </div>
      <div>
        <button className='button' value='7' onClick={(e)=>setVal(val +
e.target.value)}>7</button>
        <button className='button' value='8' onClick={(e)=>setVal(val +
e.target.value)}>8</button>
        <button className='button' value='9' onClick={(e)=>setVal(val +
e.target.value)}>9</button>
        <button className='button' value='-' onClick={(e)=>setVal(val +
e.target.value)}>-</button>
        <button className='button' value=''
onClick={(e)=>setVal(val*val)}>x<sup>2</sup></button>
      </div>
      <div>
        <button className='button' value='0' onClick={(e)=>setVal(val +
e.target.value)}>0</button>
        <button className='button' value='.' onClick={(e)=>setVal(val +
e.target.value)}>.</button>
        <button className='button' value='*' onClick={(e)=>setVal(val +
e.target.value)}>*</button>
        <button className='button' value='/' onClick={(e)=>setVal(val +
e.target.value)}>/</button>

        <button className='button' value='='
onClick={(e)=>setVal(eval(val))}>=</button>
      </div>
    </div>
  )
}

export default Calculator
```

index.js

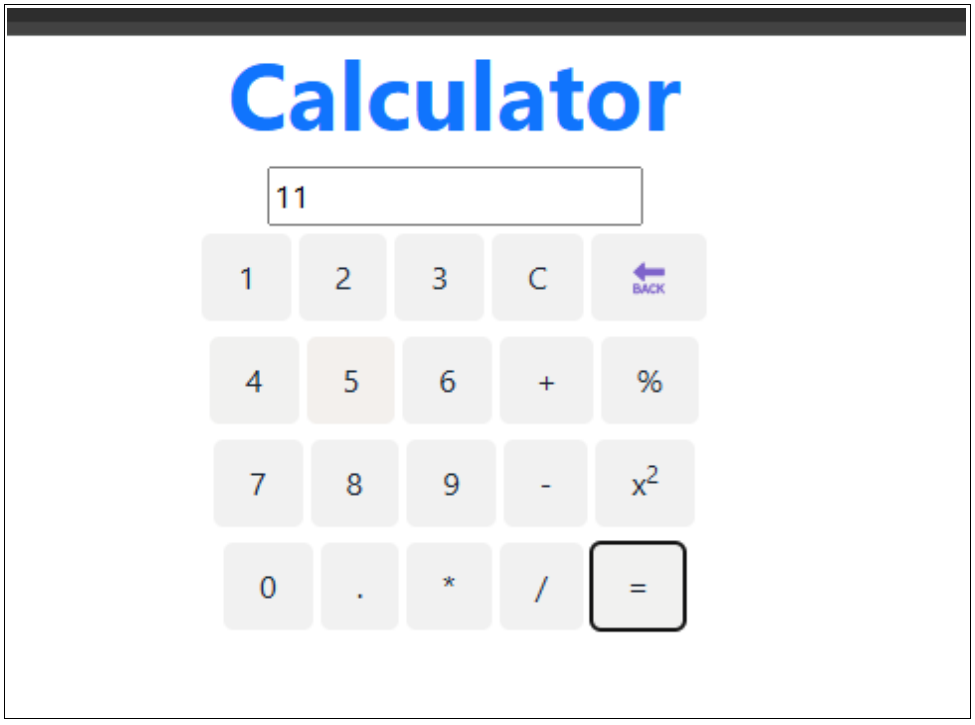
```
import React from 'react';
import ReactDOM from 'react-dom/client';
import '../node_modules/bootstrap/dist/css/bootstrap.min.css'
import './index.css';
import App1 from './App1';
import App2 from './App2';
import App3 from './App3';
import App4 from './App4';
import App5 from './App5';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
 <React.StrictMode>
  <App5 />
 </React.StrictMode>
);

// If you want to start measuring performance in your app, pass a fu-nction
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

OUTPUT -

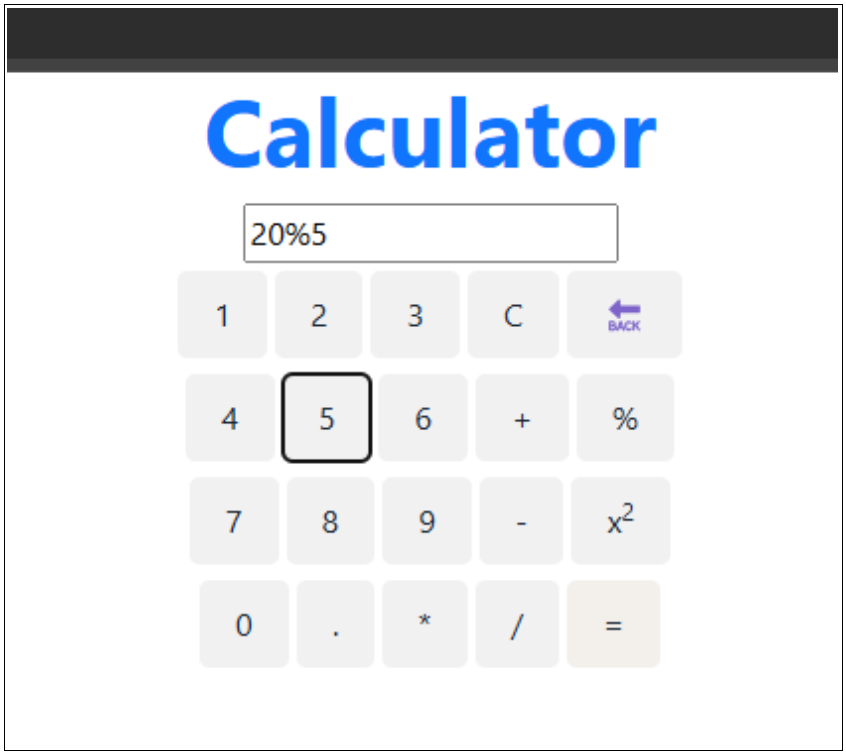
Add


Calculator
5+6
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

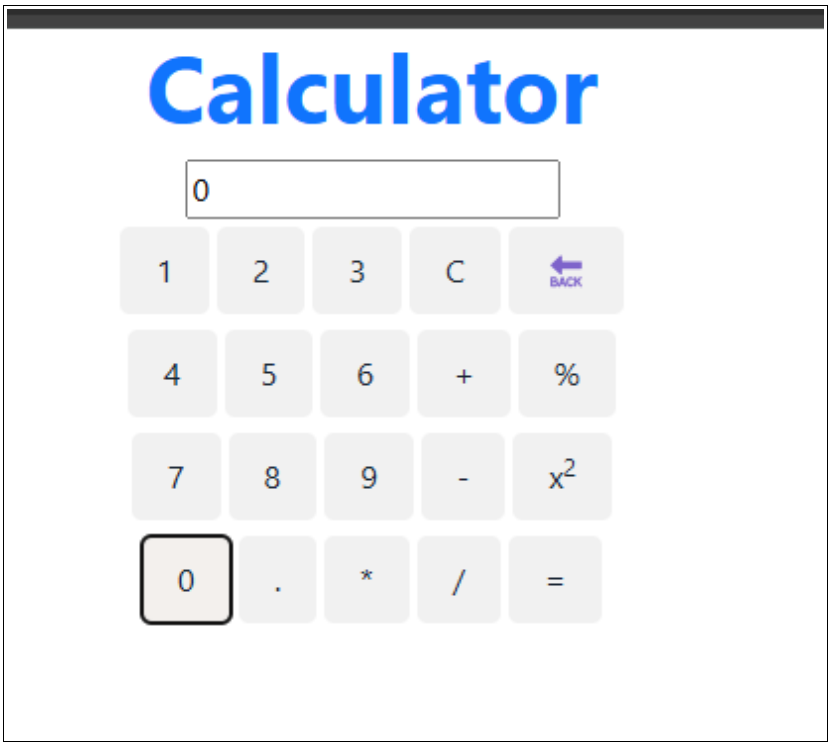
=


Calculator
11
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

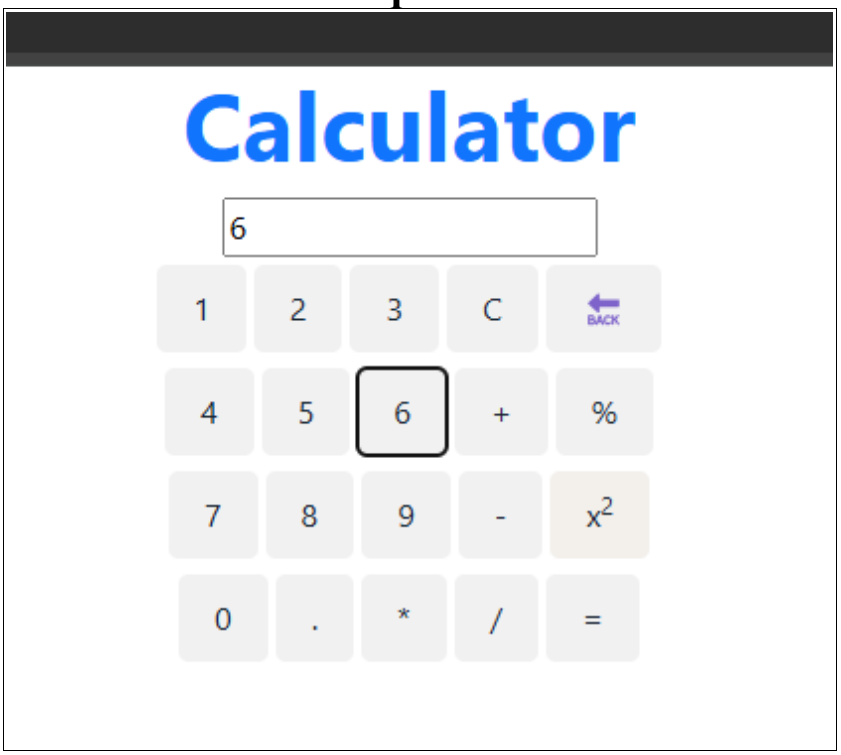
Modulo


Calculator
20%5
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

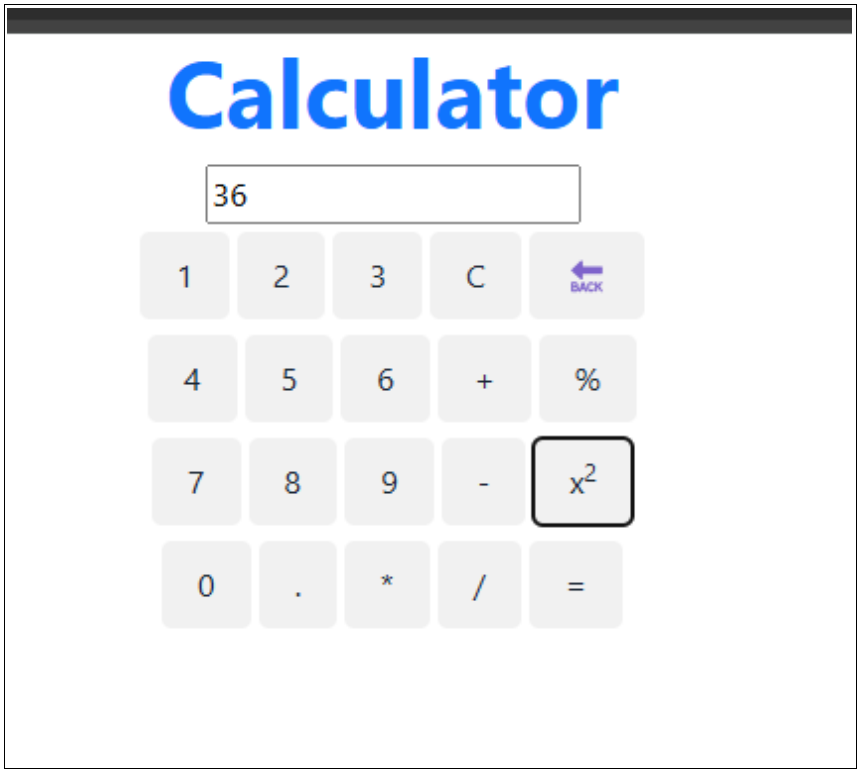
=


Calculator
0
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

Sqaure


Calculator
6
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

=


Calculator
36
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

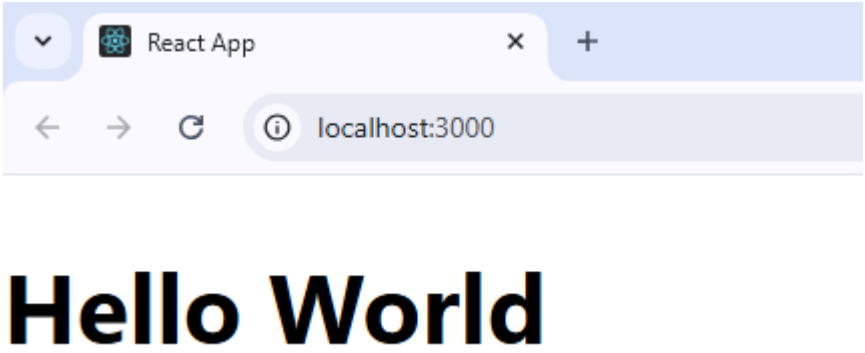
Name:	Siddharth Surve
Roll No:	1563
Program No:	12
Title:	React Js
Objective:	To print Hello World using React JS

Source Code:

Step 1 : Create ReactJs Folder
Step 2 : Create a npm folder in C:\Users\mcamock\AppData\Roaming
Step 3 : Install npm
Step 4 : Create React App by npx create-react-app my-app
Step 5 : Go to cd my-app
Step 6 : npm start in terminal
Step 7 : Type program in App.js

```
import logo from './logo.svg';
import './App.css';

function App() {
 return <h1>Hello World</h1>
 {alert("India")};

}
export default App;
```

Output:



Roll No:	1563
Program No:	13
Title:	Exploring JSX B2
Objective:	Adding multiple lines Creating new App Add new style in App.css Making use of variables

Source Code 1:

```
import './App.css';

function App() {
 return (
  <>
   <h1>Hello world</h1>
   <h2>Welcome to MET</h2>

   <table border='1'>
    <tbody>
     <tr>
      <th>ROLL NO.</th>
      <th>NAME</th>
     </tr>
     <tr>
      <td>101</td>
      <td>Shivam</td>
     </tr>
     <tr>
      <td>102</td>
      <td>Siddharth</td>
     </tr>
     <tr>
      <td>103</td>
      <td>Rohit</td>
     </tr>
     <tr>
```

```
      <td>104</td>
      <td>Ritesh</td>
     </tr>
     <tr>
      <td>105</td>
      <td>Bambam</td>
     </tr>
    </tbody>
   </table>
  </>
 )
}

export default App;
```

Output:

Hello world

Welcome to MET

ROLL NO.	NAME
101	Shivam
102	Siddharth
103	Rohit
104	Ritesh
105	Shubham

Source Code 2:

```
import './App1.css';
import React from 'react';
function App1() {
  const clr="red";
 return (
  <>
   <div class='container' style={{'background-color':clr}}>
    <h1>Hello World</h1>
    <p>Siddharth Surve</p>?¬ﾚ
    </div>
  </>
 )
}

export default App1;
```

Output:

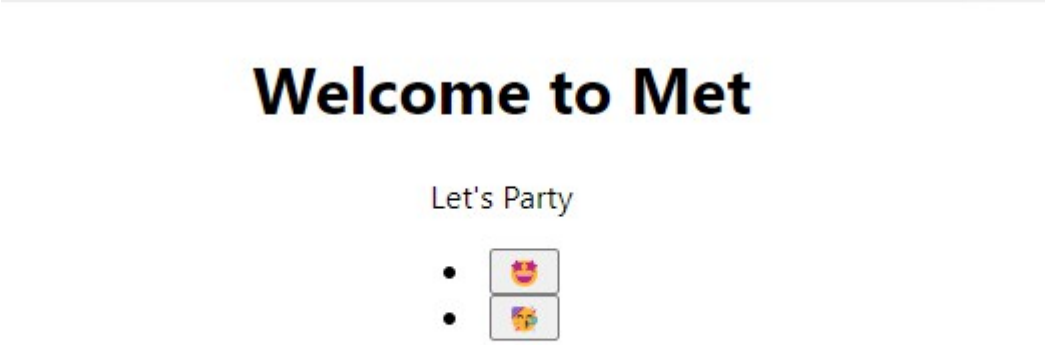
Roll No:	1563
Program No:	14
Title:	JSX to add multiple HTML elements and create Complex Pages
Objective:	A. To create buttons with emoji and handle click event B. To generalise the event funtion create an array for emojis and use map function

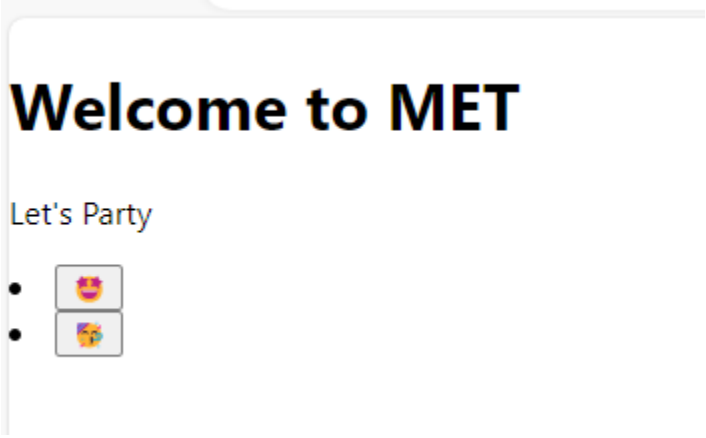
Source Code:

```
//JSX to add many HTML elements and nest them to
import React from 'react';
import './App.css'

function App2()
{
  return (
    <div className="container" style={{textAlign:'center'}}>
      <h1>Welcome to Met</h1>
      <p>Let's Party</p>
      <u>
        <li>
          <button onClick={event =>alert(event.target.id)}>
          <span role='img' aria-label="grinning_face" id='grinning_face'>
            ?
          </span>
          </button>
        </li>
        <li>
        <button onClick={event =>alert(event.target.id)}>
          <span role='img' aria-label="party_popper" id='party_popper'>
            ?
          </span>
        </button>
        </li>
      </u>
    </div>
  )
}
```

```
export default App2;
```

Output:



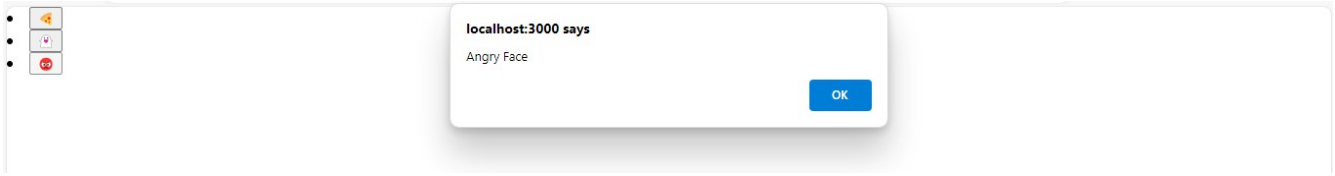


Source Code 2:

```
import React from 'react';
import './App.css'


const displayEmojiName = event => alert(event.target.id);
const emoji = [
  {
    emoji : '?',
    name : "Pizza"
  },
  {
```

```
    emoji : '?',
    name : "Ghost"
  },
  {
    emoji : '?',
    name : "Angry Face"
  }
]

function App4(){
  return(
    <div className="container">
      {
        emoji.map(emoji => (
          <li key={emoji.name}>
            <button onClick={displayEmojiName} >
              <span role="img" src={emoji.emoji} aria-label={emoji.name}
id={emoji.name}>{emoji.emoji}</span>
            </button>
          </li>
        ))
      }
    </div>
  )
}
export default App4
```

Output:


localhost:3000 says
Angry Face
OK


localhost:3000 says
Ghost
OK

Name:	Siddharth Surve
Roll No:	1563
Program No:	15
Title:	Components
Objective	To create Functional and Class components using props and State

Source Code:

App.js

```
import React from 'react';
import './App.css';
import Greet from './components/Greet';
import Welcome from './components/Welcome';
import ProductImg from './components/ProductImg';
import pimg from './components/camera.jpg';
import pimg1 from './components/phone.jfif';
import Count from './components/counter';
function App() {
  return (
    <div className="container">
      <Greet name="Flavia" Surname="Gonsalves" />
      <Greet name="Chetna" Surname="Achar" />
      <Greet name="Neha" Surname="Lodhe" />
      <Welcome />
      <ProductImg im={pimg}  />
      <ProductImg im={pimg1} />
      <Count></Count>
    </div>
  );
}

export default App;
```

Greet.js

```
import React from 'react'

function Greet(props){
  console.log(props);
  return <h1>Hello {props.name} {props.Surname}</h1>;
}
export default Greet;
```

Welcome.js

```
import React,{Component} from 'react';

class Welcome extends Component {
  render() {
    return <div>
      <h1>Class Component</h1>
      <h3>Inside Welcome</h3>
    </div>

  }
}
export default Welcome;
```

ProductImg.js

```
import React,{Component} from 'react';
class ProductImg extends Component
{
  render()
  {
    return(
      <div>
        <img src={this.props.im} alt="Product Image"></img>
        <p>This is Product Component</p>
      </div>
    )
```

```
      }
    }

    export default ProductImg
```

Counter.js

```
    import React, { Component } from "react";

class Counter extends Component{
  constructor(props){
    super(props);
    this.state = {
      count: 0
    }
  }
  updateClick=()=>{
    this.setState({count:this.state.count +1})
  }
  render()
  {
    const {count} = this.state.count;
    return (
      <>
      <button onClick = {this.updateClick}>Visit Click</button>
      <p>Visited {this.state.count} times</p>
      </>
    )
  }
}
export default Counter;
```

Output:


Hello Flavia Gonsalves
Hello Chetna Achar
Hello Neha Lodhe
Class Component
Inside Welcome

Inside Welcome

This is Product Component

This is Product Component

Visit Click

Visited 3 times

Name:	Siddharth Surve
Roll No:	1563
Program No:	17
Title:	React
Objective:	Implementation of use State hook.

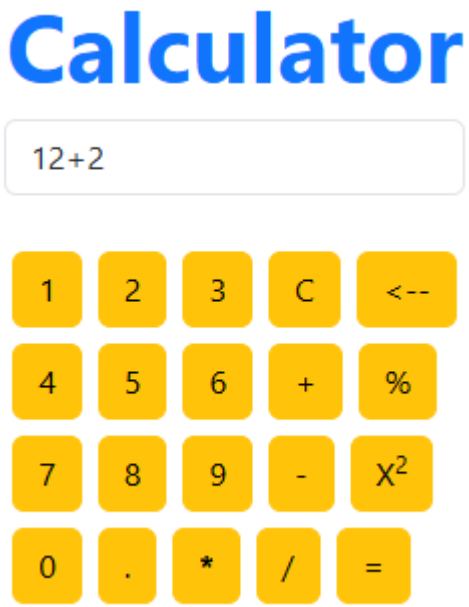
Source Code:

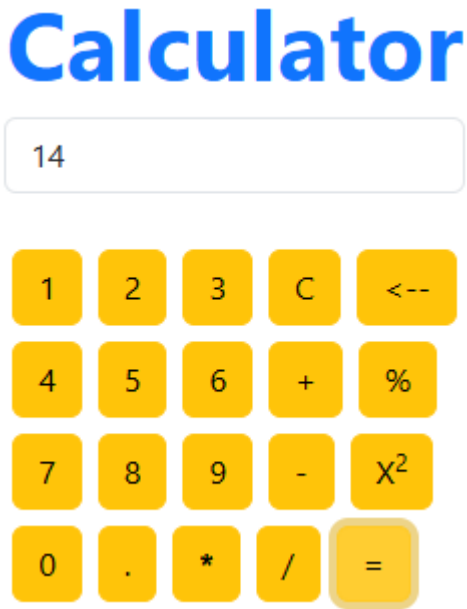
```
import React, {useState} from 'react'

function Calculator() {
 const [val,setVal]=useState("");
 return (
  <div className='container'>
   <div className='row'>
    <div className='col-md-12'>
     <h1 className='display-5 fw-bolder text-primary'
style={{textAlign:'center'}}>
      Calculator
     </h1>
     <form>
     <input type='text' value={val} className='form-control'/>
     </form>
     <br/>
     <div>
     <button className='btn btn-warning m-1' value={1}
onClick={(e)=>setVal(val+e.target.value)}>1</button>
     <button className='btn btn-warning m-1' value={2}
onClick={(e)=>setVal(val+e.target.value)}>2</button>
     <button className='btn btn-warning m-1' value={3}
onClick={(e)=>setVal(val+e.target.value)}>3</button>
     <button className='btn btn-warning m-1' value='c'
onClick={(e)=>setVal(val.slice(0,0))}>C</button>
     <button className='btn btn-warning m-1' value='ce'
onClick={(e)=>setVal(val.slice(0,-1))}>{'<--'}</button>
     <br/>
     <button className='btn btn-warning m-1' value={4}
```

```
onClick={(e)=>setVal(val+e.target.value)}>4</button>
     <button className='btn btn-warning m-1' value={5}
onClick={(e)=>setVal(val+e.target.value)}>5</button>
     <button className='btn btn-warning m-1' value={6}
onClick={(e)=>setVal(val+e.target.value)}>6</button>
     <button className='btn btn-warning m-1' value='+'
onClick={(e)=>setVal(val+e.target.value)}>+</button>
     <button className='btn btn-warning m-1' value='%'
onClick={(e)=>setVal(val+e.target.value)}>%</button>
     <br></br>
     <button className='btn btn-warning m-1' value={7}
onClick={(e)=>setVal(val+e.target.value)}>7</button>
     <button className='btn btn-warning m-1' value={8}
onClick={(e)=>setVal(val+e.target.value)}>8</button>
     <button className='btn btn-warning m-1' value={9}
onClick={(e)=>setVal(val+e.target.value)}>9</button>
     <button className='btn btn-warning m-1' value='-'
onClick={(e)=>setVal(val+e.target.value)}>-</button>
     <button className='btn btn-warning m-1' value='^2'
onClick={(e)=>setVal(val*val)}>X<sup>2</sup></button>
     <br></br>
     <button className='btn btn-warning m-1' value='0'
onClick={(e)=>setVal(val+e.target.value)}>0</button>
     <button className='btn btn-warning m-1' value='.'
onClick={(e)=>setVal(val+e.target.value)}>.</button>
     <button className='btn btn-warning m-1' value='*'
onClick={(e)=>setVal(val+e.target.value)}><b>*</b></button>
     <button className='btn btn-warning m-1' value='/'
onClick={(e)=>setVal(val+e.target.value)}>/</button>
     <button className='btn btn-warning m-1' value='='
onClick={(e)=>setVal(eval(val))}>=</button>
     </div>
    </div>
  </div>
 </div>
 )
}
```

```
export default Calculator;
```

Output:





Calculator

36%2

1	2	3	C	<--
4	5	6	+	%
7	8	9	-	X^2
0	.	*	/	=

Calculator

0

1	2	3	C	<--
4	5	6	+	%
7	8	9	-	X^2
0	.	*	/	=

Calculator

4

1	2	3	C	<--
4	5	6	+	%
7	8	9	-	X^2
0	.	*	/	=

Calculator

16

1	2	3	C	<--
4	5	6	+	%
7	8	9	-	X^2
0	.	*	/	=

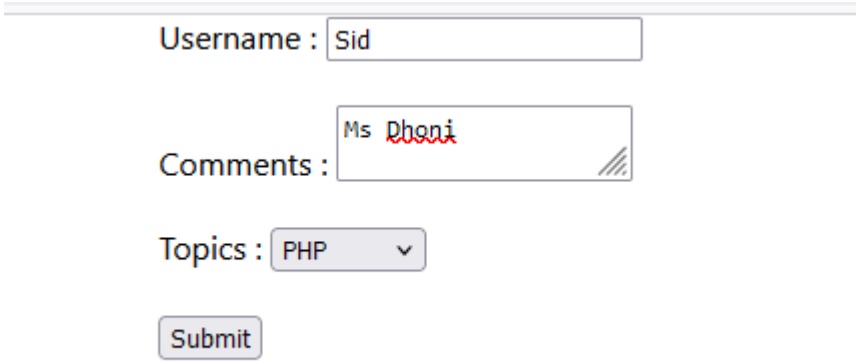
Roll no	1563
program	16
title	Form Handling REACT
objective	Create a form handling to show the working of form using React

Source Code:

Form.js

```
import React, { Component } from 'react'

export class Form extends Component{
  constructor(props){
    super(props)

    this.state = {
      username : '',
      comments : '',
      topic : 'ReactJs'
    }
  }

  handleUsernameChange = (event)=>{
    this.setState({username : event.target.value})
  }
  handleCommentsChange = (event)=>{
    this.setState({comments : event.target.value})
  }
  handleTopicsChange = (event)=>{
    this.setState({topic : event.target.value})
  }
  handleSubmit=(event)=>{
   alert(`${this.state.username}`)
   event.preventDefault()
  }
  render(){
    return (
      <form onSubmit={this.handleSubmit}>
        <div>
          <label>Username : </label>
          <input type='text' value={this.state.username}
onChange={this.handleUsernameChange}/>
```

```
          <br />
          < br/>
          <label>Comments : </label>
          <textarea value={this.state.comments}  onChange={this.handleCommentsChange}
/>
          <br />
          < br/>
          <label>Topics : </label>
          <select value={this.state.topic} onChange={this.handleTopicsChange}>
            <option value='NodeJs'>NodeJs</option>
            <option value='ReactJs'>ReactJs</option>
            <option value='PHP'>PHP</option>
            <option value='ASP.NET'>ASP.NET</option>
          </select><br/><br/>
          <button>Submit</button>
        </div>
      </form>
    )
  }
}
export default Form
```



Output

Program No	17
Roll No	1554
Title	Build a Calculator
Objective	Design a Calculator using useState

SOURCE CODE -

App.js

```
import React from 'react'
import Form from './components/form'
import Calculator from './components/Calculator'

function App5() {
 return (
  <div>
    <Calculator />
  </div>
 )
}

export default App5
```

Calculator.js

```
import React, {useState} from 'react'

function Calculator() {
  const [val, setVal] = useState("");
  return (
    <div className='container'>
      <div className='row'>
        <div className='col-12'>
          <h1 className='display-5 fw-bolder text-primary'>Calculator</h1>
        </div>
      </div>

      <div>
        <input type='text' value={val}></input>
      </div>

      <div>
        <button className='button' value='1' onClick={(e)=>setVal(val +
e.target.value)}>1</button>
        <button className='button' value='2' onClick={(e)=>setVal(val +
e.target.value)}>2</button>
        <button className='button' value='3' onClick={(e)=>setVal(val +
e.target.value)}>3</button>
```

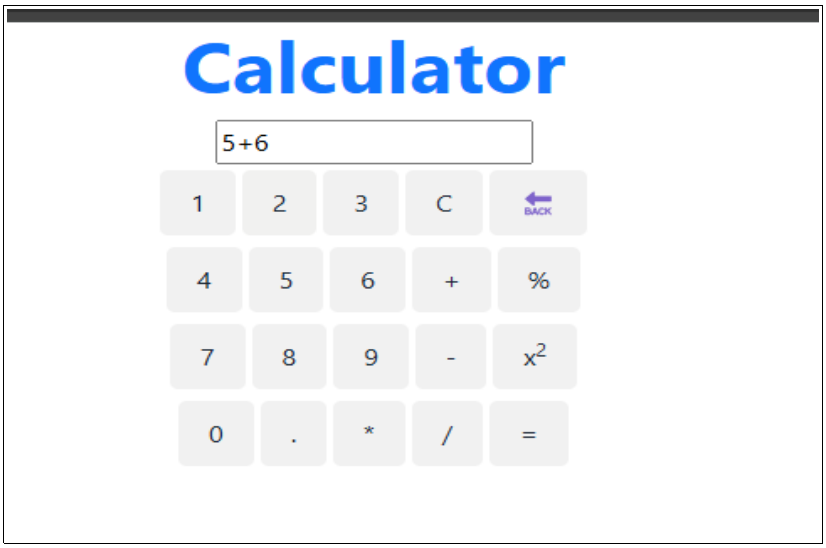
```
        <button className='button' value='C' onClick={(e)=>setVal(val.slice(0,
0))}>C</button>
        <button className='button' value='CE' onClick={(e)=>setVal(val.slice(0,
-1))}>?</button>
      </div>
      <div>
        <button className='button' value='4' onClick={(e)=>setVal(val +
e.target.value)}>4</button>
        <button className='button' value='5' onClick={(e)=>setVal(val +
e.target.value)}>5</button>
        <button className='button' value='6' onClick={(e)=>setVal(val +
e.target.value)}>6</button>
        <button className='button' value='+' onClick={(e)=>setVal(val +
e.target.value)}>+</button>
        <button className='button' value='%' onClick={(e)=>setVal(val +
 e.target.value)}>%</button>
      </div>
      <div>
        <button className='button' value='7' onClick={(e)=>setVal(val +
e.target.value)}>7</button>
        <button className='button' value='8' onClick={(e)=>setVal(val +
e.target.value)}>8</button>
        <button className='button' value='9' onClick={(e)=>setVal(val +
e.target.value)}>9</button>
        <button className='button' value='-' onClick={(e)=>setVal(val +
e.target.value)}>-</button>
        <button className='button' value=''
onClick={(e)=>setVal(val*val)}>x<sup>2</sup></button>
      </div>
      <div>
        <button className='button' value='0' onClick={(e)=>setVal(val +
e.target.value)}>0</button>
        <button className='button' value='.' onClick={(e)=>setVal(val +
e.target.value)}>.</button>
        <button className='button' value='*' onClick={(e)=>setVal(val +
e.target.value)}>*</button>
        <button className='button' value='/' onClick={(e)=>setVal(val +
e.target.value)}>/</button>

        <button className='button' value='='
onClick={(e)=>setVal(eval(val))}>=</button>
      </div>
    </div>
  )
}

export default Calculator
```

index.js

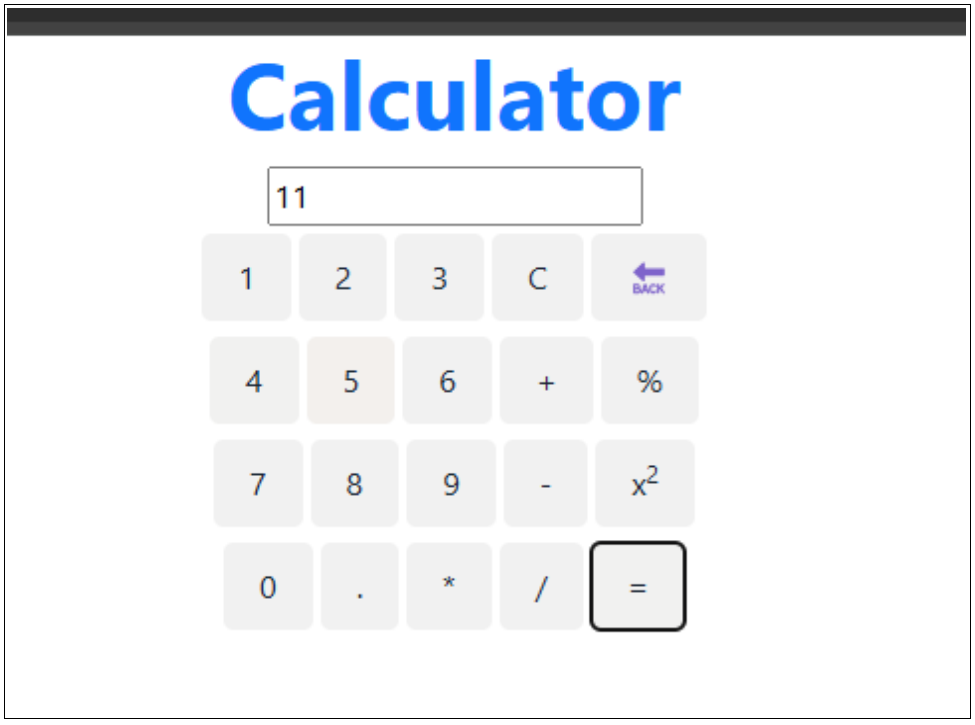
```
import React from 'react';
import ReactDOM from 'react-dom/client';
import '../node_modules/bootstrap/dist/css/bootstrap.min.css'
import './index.css';
import App1 from './App1';
import App2 from './App2';
import App3 from './App3';
import App4 from './App4';
import App5 from './App5';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
 <React.StrictMode>
  <App5 />
 </React.StrictMode>
);

// If you want to start measuring performance in your app, pass a fu-nction
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

OUTPUT -

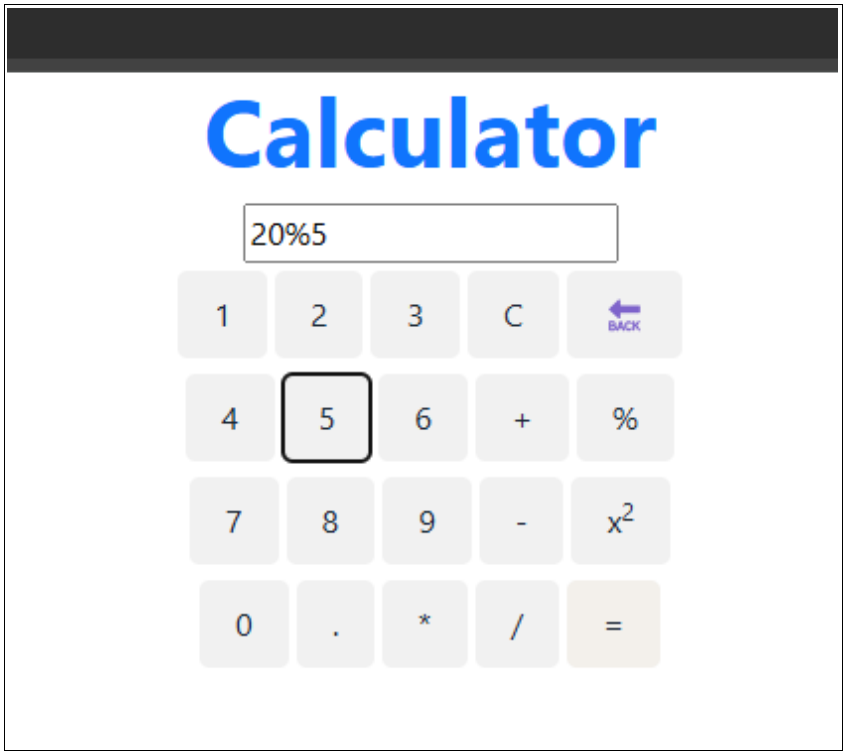
Add


Calculator
5+6
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

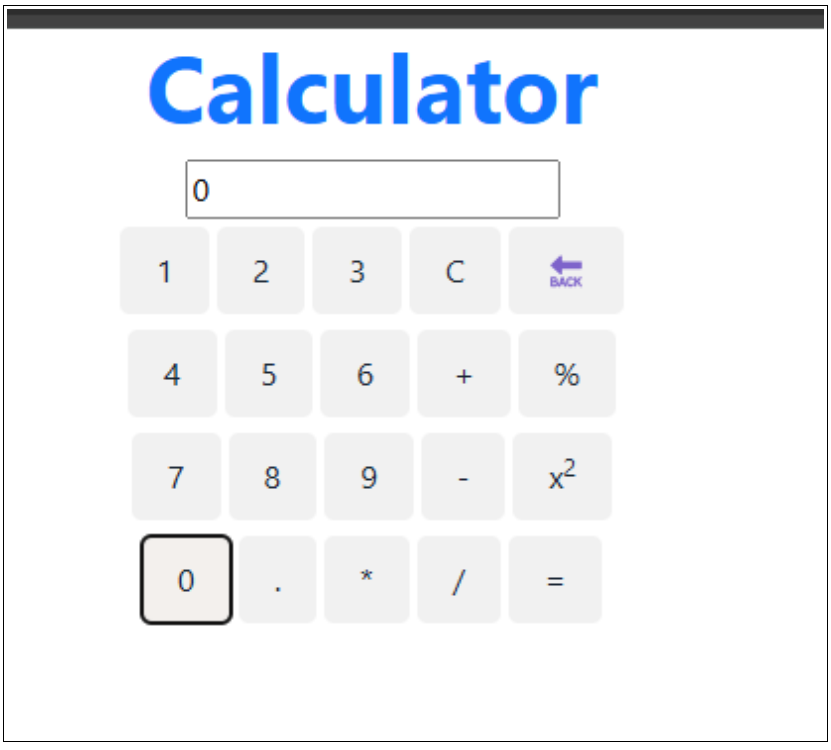
=


Calculator
11
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

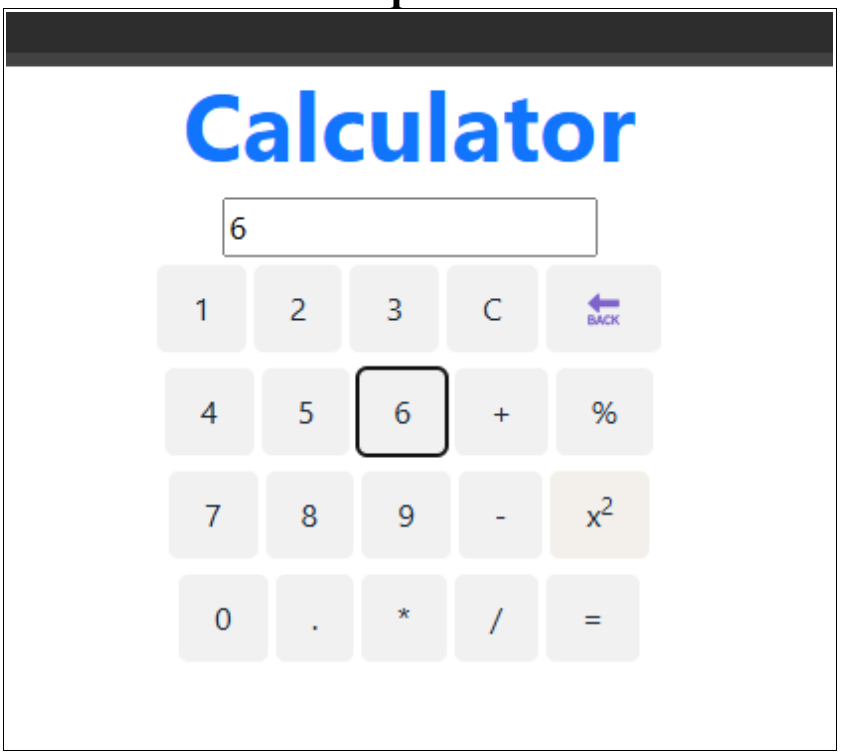
Modulo


Calculator
20%5
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

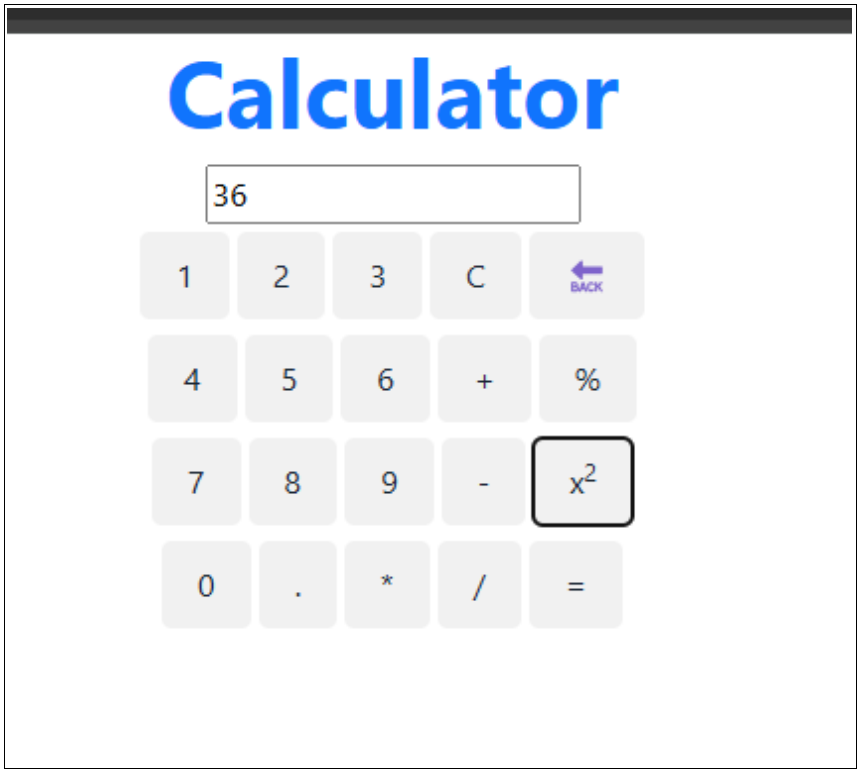
=


Calculator
0
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

Sqaure


Calculator
6
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=

=


Calculator
36
1
2
3
C
BACK
4
5
6
+
%
7
8
9
-
x2
0
.
*
/
=