

TASK1

Task : // Nested array data for customers - details Id, name, place, email, amount

```
const customerData = [  
  [3000, 'John Doe', 'New York', 'john.doe@email.com', 1000],  
  [3001, 'Alice Smith', 'Los Angeles', 'alice.smith@email.com', 1200],  
  [3002, 'Bob Johnson', 'Chicago', 'bob.johnson@email.com', 800],  
  [3003, 'Emily Davis', 'San Francisco', 'emily.davis@email.com', 1500],  
  [3004, 'David Lee', 'Houston', 'david.lee@email.com', 1100],  
  [3005, 'Sophia Chen', 'Miami', 'sophia.chen@email.com', 900],  
  [3006, 'Michael Wilson', 'Seattle', 'michael.wilson@email.com', 1300],  
  [3007, 'Emma Brown', 'Dallas', 'emma.brown@email.com', 950],  
  [3008, 'Daniel Miller', 'Boston', 'daniel.miller@email.com', 1050],  
];  
  
// Questions for the customer data  
  
// 1. Print the names of all customers.
```

The screenshot shows the VS Code editor with a file named `nesttask.js` open. The file contains a `for` loop that iterates over `customerData` and logs the names of all customers. The terminal output shows the following names: John Doe, Alice Smith, Bob Johnson, Emily Davis, David Lee, Sophia Chen, and Michael Wilson.

```
12 }
13
14
15 // Questions for the customer data
16 // 1. Print the names of all customers.
17 // 2. How many customers are there in the dataset?
18 // 3. Provide details for customers from the city "Chicago."
19 // 4. Display details for customers with a transaction amount greater than 1000.
20 // 5. Retrieve the details of the customer named "David Lee."
21 // 6. Sort customers in descending order based on their transaction amount.
22 // 7. Arrange customers in ascending order based on their customer ID.
23 // 8. How many customers have a transaction amount between 900 and 1100?
24 // 9. Provide details for customers who are not from New York.
25
26
27 // 1. Print the names of all customers.
28
29 for(i of customerData){
30     names=i[1];
31     console.log(names);
32 }
33
```

Terminal Output:

```
PS C:\Users\Philip\Desktop\meann luminartechlab\JAVASCRIPT\DAY10> node nesttask.js
John Doe
Alice Smith
Bob Johnson
Emily Davis
David Lee
Sophia Chen
Michael Wilson
```

// 2. How many customers are there in the dataset?

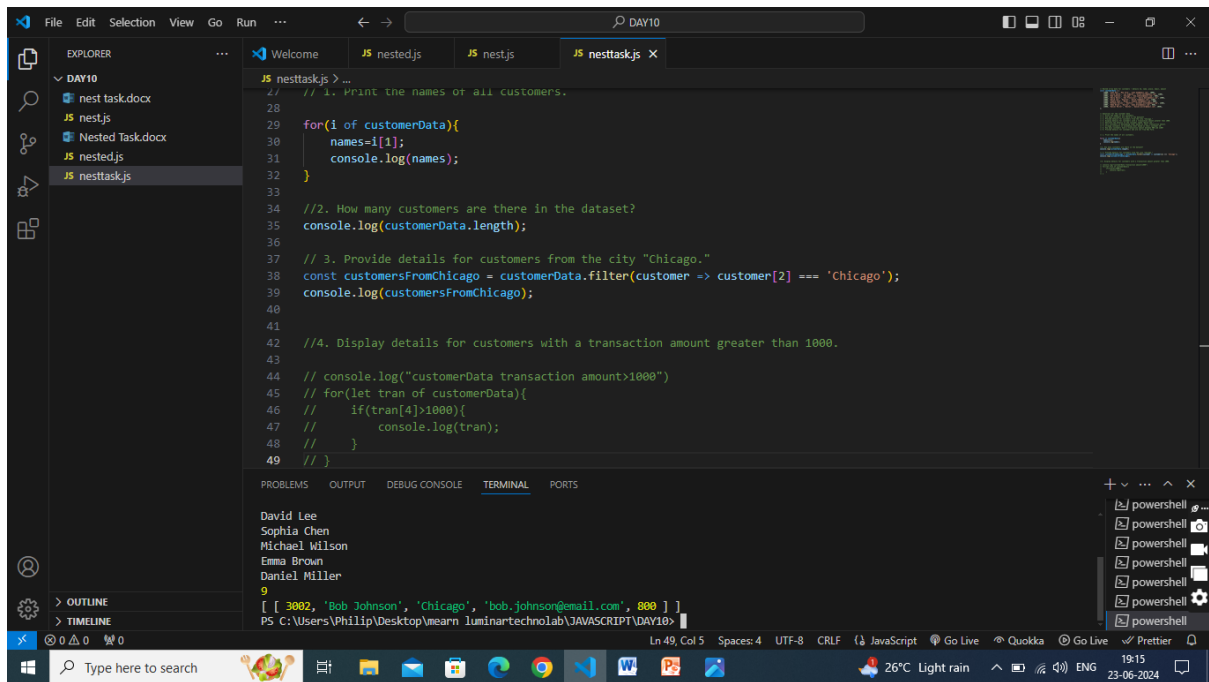
The screenshot shows the VS Code editor with the same file `nesttask.js`. A new `console.log` statement has been added to log the length of `customerData`. The terminal output now includes the number 9, which represents the total number of customers.

```
19 // 4. Display details for customers with a transaction amount greater than 1000.
20 // 5. Retrieve the details of the customer named "David Lee."
21 // 6. Sort customers in descending order based on their transaction amount.
22 // 7. Arrange customers in ascending order based on their customer ID.
23 // 8. How many customers have a transaction amount between 900 and 1100?
24 // 9. Provide details for customers who are not from New York.
25
26
27 // 1. Print the names of all customers.
28
29 for(i of customerData){
30     names=i[1];
31     console.log(names);
32 }
33
34 //2. How many customers are there in the dataset?
35 console.log(customerData.length);
```

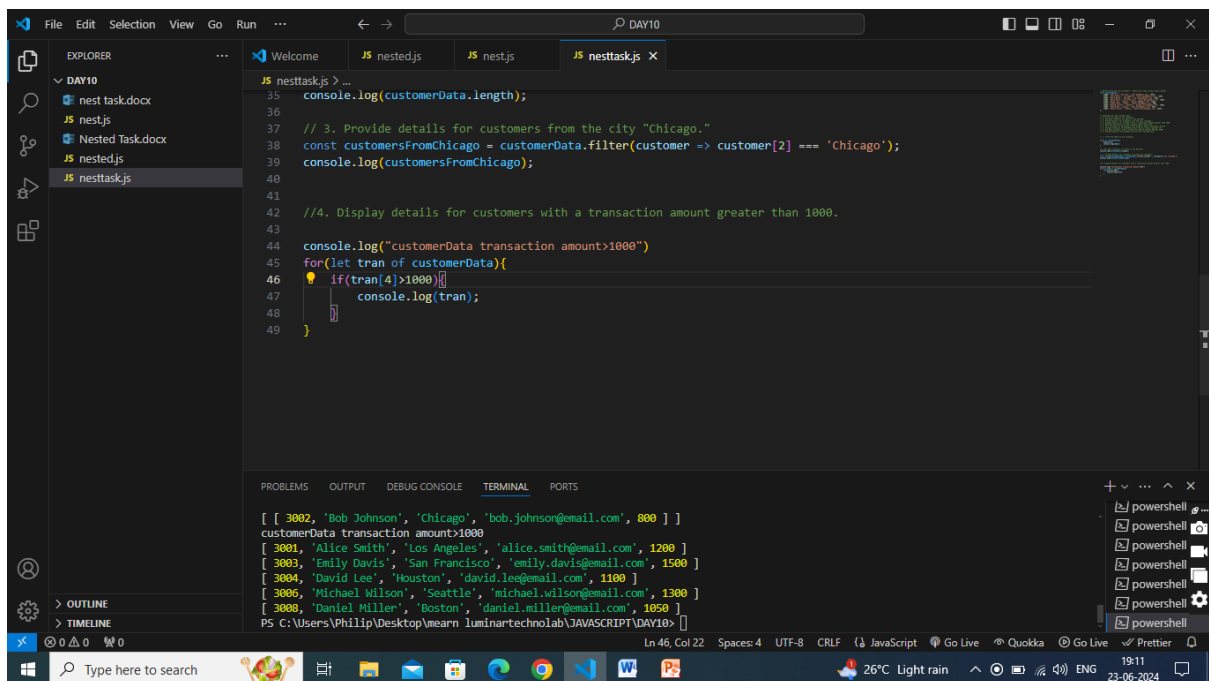
Terminal Output:

```
Emily Davis
David Lee
Sophia Chen
Michael Wilson
Emma Brown
Daniel Miller
9
```

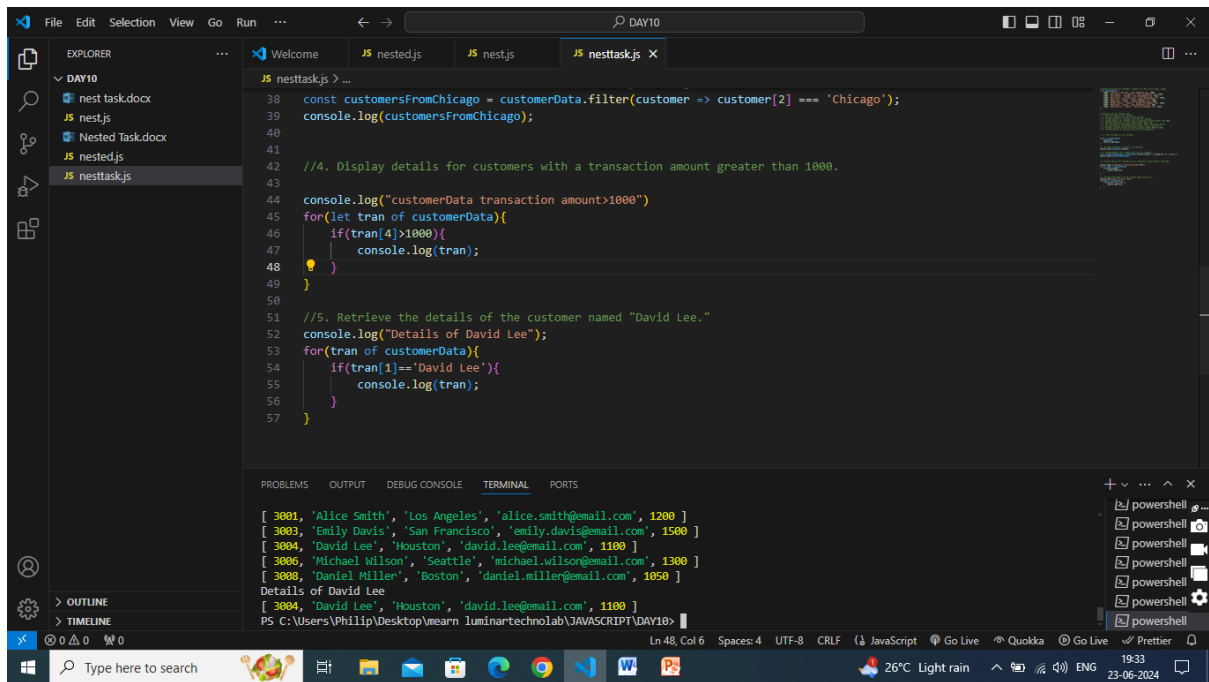
// 3. Provide details for customers from the city "Chicago."



// 4. Display details for customers with a transaction amount greater than 1000.



// 5. Retrieve the details of the customer named "David Lee."



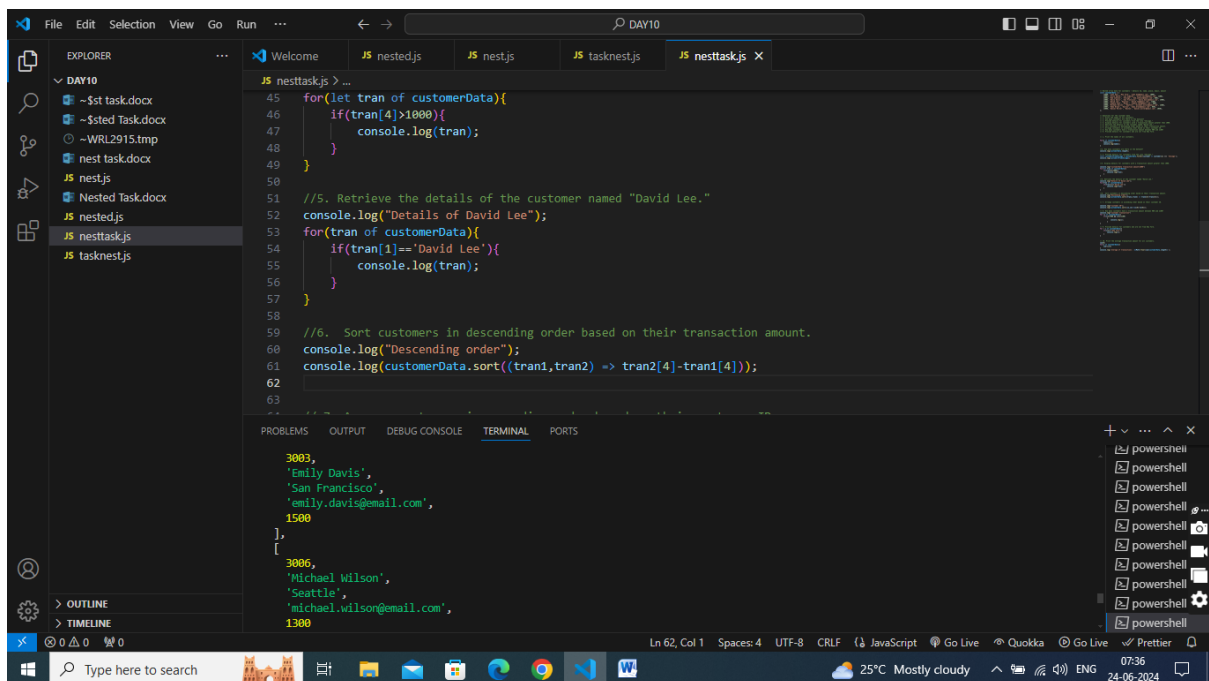
```
File Edit Selection View Go Run ... DAY10
EXPLORER
  DAY10
    nest task.docx
    nest.js
    Nested Task.docx
    nest.js
    nesttask.js
  PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
  > OUTLINE
  > TIMELINE
  PS C:\Users\Philip\Desktop\mearn_luminartechnolab\JAVASCRIPT\DAY10>

38 const customersFromChicago = customerData.filter(customer => customer[2] === 'Chicago');
39 console.log(customersFromChicago);
40
41
42 //4. Display details for customers with a transaction amount greater than 1000.
43
44 console.log("customerData transaction amount>1000")
45 for(let tran of customerData){
46   if(tran[4]>1000){
47     console.log(tran);
48   }
49 }
50
51 //5. Retrieve the details of the customer named "David Lee."
52 console.log("Details of David Lee.");
53 for(tran of customerData){
54   if(tran[1]== 'David Lee'){
55     console.log(tran);
56   }
57 }
```

Terminal Output:

```
[ 3001, 'Alice Smith', 'Los Angeles', 'alice.smith@email.com', 1200 ]
[ 3003, 'Emily Davis', 'San Francisco', 'emily.davis@email.com', 1500 ]
[ 3004, 'David Lee', 'Houston', 'david.lee@email.com', 1100 ]
[ 3006, 'Michael Wilson', 'Seattle', 'michael.wilson@email.com', 1300 ]
[ 3008, 'Daniel Miller', 'Boston', 'daniel.miller@email.com', 1050 ]
Details of David Lee
[ 3004, 'David Lee', 'Houston', 'david.lee@email.com', 1100 ]
PS C:\Users\Philip\Desktop\mearn_luminartechnolab\JAVASCRIPT\DAY10>
```

// 6. Sort customers in descending order based on their transaction amount.



```
File Edit Selection View Go Run ... DAY10
EXPLORER
  DAY10
    nest task.docx
    nest.js
    Nested Task.docx
    nest.js
    nesttask.js
    tasknest.js
  PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
  > OUTLINE
  > TIMELINE
  PS C:\Users\Philip\Desktop\mearn_luminartechnolab\JAVASCRIPT\DAY10>

45 for(let tran of customerData){
46   if(tran[4]>1000){
47     console.log(tran);
48   }
49 }
50
51 //5. Retrieve the details of the customer named "David Lee."
52 console.log("Details of David Lee.");
53 for(tran of customerData){
54   if(tran[1]== 'David Lee'){
55     console.log(tran);
56   }
57 }
58
59 //6. Sort customers in descending order based on their transaction amount.
60 console.log("Descending order");
61 console.log(customerData.sort((tran1,tran2) => tran2[4]-tran1[4]));
62
63 }
```

Terminal Output:

```
3003,
'Emily Davis',
'San Francisco',
'emily.davis@email.com',
1500
],
[
3006,
'Michael Wilson',
'Seattle',
'michael.wilson@email.com',
1300
]
```

// 7. Arrange customers in ascending order based on their customer ID.

```
58
59 //6. Sort customers in descending order based on their transaction amount.
60 console.log("Descending order");
61 customerData.sort((tran1,tran2) => tran2[4]-tran1[4]);
62 console.log(customerData);
63
64 // 7. Arrange customers in ascending order based on their customer ID.
65
66 console.log("customer ID");
67 console.log(customerData.sort((c1,c2)->c1[0]-c2[0]));
68
69 // 8. How many customers have a transaction amount between 900 and 1100?
70 console.log("customer transaction")
71 for (i of customerData){
72     if(i[4]>900 && i[4]<1100)
73     {
74         console.log(i);
75     }
76 }
77
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
[ 3005, 'Sophia Chen', 'Miami', 'sophia.chen@email.com', 900 ],
[ 3002, 'Bob Johnson', 'Chicago', 'bob.johnson@email.com', 800 ]
]
customer ID
[ 3000, 'John Doe', 'New York', 'john.doe@email.com', 1000 ],
[ 3001, 'Alice Smith', 'Los Angeles', 'alice.smith@email.com', 1200 ],
[ 3002, 'Bob Johnson', 'Chicago', 'bob.johnson@email.com', 800 ],
[
  3003,
  'Emily Davis',
  'San Francisco',
  1500
]
```

// 8. How many customers have a transaction amount between 900 and 1100?

```
58
59 //6. Sort customers in descending order based on their transaction amount.
60 console.log("Descending order");
61 customerData.sort((tran1,tran2) => tran2[4]-tran1[4]);
62 console.log(customerData);
63
64 // 7. Arrange customers in ascending order based on their customer ID.
65
66 console.log("customer ID");
67 const sortByIDAscending = [...customerData].sort((a, b) => a[0] - b[0]);
68 console.log("Customers sorted by ID (ascending):", sortByIDAscending);
69
70 // 8. How many customers have a transaction amount between 900 and 1100?
71 console.log("customer transaction")
72 for (i of customerData){
73     if(i[4]>900 && i[4]<1100)
74     {
75         console.log(i);
76     }
77 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
],
[ 3007, 'Emma Brown', 'Dallas', 'emma.brown@email.com', 950 ],
[ 3008, 'Daniel Miller', 'Boston', 'daniel.miller@email.com', 1050 ]
]
customer transaction
[ 3008, 'Daniel Miller', 'Boston', 'daniel.miller@email.com', 1050 ]
[ 3000, 'John Doe', 'New York', 'john.doe@email.com', 1000 ]
[ 3007, 'Emma Brown', 'Dallas', 'emma.brown@email.com', 950 ]
[ 3003, 'Emily Davis', 'San Francisco', 'emily.davis@email.com', 1500 ]
[ 3006, 'Michael Wilson', 'Seattle', 'michael.wilson@email.com', 1300 ]
[ 3001, 'Alice Smith', 'Los Angeles', 'alice.smith@email.com', 1200 ]
[ 3004, 'David Lee', 'Houston', 'david.lee@email.com', 1100 ]
```

// 9. Provide details for customers who are not from New York.

```
JS nesttaskjs > ...
64 // 7. Arrange customers in ascending order based on their customer ID.
65
66 console.log("customer ID");
67 const sortedByIdAscending = [...customerData].sort((a, b) => a[0] - b[0]);
68 console.log("Customers sorted by ID (ascending):", sortedByIdAscending);
69
70 // 8. How many customers have a transaction amount between 900 and 1100?
71
72
73 // 9. Provide details for customers who are not from New York.
74 for (i of customerData){
75     if(i[2] != 'New York'){
76         console.log(i);
77     }
78 }
79
80
81 // 10. Print the average transaction amount for all customers.
82 sum=0;

[ 3008, 'Daniel Miller', 'Boston', 'daniel.miller@email.com', 1050 ]
[ 3003, 'Emily Davis', 'San Francisco', 'emily.davis@email.com', 1500 ]
[ 3006, 'Michael Wilson', 'Seattle', 'michael.wilson@email.com', 1300 ]
[ 3001, 'Alice Smith', 'Los Angeles', 'alice.smith@email.com', 1200 ]
[ 3004, 'David Lee', 'Houston', 'david.lee@email.com', 1100 ]
[ 3008, 'Daniel Miller', 'Boston', 'daniel.miller@email.com', 1050 ]
[ 3007, 'Emma Brown', 'Dallas', 'emma.brown@email.com', 950 ]
[ 3005, 'Sophia Chen', 'Miami', 'sophia.chen@email.com', 900 ]
[ 3002, 'Bob Johnson', 'Chicago', 'bob.johnson@email.com', 800 ]
Average of transactions : 1088
PS C:\Users\Philip\Desktop\mearn_luminartechnolab\JAVASCRIPT\DAY10>
```

// 10. Print the average transaction amount for all customers.

```
JS nesttaskjs > ...
74
75
76
77 // 10. Print the average transaction amount for all customers.
78 sum=0;
79 for(i of customerData){
80     sum+=i[4]
81 }
82 console.log("Average of transactions : ${Math.floor(sum/customerData.length)}");

[
  3006,
  'Michael Wilson',
  'Seattle',
  'michael.wilson@email.com',
  1300
],
[ 3007, 'Emma Brown', 'Dallas', 'emma.brown@email.com', 950 ],
[ 3008, 'Daniel Miller', 'Boston', 'daniel.miller@email.com', 1050 ]
]
Average of transactions : 1088
PS C:\Users\Philip\Desktop\mearn_luminartechnolab\JAVASCRIPT\DAY10>
```