Question 1: **DSA** Given an array of integers nums and an integer target, return the indices of the two numbers such that they add up to target. You may assume that each input would have exactly one solution, and you may not use the same element twice. You can return the answer in any order.

For example, given:

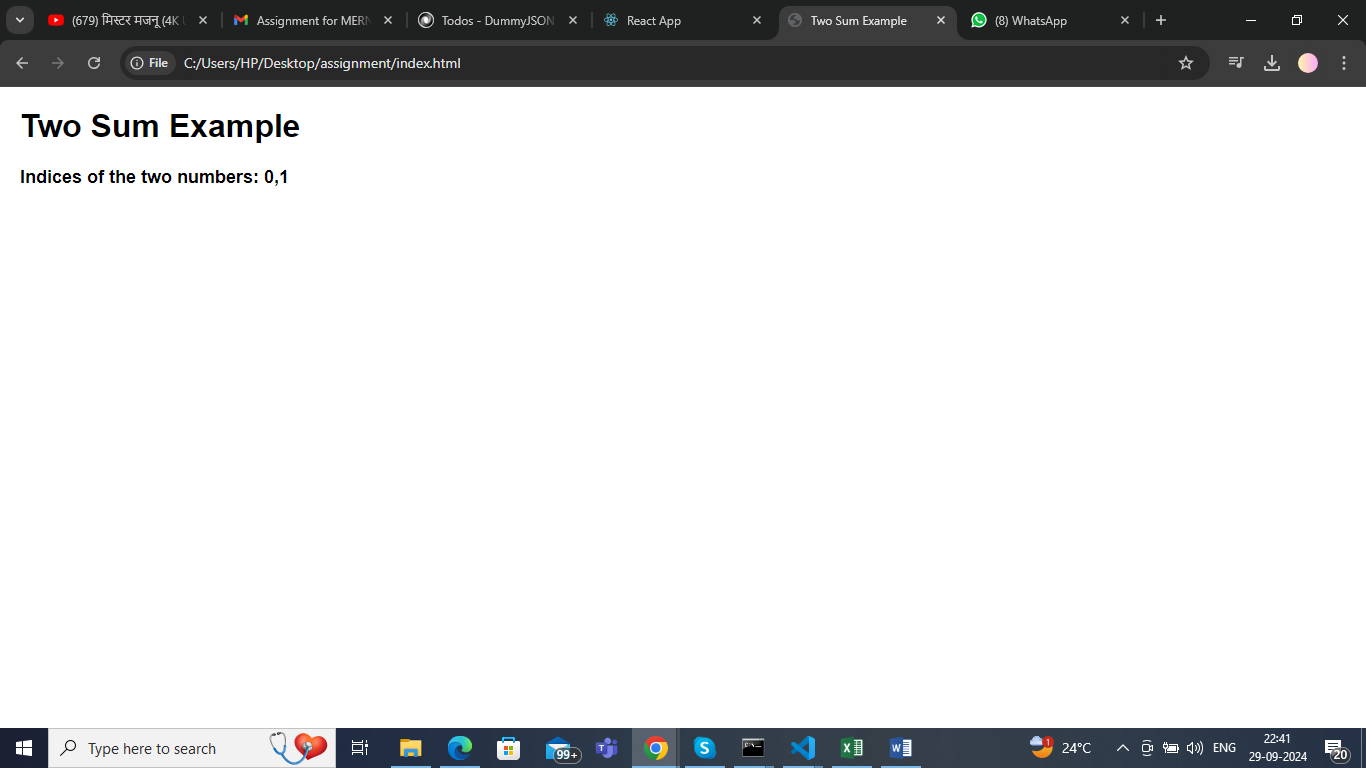
const nums = [2, 7, 11, 15];

const target = 9;

The function should return [0, 1] because nums[0] + nums[1] = 2 + 7 = 9.

**Requirements:**

* Implement the solution in JavaScript.
* The solution should have a time complexity better than O(n^2).
* Include proper error handling for edge cases.



Question 2:

db.sales.aggregate([

{

// Stage 1: Unwind the items array to create a separate document for each item

$unwind: "$items"

},

{

// Stage 2: Group by store and month

$group: {

\_id: {

store: "$store",

month: { $dateToString: { format: "%Y-%m", date: "$date" } } // Format date to "YYYY-MM"

},

totalRevenue: {

$sum: { $multiply: ["$items.quantity", "$items.price"] } // Calculate total revenue for each group

},

averagePrice: {

$avg: "$items.price" // Calculate average price of items sold

}

}

},

{

// Stage 3: Project the output fields to format the final result

$project: {

\_id: 0, // Exclude the default \_id field

store: "$\_id.store",

month: "$\_id.month",

totalRevenue: 1,

averagePrice: 1

}

},

{

// Stage 4: Sort by store and month

$sort: {

store: 1, // Ascending order by store name

month: 1 // Ascending order by month

}

}

]);

Output

[

{

"store": "Store A",

"month": "2024-06",

"totalRevenue": 230.0,

"averagePrice": 15.0

},

{

"store": "Store B",

"month": "2024-06",

"totalRevenue": 150.0,

"averagePrice": 12.5

}

]

Question 3: **React** Build a Dynamic To-Do List Application in React.

