

Design a web page for adding js for user count each webpage 2

Java script callback function 4

Arrow function 4

Types of style sheet 5

college bill 6

visitor counter 9

module in node js 12

Functional component and class component 12

Internal extranal css 15

Event loop node js 16

Events emitters 16

Form uses functionality of js 18

multi-page website using React.js 19

bio data 24

medical bill 26

display user review using express rest api 45

- Design a web page for adding js for user count each webpage

```
const users = {
```

```
  Alex: {
```

```
    email: 'alex@alex.com',
```

```
    age: 20,
```

```
    isLoggedIn: false,
```

```
    points: 30
```

```
  },
```

```
  Asab: {
```

```
    email: 'asab@asab.com',
```

```
    age: 25,
```

```
    isLoggedIn: false,
```

```
    points: 50
```

```
  },
```

```
  Brook: {
```

```
    email: 'daniel@daniel.com',
```

```
    age: 30,
```

```
    isLoggedIn: true,
```

```
    points: 50
```

```
  },
```

```
  Daniel: {
```

```
    email: 'daniel@alex.com',
```

```
    age: 20,
```

```
    isLoggedIn: false,
    points: 40
  },
  John: {
    email: 'john@john.com',
    age: 20,
    isLoggedIn: true,
    points: 50
  },
  Thomas: {
    email: 'thomas@thomas.com',
    age: 20,
    isLoggedIn: false,
    points: 40
  },
  Paul: {
    email: 'paul@paul.com',
    age: 20,
    isLoggedIn: false,
    points: 40
  }
}

var user = Object.values(users).filter(user => user.isLoggedIn).length
console.log(user)
```

- Java script callback function

```
// function
function greet(name, callback) {
    console.log('Hi' + ' ' + name);
    callback();
}

// callback function
function callMe() {
    console.log('I am callback function');
}

// passing function as an argument
greet('Peter', callMe);
```

- Arrow function

```
<html>
```

```
<body>
```

```
<fieldset style="width:280px">
```

```
<legend>Calculator Demo Using Arrow Function:</legend>
```

```
Number1:<input type="text" id="txtFirstNumber" /><br><br>
```

```
Number2:<input type="text" id="txtSecondNumber" /><br><br>
```

```
<input type="button" id="btnADD" value="Addition"/>
```

```
<input type="button" id="btnSUB" value="Subtraction"/><br><br>
```

```
<input type="button" id="btnMUL" value="Multiplication"/>
```

```
<input type="button" id="btnDIV" value="Division"/>
```

```

    </fieldset>

</body>

</html>

```

```

function Operation(operand)
{
    var firstValue = parseInt(document.getElementById('txtFirstNumber').value);
    var secondValue = parseInt(document.getElementById('txtSecondNumber').value)
;
    var result = 0;
    switch(operand)
    {
case '+' :      result = ( (x, y) => { return x + y; } )( firstValue, secondValue );
break;
case '-' :      result = ( (x, y) => { return x - y; } )( firstValue, secondValue );
break;
case '*' :      result = ( (x, y) => { return x * y; } )( firstValue, secondValue );
break;
case '/' :      result = ( (x, y) => { return x / y; } )( firstValue, secondValue );
break;
    }

    alert(result);
}

```

- Types of style sheet

```

<!DOCTYPE html>
<html>
    <head>
        <title>Internal CSS</title>
        <style>
            .main {
                text-align:center;
            }
            .GFG {
                color:#009900;
                font-size:50px;
                font-weight:bold;
            }
            .geeks {
                font-style:bold;
                font-size:20px;
            }
        </style>
    </head>
    <body>
        <div class="main">
            <div class="GFG">
                <div class="geeks">
                    <h1>GeeksforGeeks</h1>
                </div>
            </div>
        </div>
    </body>
</html>

```

```

        }
    </style>
</head>
<body>
    <div class = "main">
        <div class ="GFG">GeeksForGeeks</div>

        <div class ="geeks">
            A computer science portal for geeks
        </div>
    </div>
</body>
</html>

```

- College bill

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
    <meta charset="UTF-8" />
```

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

```
    <meta
```

```
    name="viewport"
```

```
    content="width=device-width,
```

```
        initial-scale=1.0"/>
```

```
    <link rel="stylesheet" href="style.css"
```

```
        class="css" />
```

```

</head>

<body>

  <div class="container">

    <div class="main-content">

      <p class="text">GeeksforGeeks</p>

    </div>

    <div class="centre-content">

      <h1 class="price">2499<span>/-</span></h1>

      <p class="course">

        Buy DSA Self-Paced Course Now !

      </p>

    </div>

    <div class="last-content">

      <button type="button" class="pay-now-btn">

        Apply Coupons

      </button>

      <button type="button" class="pay-now-btn">

        Pay with Netbanking

      </button>

      <!-- <button type="button" class="pay-now-btn">

        Netbanking options

      </button> -->

```

</div>

<div class="card-details">

<p>Pay using Credit or Debit card</p>

<div class="card-number">

<label> Card Number </label>

<input

type="text"

class="card-number-field"

placeholder="###-###-###"/>

</div>

<div class="date-number">

<label> Expiry Date </label>

<input type="text" class="date-number-field"

placeholder="DD-MM-YY" />

</div>

<div class="cvv-number">

<label> CVV number </label>

<input type="text" class="cvv-number-field"

placeholder="xxx" />

</div>

<div class="nameholder-number">


```

        <label> Card Holder name </label>

        <input
            type="text"
            class="card-name-field"
            placeholder="Enter your Name"/>

    </div>

    <button type="submit"
        class="submit-now-btn">

        submit
    </button>

</div>

</div>

</body>

</html>

```

- Visitor counter

index.html

```

<html>

<head>

    <title>Website Counter</title>

    <script defer src="index.js"></script>

    <link rel="stylesheet" href="styles.css" />

</head>

<body>

```

```
<div>Website visit count:</div>

<div class="website-counter"></div>

<button id="reset">Reset</button>

</body>

</html>
```

index.js

```
var counterContainer = document.querySelector(".website-counter");

var resetButton = document.querySelector("#reset");

var visitCount = localStorage.getItem("page_view");

// Check if page_view entry is present
if (visitCount) {

    visitCount = Number(visitCount) + 1;

    localStorage.setItem("page_view", visitCount);

} else {

    visitCount = 1;

    localStorage.setItem("page_view", 1);

}

counterContainer.innerHTML = visitCount;

// Adding onClick event listener
resetButton.addEventListener("click", () => {

    visitCount = 1;

    localStorage.setItem("page_view", 1);

})
```

```
counterContainer.innerHTML = visitCount;

});

style.css
```

```
/* Center child elements */
```

```
body,
```

```
.website-counter {
```

```
display: flex;
```

```
justify-content: center;
```

```
align-items: center;
```

```
flex-direction: column;
```

```
}
```

```
body {
```

```
height: 100vh;
```

```
}
```

```
/* Styles for website counter container */
```

```
.website-counter {
```

```
background-color: #ff4957;
```

```
height: 50px;
```

```
width: 80px;
```

```
color: white;
```

```
border-radius: 30px;
```

```
font-weight: 700;
```

```

font-size: 25px;

margin-top: 10px;

}

/* Styles for reset button */

#reset {

margin-top: 20px;

background-color: #008cba;

cursor: pointer;

font-size: 18px;

padding: 8px 20px;

color: white;

border: 0;

}

```

- Module in node.js

```

var http = require('http');

var dt = require('./myfirstmodule');

http.createServer(function (req, res) {

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

res.write("The date and time are currently: " + dt.myDateTime());

res.end();

}).listen(8080);

```

- Functional component and class component

```

import React, { useState } from "react";

```

```

const FunctionalComponent=()=>{
  const [count, setCount] = useState(0);

  const increase = () => {
    setCount(count+1);
  }

  return (
    <div style={{margin:'50px'}}>
      <h1>Welcome to Geeks for Geeks </h1>
      <h3>Counter App using Functional Component : </h3>
      <h2>{count}</h2>
      <button onClick={increase}>Add</button>
    </div>
  )
}

export default FunctionalComponent;

```

class-

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

```

class ClassComponent extends React.Component{
  constructor(){
    super();
  }

```

```

    this.state={
        count :0
    };
    this.increase=this.increase.bind(this);
}

increase(){
    this.setState({count : this.state.count +1});
}

render(){
    return (
        <div style={{margin:'50px'}}>
            <h1>Welcome to Geeks for Geeks </h1>
            <h3>Counter App using Class Component : </h3>
            <h2> {this.state.count}</h2>
            <button onClick={this.increase}> Add</button>

        </div>
    )
}
}

export default ClassComponent;

```

- Internal extranal css

```
<!DOCTYPE html>

<html>

<head>

<style>

body {

    background-color: blue;

}

h1 {

    color: red;

    padding: 60px;

}

</style>

</head>

<body>

<h1>Hostinger Tutorials</h1>

<p>This is our paragraph.</p>

</body>

</html>
```

Internal-

```
.class {
  property1 : value1;
  property2 : value2;
  property3 : value3;
}
```

```
#id {
  property1 : value1;
  property2 : value2;
  property3 : value3;
}
```

External-

```
.xleftcol {
  float: left;
  width: 33%;
  background:#809900;
}
.xmiddlecol {
  float: left;
  width: 34%;
  background:#eff2df;
}
```

- Event loop node js

```
console.log("This is the first statement");
```

```
setTimeout(function(){
  console.log("This is the second statement");
}, 1000);
```

```
console.log("This is the third statement");
```


- Events emitters

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

```
var eventEmitter = new events.EventEmitter();
```

```
// listener #1
```

```
var listner1 = function listner1() {  
    console.log('listner1 executed.');
```

```
}
```

```
// listener #2
```

```
var listner2 = function listner2() {  
    console.log('listner2 executed.');
```

```
}
```

```
// Bind the connection event with the listner1 function
```

```
eventEmitter.addListener('connection', listner1);
```

```
// Bind the connection event with the listner2 function
```

```
eventEmitter.on('connection', listner2);
```

```
var eventListeners = require('events').EventEmitter.listenerCount
```

```
(eventEmitter, 'connection');
```

```
console.log(eventListeners + " Listner(s) listening to connection event");
```

```
// Fire the connection event
```

```
eventEmitter.emit('connection');
```

```

// Remove the binding of listner1 function

eventEmitter.removeListener('connection', listner1);

console.log("Listner1 will not listen now.");


// Fire the connection event

eventEmitter.emit('connection');


eventListeners = require('events').EventEmitter.listenerCount(eventEmitter,'connection');

console.log(eventListeners + " Listner(s) listening to connection event");


console.log("Program Ended.");

```

- Form uses functionality of js

```

<html>

<head>

<title> Login Form</title>

</head>

<body>

<h3> LOGIN </h3>

<form form = "Login_form" onsubmit="submit_form()">

<h4> USERNAME</h4>

<input type="text" placeholder="Enter your email id"/>

<h4> PASSWORD</h4>

```

```

<input type="password" placeholder="Enter your password"/></br></br>

<input type="submit" value="Login"/>

<input type="button" value="SignUp" onClick="create()"/>

</form>

<script type="text/javascript">

function submit_form(){

alert("Login successfully");

}

function create(){

window.location="signup.html";

}

</script>

</body>

</html>

```

- Multi pages using react

Index.js

```

import React from "react";
import { Nav, NavLink, NavMenu }
  from "./NavbarElements";

```

```

const Navbar = () => {
  return (
    <>
      <Nav>
        <NavMenu>
          <NavLink to="/about" activeStyle>
            About
          </NavLink>
          <NavLink to="/contact" activeStyle>
            Contact Us
          </NavLink>

```

```

        <NavLink to="/blogs" activeStyle>
            Blogs
        </NavLink>
        <NavLink to="/sign-up" activeStyle>
            Sign Up
        </NavLink>
    </NavMenu>
</Nav>
</>
);
};

```

export default Navbar;

NavbarElements.js

```

import { FaBars } from "react-icons/fa";
import { NavLink as Link } from "react-router-dom";
import styled from "styled-components";

export const Nav = styled.nav`
    background: #ffb3ff;
    height: 85px;
    display: flex;
    justify-content: space-between;
    padding: 0.2rem calc((100vw - 1000px) / 2);
    z-index: 12;
`;

export const NavLink = styled(Link)`
    color: #808080;
    display: flex;
    align-items: center;
    text-decoration: none;
    padding: 0 1rem;
    height: 100%;
    cursor: pointer;
    &.active {
        color: #4d4dff;
    }
`;

export const Bars = styled(FaBars)`
    display: none;
    color: #808080;
    @media screen and (max-width: 768px) {
        display: block;
        position: absolute;
        top: 0;
        right: 0;
        transform: translate(-100%, 75%);
    }

```

```

        font-size: 1.8rem;
        cursor: pointer;
    }
`;

export const NavMenu = styled.div`
    display: flex;
    align-items: center;
    margin-right: -24px;
    /* Second Nav */
    /* margin-right: 24px; */
    /* Third Nav */
    /* width: 100vw;
white-space: nowrap; */
    @media screen and (max-width: 768px) {
        display: none;
    }
`;

```

About.js

```

import { FaBars } from "react-icons/fa";
import { NavLink as Link } from "react-router-dom";
import styled from "styled-components";

export const Nav = styled.nav`
    background: #ffb3ff;
    height: 85px;
    display: flex;
    justify-content: space-between;
    padding: 0.2rem calc((100vw - 1000px) / 2);
    z-index: 12;
`;

export const NavLink = styled(Link)`
    color: #808080;
    display: flex;
    align-items: center;
    text-decoration: none;
    padding: 0 1rem;
    height: 100%;
    cursor: pointer;
    &.active {
        color: #4d4dff;
    }
`;

export const Bars = styled(FaBars)`
    display: none;
    color: #808080;
    @media screen and (max-width: 768px) {
        display: block;
        position: absolute;
        top: 0;
        right: 0;
    }
`;

```

```

        transform: translate(-100%, 75%);
        font-size: 1.8rem;
        cursor: pointer;
    }
`;

export const NavMenu = styled.div`
    display: flex;
    align-items: center;
    margin-right: -24px;
    /* Second Nav */
    /* margin-right: 24px; */
    /* Third Nav */
    /* width: 100vw;
white-space: nowrap; */
    @media screen and (max-width: 768px) {
        display: none;
    }
`;

```

```

Blogs.js
import React from 'react';

const Blogs = () => {
    return (
        <h1>You can write your blogs!</h1>
    );
};

export default Blogs;

```

```

index.js
import React from 'react';

const Home = () => {
    return (
        <div>
            <h1>Welcome to GeeksforGeeks</h1>
        </div>
    );
};

export default Home;

```

```

signup.js
import React from 'react';

const SignUp = () => {
    return (
        <div>
            <h1>Sign Up Successful</h1>
        </div>
    );
};

```

```
export default SignUp;
```

index.js

```
import React from 'react';

import ReactDOM from 'react-dom';

import App from './App';

ReactDOM.render(

  <React.StrictMode>

    <App />

  </React.StrictMode>,

  document.getElementById('root')

);
```

App.js

```
import React from 'react';
import './App.css';
import Navbar from './components/Navbar';
import { BrowserRouter as Router, Routes, Route }
  from 'react-router-dom';
import Home from './pages';
import About from './pages/about';
import Blogs from './pages/blogs';
import SignUp from './pages/signup';
import Contact from './pages/contact';

function App() {
  return (
    <Router>
```

```

    <Navbar />
    <Routes>
      <Route exact path="/" exact element={<Home />} />
      <Route path="/about" element={<About/>} />
      <Route path="/contact" element={<Contact/>} />
      <Route path="/blogs" element={<Blogs/>} />
      <Route path="/sign-up" element={<SignUp/>} />
    </Routes>
  </Router>
);
}

export default App;

```

- Bio data in react (<https://www.freecodecamp.org/news/portfolio-app-using-react-618814e35843/>)

```

<section id="colorlib-hero" class="js-fullheight" data-section="home">

  <div class="flexslider js-fullheight">

    <ul class="slides">

      <li style="background-image: url(images/img_bg_1.jpg);">

        <div class="overlay"></div>

        <div class="container-fluid">

          <div class="row">

            <div class="col-md-6 col-md-offset-3 col-md-pull-3 col-sm-12 col-xs-12 js-fullheight
slider-text">

              <div class="slider-text-inner js-fullheight">

                <div class="desc">

                  <h1>Hi! <br>I'm Jackson</h1>

                  <h2>100% html5 bootstrap templates Made by <a href="https://colorlib.com/"
target="_blank">colorlib.com</a></h2>

                  <p><a class="btn btn-primary btn-learn">Download CV <em class="icon-
download4"></em></a></p>

                </div>

```



```

        </div>

    </div>

</div>

</div>

</li>

<li style="background-image: url(images/img_bg_2.jpg);">

    <div class="overlay"></div>

    <div class="container-fluid">

        <div class="row">

            <div class="col-md-6 col-md-offset-3 col-md-pull-3 col-sm-12 col-xs-12 js-fullheight
slider-text">

                <div class="slider-text-inner">

                    <div class="desc">

                        <h1>I am <br>a Designer</h1>

                        <h2>100% html5 bootstrap templates Made by <a href="https://colorlib.com/"
target="_blank">colorlib.com</a></h2>

                        <p><a class="btn btn-primary btn-learn">View Portfolio <em class="icon-
briefcase3"></em></a></p>

                    </div>

                </div>

            </div>

        </div>

    </div>

</li>

</ul>

</div>

</section>

```

- Medical bill in html css & javascript :

Html code :

```
<html>

<head>

<meta charset="utf-8">

<title>Medicclaim medical</title>

<link rel="stylesheet" href="style.css">

<link rel="license" href="https://www.opensource.org/licenses/mit-license/">

<script src="script.js"></script>

</head>

<body>

<header>

<h1>Medicclaim Medical</h1>

<address contenteditable>

<p>Jonathan Neal</p>

<p>101 E. Chapman Ave<br>Orange, CA 92866</p>

<p>(800) 555-1234</p>

</address>

<span><input type="file"
accept="image/*"></span>

</header>

<article>

<h1>Recipient</h1>

<address contenteditable>

<p>Payment Info :<br>Total Medicines</p>
```

```

</address>

<table class="meta">

<tr>

<th><span contenteditable>Patient Name</span></th>

<td><span contenteditable>Raju</span></td>

</tr>

<tr>

<th><span contenteditable>Date</span></th>

<td><span contenteditable>January 1, 2012</span></td>

</tr>

<tr>

<th><span contenteditable>Amount Due</span></th>

<td><span id="prefix" contenteditable>Rs.</span><span>600.00</span></td>

</tr>

</table>

<table class="inventory">

<thead>

<tr>

<th><span contenteditable>Item</span></th>

<th><span contenteditable>Description</span></th>

<th><span contenteditable>Rate</span></th>

<th><span contenteditable>Quantity</span></th>

<th><span contenteditable>Price</span></th>

</tr>

</thead>

```

```

<tbody>
<tr>
<td><a class="cut">-</a><span contenteditable>Dolo 600</span></td>
<td><span contenteditable>Fever</span></td>
<td><span data-prefix>$</span><span contenteditable>150.00</span></td>
<td><span contenteditable>4</span></td>
<td><span data-prefix>Rs:</span><span>600.00</span></td>
</tr>
</tbody>
</table>
<a class="add">+</a>
<table class="balance">
<tr>
<th><span contenteditable>Total</span></th>
<td><span data-prefix>$</span><span>600.00</span></td>
</tr>
<tr>
<th><span contenteditable>Amount Paid</span></th>
<td><span data-prefix>$</span><span contenteditable>0.00</span></td>
</tr>
<tr>
<th><span contenteditable>Balance Due</span></th>
<td><span data-prefix>$</span><span>600.00</span></td>
</tr>
</table>

```

```
</article>

<aside>

<h1><span contenteditable>Additional Notes</span></h1>

<div contenteditable>

<p>A finance charge of 1.5% will be made on unpaid balances after 30 days.</p>

</div>

</aside>

</body>

</html>
```

CSS CODE :

```
/* reset */

*

{

border: 0;

box-sizing: content-box;

color: inherit;

font-family: inherit;

font-size: inherit;

font-style: inherit;

font-weight: inherit;

line-height: inherit;

list-style: none;

margin: 0;
```

```
padding: 0;

text-decoration: none;

vertical-align: top;

}
```

```
/* content editable */
```

```
*[contenteditable] { border-radius: 0.25em; min-width: 1em; outline: 0; }
```

```
*[contenteditable] { cursor: pointer; }
```

```
*[contenteditable]:hover, *[contenteditable]:focus, td:hover *[contenteditable], td:focus
*[contenteditable], img.hover { background: #DEF; box-shadow: 0 0 1em 0.5em #DEF; }
```

```
span[contenteditable] { display: inline-block; }
```

```
/* heading */
```

```
h1 { font: bold 100% sans-serif; letter-spacing: 0.5em; text-align: center; text-transform: uppercase; }
```

```
/* table */
```

```
table { font-size: 75%; table-layout: fixed; width: 100%; }
```

```
table { border-collapse: separate; border-spacing: 2px; }
```

```
th, td { border-width: 1px; padding: 0.5em; position: relative; text-align: left; }
```

```
th, td { border-radius: 0.25em; border-style: solid; }
```

```
th { background: #EEE; border-color: #BBB; }
```

```
td { border-color: #DDD; }
```

```
/* page */
```

```
html { font: 16px/1 'Open Sans', sans-serif; overflow: auto; padding: 0.5in; }
```

```
html { background: #999; cursor: default; }
```

```
body { box-sizing: border-box; height: 11in; margin: 0 auto; overflow: hidden; padding: 0.5in; width: 8.5in; }
```

```
body { background: #FFF; border-radius: 1px; box-shadow: 0 0 1in -0.25in rgba(0, 0, 0, 0.5); }
```

```
/* header */
```

```
header { margin: 0 0 3em; }
```

```
header:after { clear: both; content: ""; display: table; }
```

```
header h1 { background: #000; border-radius: 0.25em; color: #FFF; margin: 0 0 1em; padding: 0.5em 0; }
```

```
header address { float: left; font-size: 75%; font-style: normal; line-height: 1.25; margin: 0 1em 1em 0; }
```

```
header address p { margin: 0 0 0.25em; }
```

```
header span, header img { display: block; float: right; }
```

```
header span { margin: 0 0 1em 1em; max-height: 25%; max-width: 60%; position: relative; }
```

```
header img { max-height: 100%; max-width: 100%; }
```

```
header input { cursor: pointer; -ms-filter:"progid:DXImageTransform.Microsoft.Alpha(Opacity=0)";  
height: 100%; left: 0; opacity: 0; position: absolute; top: 0; width: 100%; }
```

```
/* article */
```

```
article, article address, table.meta, table.inventory { margin: 0 0 3em; }
```

```
article:after { clear: both; content: ""; display: table; }
```

```
article h1 { clip: rect(0 0 0 0); position: absolute; }
```

```
article address { float: left; font-size: 125%; font-weight: bold; }
```

```
/* table meta & balance */
```

```
table.meta, table.balance { float: right; width: 36%; }
```

```
table.meta:after, table.balance:after { clear: both; content: ""; display: table; }
```

```
/* table meta */
```

```
table.meta th { width: 40%; }
```

```
table.meta td { width: 60%; }
```

```
/* table items */
```

```
table.inventory { clear: both; width: 100%; }
```

```
table.inventory th { font-weight: bold; text-align: center; }
```

```
table.inventory td:nth-child(1) { width: 26%; }
```

```
table.inventory td:nth-child(2) { width: 38%; }
```



```
table.inventory td:nth-child(3) { text-align: right; width: 12%; }
```

```
table.inventory td:nth-child(4) { text-align: right; width: 12%; }
```

```
table.inventory td:nth-child(5) { text-align: right; width: 12%; }
```

```
/* table balance */
```

```
table.balance th, table.balance td { width: 50%; }
```

```
table.balance td { text-align: right; }
```

```
/* aside */
```

```
aside h1 { border: none; border-width: 0 0 1px; margin: 0 0 1em; }
```

```
aside h1 { border-color: #999; border-bottom-style: solid; }
```

```
/* javascript */
```

```
.add, .cut
```

```
{
```

```
border-width: 1px;
```

```
display: block;
```

```
font-size: .8rem;
```

```
padding: 0.25em 0.5em;
```

```
float: left;
```

```
text-align: center;
```

```
width: 0.6em;
```

```
}
```

```
.add, .cut
```

```
{
```

```
background: #9AF;
```

```
box-shadow: 0 1px 2px rgba(0,0,0,0.2);
```

```
background-image: -moz-linear-gradient(#00ADEE 5%, #0078A5 100%);
```

```
background-image: -webkit-linear-gradient(#00ADEE 5%, #0078A5 100%);
```

```
border-radius: 0.5em;
```

```
border-color: #0076A3;
```

```
color: #FFF;
```

```
cursor: pointer;
```

```
font-weight: bold;
```

```
text-shadow: 0 -1px 2px rgba(0,0,0,0.333);
```

```
}
```

```
.add { margin: -2.5em 0 0; }
```

```
.add:hover { background: #00ADEE; }
```

```
.cut { opacity: 0; position: absolute; top: 0; left: -1.5em; }
```

```
.cut { -webkit-transition: opacity 100ms ease-in; }
```

```
tr:hover .cut { opacity: 1; }
```

```

@media print {
* { -webkit-print-color-adjust: exact; }

html { background: none; padding: 0; }

body { box-shadow: none; margin: 0; }

span:empty { display: none; }

.add, .cut { display: none; }

}

```

```

@page { margin: 0; }

```

Javascript code :

```

/* Shivving (IE8 is not supported, but at least it won't look as awful)

/* ===== */

(function (document) {

var

head = document.head = document.getElementsByTagName('head')[0] || document.documentElement,

elements = 'article aside audio bdi canvas data datalist details figcaption figure footer header hgroup
mark meter nav output picture progress section summary time video x'.split(' '),

elementsLength = elements.length,

elementsIndex = 0,

element;

```

```

while (elementsIndex < elementsLength) {

element = document.createElement(elements[++elementsIndex]);

}


element.innerHTML = 'x<style>' +

'article,aside,details,figcaption,figure,footer,header,hgroup,nav,section{display:block}' +

'audio[controls],canvas,video{display:inline-block}' +

'[/hidden],audio{display:none}' +

'mark{background:#FF0;color:#000}' +

'</style>';


return head.insertBefore(element.lastChild, head.firstChild);

})(document);


/* Prototyping

/* ===== */


(function (window, ElementPrototype, ArrayPrototype, polyfill) {

function NodeList() { [polyfill] }

NodeList.prototype.length = ArrayPrototype.length;


ElementPrototype.matchesSelector = ElementPrototype.matchesSelector ||

ElementPrototype.mozMatchesSelector ||

ElementPrototype.msMatchesSelector ||

ElementPrototype.oMatchesSelector ||

```

```
ElementPrototype.webkitMatchesSelector ||
```

```
function matchesSelector(selector) {
```

```
return ArrayPrototype.indexOf.call(this.parentNode.querySelectorAll(selector), this) > -1;
```

```
};
```

```
ElementPrototype.ancestorQuerySelectorAll = ElementPrototype.ancestorQuerySelectorAll ||
```

```
ElementPrototype.mozAncestorQuerySelectorAll ||
```

```
ElementPrototype.msAncestorQuerySelectorAll ||
```

```
ElementPrototype.oAncestorQuerySelectorAll ||
```

```
ElementPrototype.webkitAncestorQuerySelectorAll ||
```

```
function ancestorQuerySelectorAll(selector) {
```

```
for (var cite = this, newNodeList = new NodeList; cite = cite.parentElement;) {
```

```
if (cite.matchesSelector(selector)) ArrayPrototype.push.call(newNodeList, cite);
```

```
}
```

```
return newNodeList;
```

```
};
```

```
ElementPrototype.ancestorQuerySelector = ElementPrototype.ancestorQuerySelector ||
```

```
ElementPrototype.mozAncestorQuerySelector ||
```

```
ElementPrototype.msAncestorQuerySelector ||
```

```
ElementPrototype.oAncestorQuerySelector ||
```

```
ElementPrototype.webkitAncestorQuerySelector ||
```

```
function ancestorQuerySelector(selector) {
```

```
return this.ancestorQuerySelectorAll(selector)[0] || null;
```

```

};

))(this, Element.prototype, Array.prototype);

/* Helper Functions

/* ===== */

function generateTableRow() {
var emptyColumn = document.createElement('tr');

emptyColumn.innerHTML = '<td><a class="cut">-</a><span contenteditable></span></td>' +
'<td><span contenteditable></span></td>' +
'<td><span data-prefix>$</span><span contenteditable>0.00</span></td>' +
'<td><span contenteditable>0</span></td>' +
'<td><span data-prefix>$</span><span>0.00</span></td>';

return emptyColumn;
}

function parseFloatHTML(element) {
return parseFloat(element.innerHTML.replace(/[^d\.\-]+/g, '')) || 0;
}

function parsePrice(number) {
return number.toFixed(2).replace(/(\d)(?=(\d\d\d)+([^\d]|$))/g, '$1,');
}

```

```

/* Update Number

/* ===== */

function updateNumber(e) {

var

activeElement = document.activeElement,

value = parseFloat(activeElement.innerHTML),

wasPrice = activeElement.innerHTML == parsePrice(parseFloat(activeElement.innerHTML));

if (!isNaN(value) && (e.keyCode == 38 || e.keyCode == 40 || e.wheelDeltaY)) {

e.preventDefault();

value += e.keyCode == 38 ? 1 : e.keyCode == 40 ? -1 : Math.round(e.wheelDelta * 0.025);

value = Math.max(value, 0);

activeElement.innerHTML = wasPrice ? parsePrice(value) : value;

}

updateInvoice();

}

/* Update Invoice

/* ===== */

```

```

function updateInvoice() {

var total = 0;

var cells, price, total, a, i;


// update inventory cells
// =====

for (var a = document.querySelectorAll('table.inventory tbody tr'), i = 0; a[i]; ++i) {

// get inventory row cells
cells = a[i].querySelectorAll('span:last-child');


// set price as cell[2] * cell[3]
price = parseFloatHTML(cells[2]) * parseFloatHTML(cells[3]);


// add price to total
total += price;


// set row total
cells[4].innerHTML = price;
}


// update balance cells
// =====


// get balance cells

```



```

cells = document.querySelectorAll('table.balance td:last-child span:last-child');

// set total

cells[0].innerHTML = total;

// set balance and meta balance

cells[2].innerHTML = document.querySelector('table.meta tr:last-child td:last-child span:last-child').innerHTML = parsePrice(total - parseFloatHTML(cells[1]));

// update prefix formatting

// =====

var prefix = document.querySelector('#prefix').innerHTML;

for (a = document.querySelectorAll('[data-prefix]'), i = 0; a[i]; ++i) a[i].innerHTML = prefix;

// update price formatting

// =====

for (a = document.querySelectorAll('span[data-prefix] + span'), i = 0; a[i]; ++i) if
(document.activeElement != a[i]) a[i].innerHTML = parsePrice(parseFloatHTML(a[i]));

}

/* On Content Load

/* ===== */

function onContentLoaded() {

```

```

updateInvoice();

var
input = document.querySelector('input'),
image = document.querySelector('img');

function onClick(e) {
var element = e.target.querySelector('[contenteditable]'), row;

element && e.target != document.documentElement && e.target != document.body &&
element.focus();

if (e.target.matchesSelector('.add')) {
document.querySelector('table.inventory tbody').appendChild(generateTableRow());
}
else if (e.target.className == 'cut') {
row = e.target.ancestorQuerySelector('tr');

row.parentNode.removeChild(row);
}

updateInvoice();
}

function onEnterCancel(e) {
e.preventDefault();

```

```
image.classList.add('hover');  
}
```

```
function onLeaveCancel(e) {  
    e.preventDefault();
```

```
    image.classList.remove('hover');  
}
```

```
function onFileInput(e) {  
    image.classList.remove('hover');
```

```
    var  
    reader = new FileReader(),  
    files = e.dataTransfer ? e.dataTransfer.files : e.target.files,  
    i = 0;
```

```
    reader.onload = onFileLoad;
```

```
    while (files[i]) reader.readAsDataURL(files[i++]);  
}
```

```
function onFileLoad(e) {  
    var data = e.target.result;
```

```
image.src = data;

}

if (window.addEventListener) {

document.addEventListener('click', onClick);

document.addEventListener('mousewheel', updateNumber);
document.addEventListener('keydown', updateNumber);

document.addEventListener('keydown', updateInvoice);
document.addEventListener('keyup', updateInvoice);

input.addEventListener('focus', onEnterCancel);
input.addEventListener('mouseover', onEnterCancel);
input.addEventListener('dragover', onEnterCancel);
input.addEventListener('dragenter', onEnterCancel);

input.addEventListener('blur', onLeaveCancel);
input.addEventListener('dragleave', onLeaveCancel);
input.addEventListener('mouseout', onLeaveCancel);

input.addEventListener('drop', onFileInput);
input.addEventListener('change', onFileInput);

}
```

```
}
```

```
window.addEventListener && document.addEventListener('DOMContentLoaded', onContentLoad);
```

- Display User Using Express REST API

Install node first

node-- version

Now go to the postman API

create a folder and name it node express

Now type NPM init

NPM install-- save express

index.js

```
const express = require("express");
```

```
const app = express();
```

```
app.use(express.json());
```

```
app.use(express.urlencoded({ extended: false }));
```

```
app.use("/api/users", require("./routes/api/users"));
```

```
app.listen(3000, () => console.log('Server started'));
```

Users.js

```
const users = [  
  
  {  
  
    id: 1,  
  
    name: "John",  
  
    email: "john@gmail.com"  
  
  },  
  
  {  
  
    id: 2,  
  
    name: "Smith",  
  
    email: "smith@gmail.com"  
  
  },  
  
  {  
  
    id: 3,  
  
    name: "Chris",  
  
    email: "chris@gmail.com"  
  
  },  
  
  {
```

```
    id: 4,  
  
    name: "Jack",  
  
    email: "jack@gmail.com"  
  
  }  
  
];  
  
module.exports = users;
```

User.js

In this file, we add the API routes that we want to include in the application. First, we'll create a folder named routes, and inside that, create another folder named api. After creating the folders, add a new file and name it users.js

```
const express = require("express");  
  
const router = express.Router();  
  
const uuid = require("uuid");  
  
let users = require("../Users");  
  
  
router.get("/", (req, res) => {  
  
  res.json(users);  
  
});
```

```
});
```

```
router.get("/:id", (req, res) => {
```

```
  const found = users.some(user => user.id === parseInt(req.params.id));
```

```
  if (found) {
```

```
    res.json(users.filter(user => user.id === parseInt(req.params.id)));
```

```
  } else {
```

```
    res.sendStatus(400);
```

```
  }
```

```
});
```

```
router.post("/", (req, res) => {
```

```
  const newUser = {
```

```
    id: uuid.v4(),
```

```
    name: req.body.name,
```

```
    email: req.body.email
```

```
  };
```

```
  if (!newUser.name || !newUser.email) {
```

```
    return res.sendStatus(400);
```

```
  }
```



```

    users.push(newUser);

    res.json(users);

  });

//Update User

router.put("/:id", (req, res) => {

  const found = users.some(user => user.id === parseInt(req.params.id));

  if (found) {

    const updateUser = req.body;

    users.forEach(user => {

      if (user.id === parseInt(req.params.id)) {

        user.name = updateUser.name ? updateUser.name : user.name;

        user.email = updateUser.email ? updateUser.email : user.email;

        res.json({ msg: "User updated", user });

      }

    });

  } else {

    res.sendStatus(400);

  }

});

//Delete User

router.delete("/:id", (req, res) => {

```

```

const found = users.some(user => user.id === parseInt(req.params.id))

if (found) {

  users = users.filter(user => user.id !== parseInt(req.params.id))

  res.json({

    msg: "User deleted",

    users

  });

} else {

  res.sendStatus(400);

}

});

module.exports = router;

```

Postman

Postman is an intuitive and powerful application that web developers use to check if the API requests in their applications are working properly.

- Conclusively, we use Postman for API calls:
 - GET to retrieve data: <http://localhost:3000/api/users/>
 - POST with JSON body to add a user: <http://localhost:3000/api/users/>
 - PUT with JSON body to update data for user-id: <http://localhost:3000/api/users/1>
 - DELETE to delete a user: <http://localhost:3000/api/users/1>

All API routes should work as expected.

If you have any questions for us, please leave them in the comment section below and we'll have our experts get back to you promptly.