1.<!DOCTYPE html>

<html>

<style>

.all-browsers {

margin: 0;

padding: 5px;

background-color: lightgray;

}

.all-browsers > h1, .browser {

margin: 10px;

padding: 5px;

}

.browser {

background: white;

}

.browser > h2, p {

margin: 4px;

font-size: 90%;

}

aside {

width: 30%;

padding-left: 15px;

margin-left: 15px;

float: right;

font-style: italic;

background-color: lightgray;

}</style>

<body>

<nav>

<a href="/html/">HTML</a> |

<a href="/css/">CSS</a> |

<a href="/js/">JavaScript</a> |

<a href="/jquery/">jQuery</a>

</nav>

<section>

<h1>WWF</h1>

<p>The World Wide Fund for Nature (WWF) is an international organization working on issues

regarding the conservation, research and restoration of the environment, formerly named the World

Wildlife Fund. WWF was founded in 1961.</p>

</section>

<section>

<h1>WWF's Panda symbol</h1>

<p>The Panda has become the symbol of WWF. The well-known panda logo of WWF originated

from a panda named Chi Chi that was transferred from the Beijing Zoo to the London Zoo in the

same year of the establishment of WWF.</p>

</section>

<br>

<article class="all-browsers">

<h1>Most Popular Browsers</h1>

<article class="browser">

<h2>Google Chrome</h2>

<p>Google Chrome is a web browser developed by Google, released in 2008. Chrome is the

world's most popular web browser today!</p>

</article>

<article class="browser">

<h2>Mozilla Firefox</h2>

<p>Mozilla Firefox is an open-source web browser developed by Mozilla. Firefox has been the

second most popular web browser since January, 2018.</p>

</article>

<article class="browser"><h2>Microsoft Edge</h2>

<p>Microsoft Edge is a web browser developed by Microsoft, released in 2015. Microsoft Edge

replaced Internet Explorer.</p>

</article>

</article>

<br>

<p>My family and I visited The Epcot center this summer. The weather was nice, and Epcot was

amazing! I had a great summer together with my family!</p>

<aside>

<p>The Epcot center is a theme park at Walt Disney World Resort featuring exciting attractions,

international pavilions, award-winning fireworks and seasonal special events.</p>

</aside>

<p>My family and I visited The Epcot center this summer. The weather was nice, and Epcot was

amazing! I had a great summer together with my family!</p>

</body>

</html>

2.<html>

<body>

<h1>My Header</h1>

<p>My paragraph.</p>

</body>

</html>

Demo.js

var http = require('http');

var fs = require('fs');

fs.appendFile('mynewfile1.txt', 'Hello content!', function (err) {

if (err) throw err;

console.log('Saved!');

});

fs.open('mynewfile2.txt', 'w', function (err, file) {

if (err) throw err;

console.log('Saved!');

});

fs.writeFile('mynewfile3.txt', 'Hello content!', function (err) {

if (err) throw err;

console.log('Saved!');

});

fs.appendFile('mynewfile1.txt', ' This is my text.', function (err) {

if (err) throw err;

console.log('Updated!');

});

fs.writeFile('mynewfile3.txt', 'This is my text', function (err) {

if (err) throw err;

console.log('Replaced!');

});

fs.unlink('mynewfile2.txt', function (err) {

if (err) throw err;

console.log('File deleted!');

});

fs.rename('mynewfile1.txt', 'myrenamedfile.txt', function (err) {

if (err) throw err;console.log('File Renamed!');

});

http.createServer(function (req, res) {

fs.readFile('demofile1.html', function(err, data) {

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

res.write(data);

return res.end();

});

}).listen(8080);

3.MongoClient.connect(url, (err, client) => {

if (err) {

console.error('Error connecting to MongoDB:', err);

return;

}

console.log('Connected to MongoDB successfully');

const db = client.db(dbName);

// Perform CRUD operations here

client.close();

});

*Perform CRUD operations within the callback function:*

*Create operation:*

const collection = db.collection('your-collection-name');

const document = { name: 'John Doe', age: 30 };

collection.insertOne(document, (err, result) => {

if (err) {

console.error('Error creating document:', err);

return;}

console.log('Document created successfully:', result.insertedId);

});

*Read operation:*

const collection = db.collection('your-collection-name');

collection.find({}).toArray((err, documents) => {

if (err) {

console.error('Error reading documents:', err);

return;

}

console.log('Documents:', documents);

});

*Update operation:*

const collection = db.collection('your-collection-name');

const filter = { name: 'John Doe' };

const update = { $set: { age: 35 } };

collection.updateOne(filter, update, (err, result) => {

if (err) {

console.error('Error updating document:', err);

return;

}

console.log('Document updated successfully');

});

*Delete operation:*

const collection = db.collection('your-collection-name');

const filter = { name: 'John Doe' };

collection.deleteOne(filter, (err, result) => {

if (err) {

console.error('Error deleting document:', err);

return;

}

console.log('Document deleted successfully');

});

4.function getData() {

return new Promise((resolve, reject) => {

setTimeout(() => {

const data = 'Async data';

resolve(data);

}, 2000);

});

}

// Async function using the await keyword

async function processData() {

try {

console.log('Fetching data...');

const result = await getData();

console.log('Data:', result);

console.log('Data processed successfully!');

} catch (error) {

console.error('Error:', error);

}

}

// Call the async function

processData();

5.npx create-react-app todo-list-app

cd todo-list-app

npm install bootstrap

import React, { Component } from 'react';

class TodoListClass extends Component {

constructor(props) {

super(props);

this.state = {

todos: [],

newTodo: '',

};

}

handleInputChange = (event) => {

this.setState({ newTodo: event.target.value });

};

handleAddTodo = () => {

if (this.state.newTodo.trim() !== '') {

this.setState((prevState) => ({

todos: [...prevState.todos, this.state.newTodo],

newTodo: '',

}));

}

};

handleDeleteTodo = (index) => {

this.setState((prevState) => ({

todos: prevState.todos.filter((\_, i) => i !== index),

}));

};

render() {

const { todos, newTodo } = this.state;

return (

<div>

<h1>Todo List (Class-based)</h1>

<div className="input-group mb-3">

<input

type="text"

className="form-control"

placeholder="Add new todo"value={newTodo}

onChange={this.handleInputChange}

/>

<div className="input-group-append">

<button className="btn btn-primary" onClick={this.handleAddTodo}>

Add

</button>

</div>

</div>

<ul className="list-group">

{todos.map((todo, index) => (

<li className="list-group-item" key={index}>

{todo}

<button

className="btn btn-sm btn-danger float-right"

onClick={() => this.handleDeleteTodo(index)}

>

Delete

</button>

</li>

))}

</ul>

</div>

);

}

}

export default TodoListClass;

import React, { useState } from 'react';

const TodoListFunction = () => {

const [todos, setTodos] = useState([]);

const [newTodo, setNewTodo] = useState('');

const handleInputChange = (event) => {

setNewTodo(event.target.value);

};

const handleAddTodo = () => {

if (newTodo.trim() !== '') {

setTodos([...todos, newTodo]);

setNewTodo('');

}

};

const handleDeleteTodo = (index) => {

setTodos(todos.filter((\_, i) => i !== index));

};return (

<div>

<h1>Todo List (Functional)</h1>

<div className="input-group mb-3">

<input

type="text"

className="form-control"

placeholder="Add new todo"

value={newTodo}

onChange={handleInputChange}

/>

<div className="input-group-append">

<button className="btn btn-primary" onClick={handleAddTodo}>

Add

</button>

</div>

</div>

<ul className="list-group">

{todos.map((todo, index) => (

<li className="list-group-item" key={index}>

{todo}

<button

className="btn btn-sm btn-danger float-right"

onClick={() => handleDeleteTodo(index)}

>

Delete

</button>

</li>

))}

</ul>

</div>

);

};

export default TodoListFunction;

import React from 'react';

import './App.css';

import TodoListClass from './components/TodoListClass';

import TodoListFunction from './components/TodoListFunction';

function App() {

return (

<div className="container">

<div className="row">

<div className="col">

<TodoListClass />

</div>

<div className="col">

<TodoListFunction /></div>

</div>

</div>

);

}

export default App;