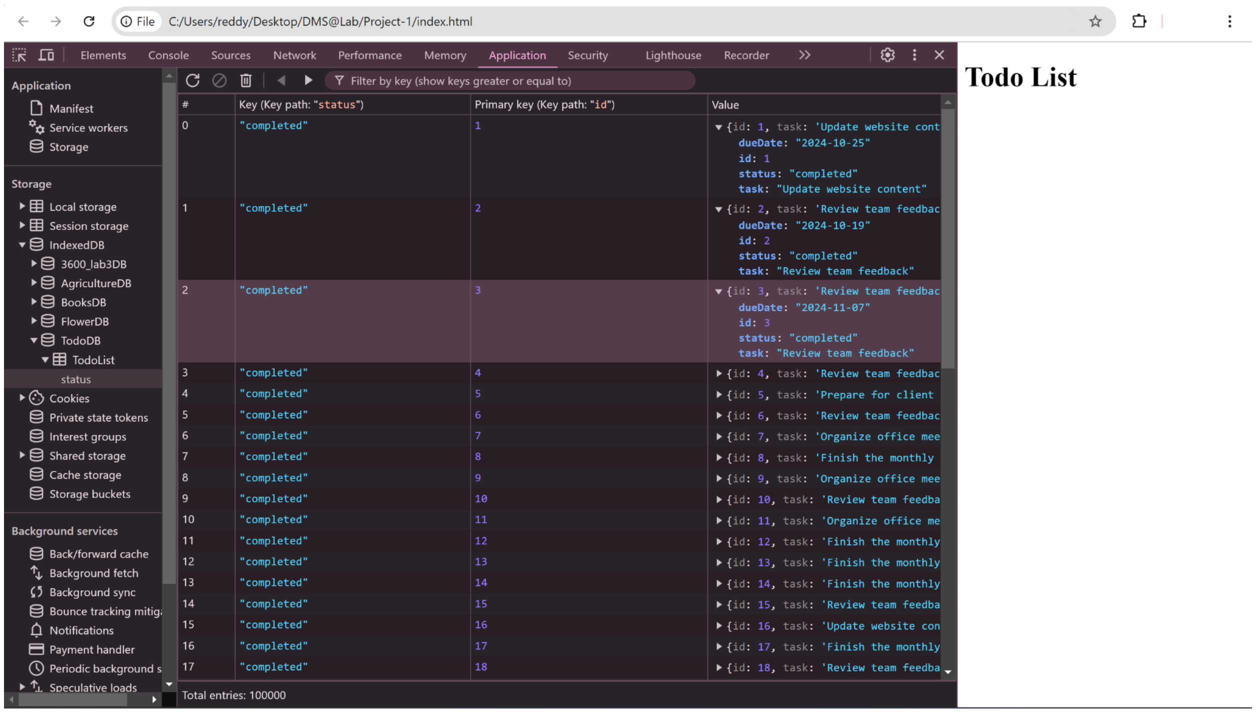
Team - 10 - Punam - Konika - Shivam

Punam Ram Pukale	: 989469907
Konika Reddy Saddikuti	: 989473600
Shivam Lnu	: 989469265

Task-1

Define an **IndexedDB** object store named "TodoList" and populate it with 100,000 randomly generated objects, each containing `task`, `status`, and `dueDate` properties and display it on the console or the browser.



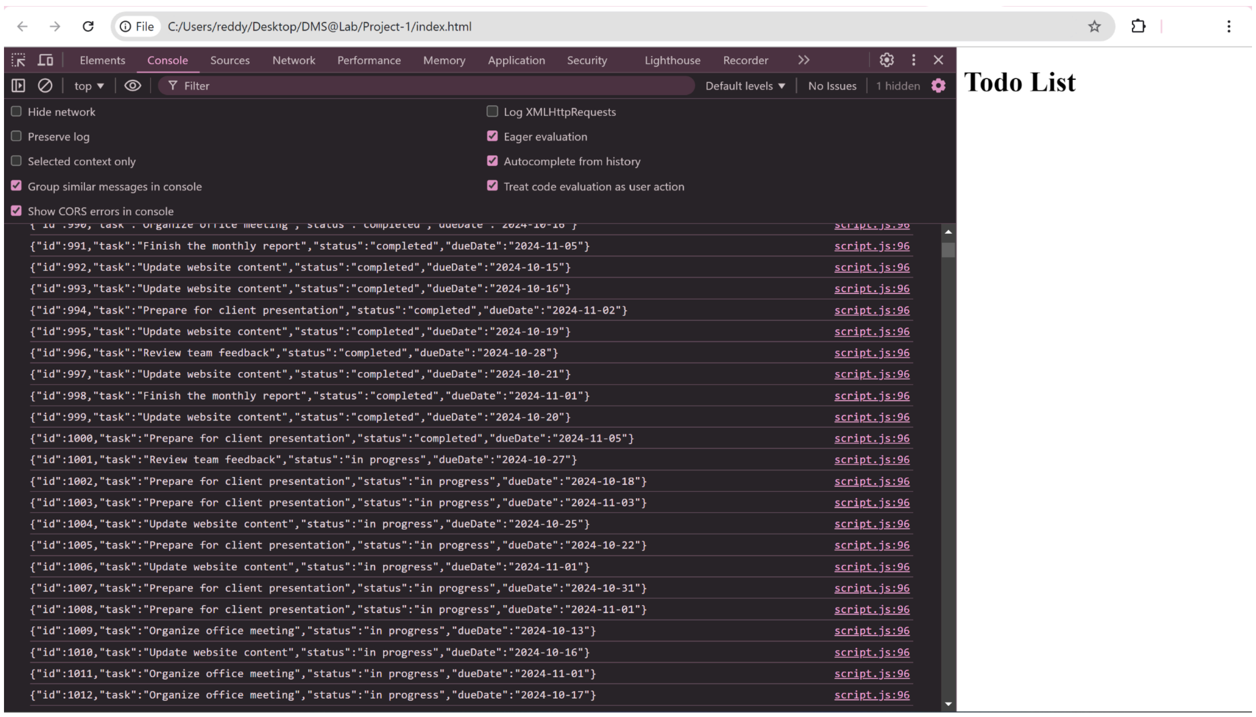
#	Key (Key path: "status")	Primary key (Key path: "id")	Value
0	"completed"	1	{id: 1, task: 'Update website cont dueDate: "2024-10-25" id: 1 status: "completed" task: "Update website content"
1	"completed"	2	{id: 2, task: 'Review team feedbac dueDate: "2024-10-19" id: 2 status: "completed" task: "Review team feedback"
2	"completed"	3	{id: 3, task: 'Review team feedbac dueDate: "2024-11-07" id: 3 status: "completed" task: "Review team feedback"
3	"completed"	4	{id: 4, task: 'Review team feedbac
4	"completed"	5	{id: 5, task: 'Prepare for client
5	"completed"	6	{id: 6, task: 'Review team feedbac
6	"completed"	7	{id: 7, task: 'Organize office mee
7	"completed"	8	{id: 8, task: 'Finish the monthly
8	"completed"	9	{id: 9, task: 'Organize office mee
9	"completed"	10	{id: 10, task: 'Review team feedba
10	"completed"	11	{id: 11, task: 'Organize office me
11	"completed"	12	{id: 12, task: 'Finish the monthly
12	"completed"	13	{id: 13, task: 'Finish the monthly
13	"completed"	14	{id: 14, task: 'Finish the monthly
14	"completed"	15	{id: 15, task: 'Review team feedba
15	"completed"	16	{id: 16, task: 'Update website con
16	"completed"	17	{id: 17, task: 'Finish the monthly
17	"completed"	18	{id: 18, task: 'Review team feedba

Total entries: 100000

Todo List

Task-2

Set 1000 objects to status "completed" and the remaining ones to status "progress"

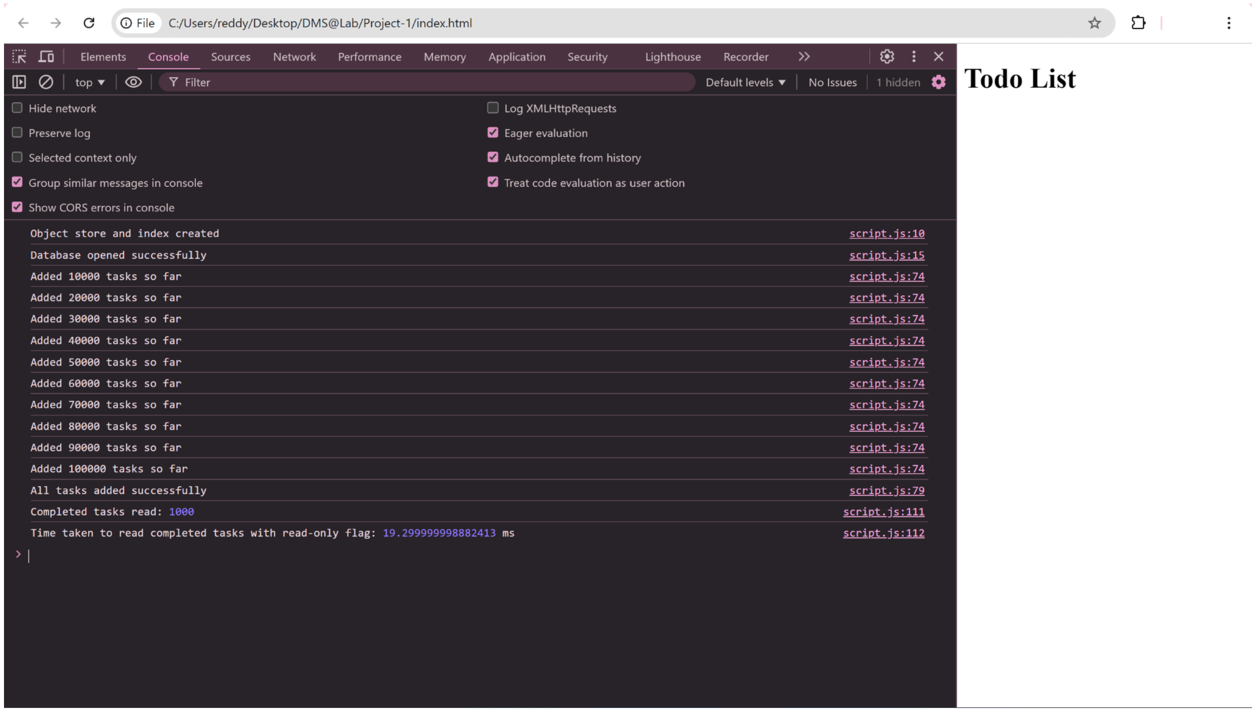


☐ Hide network
☐ Preserve log
☐ Selected context only
☑ Group similar messages in console
☑ Show CORS errors in console
☐ Log XMLHttpRequests
☑ Eager evaluation
☑ Autocomplete from history
☑ Treat code evaluation as user action

```
{"id":990,"task":"Organize office meeting","status":"completed","dueDate":"2024-10-18"}                 script.js:96
{"id":991,"task":"Finish the monthly report","status":"completed","dueDate":"2024-11-05"}               script.js:96
{"id":992,"task":"Update website content","status":"completed","dueDate":"2024-10-15"}                  script.js:96
{"id":993,"task":"Update website content","status":"completed","dueDate":"2024-10-16"}                  script.js:96
{"id":994,"task":"Prepare for client presentation","status":"completed","dueDate":"2024-11-02"}         script.js:96
{"id":995,"task":"Update website content","status":"completed","dueDate":"2024-10-19"}                  script.js:96
{"id":996,"task":"Review team feedback","status":"completed","dueDate":"2024-10-28"}                    script.js:96
{"id":997,"task":"Update website content","status":"completed","dueDate":"2024-10-21"}                  script.js:96
{"id":998,"task":"Finish the monthly report","status":"completed","dueDate":"2024-11-01"}               script.js:96
{"id":999,"task":"Update website content","status":"completed","dueDate":"2024-10-20"}                  script.js:96
{"id":1000,"task":"Prepare for client presentation","status":"completed","dueDate":"2024-11-05"}        script.js:96
{"id":1001,"task":"Review team feedback","status":"in progress","dueDate":"2024-10-27"}                 script.js:96
{"id":1002,"task":"Prepare for client presentation","status":"in progress","dueDate":"2024-10-18"}      script.js:96
{"id":1003,"task":"Prepare for client presentation","status":"in progress","dueDate":"2024-11-03"}      script.js:96
{"id":1004,"task":"Update website content","status":"in progress","dueDate":"2024-10-25"}               script.js:96
{"id":1005,"task":"Prepare for client presentation","status":"in progress","dueDate":"2024-10-22"}      script.js:96
{"id":1006,"task":"Update website content","status":"in progress","dueDate":"2024-11-01"}               script.js:96
{"id":1007,"task":"Prepare for client presentation","status":"in progress","dueDate":"2024-10-31"}      script.js:96
{"id":1008,"task":"Prepare for client presentation","status":"in progress","dueDate":"2024-11-01"}      script.js:96
{"id":1009,"task":"Organize office meeting","status":"in progress","dueDate":"2024-10-13"}              script.js:96
{"id":1010,"task":"Update website content","status":"in progress","dueDate":"2024-10-16"}               script.js:96
{"id":1011,"task":"Organize office meeting","status":"in progress","dueDate":"2024-11-01"}              script.js:96
{"id":1012,"task":"Organize office meeting","status":"in progress","dueDate":"2024-10-17"}              script.js:96
```

Task-3

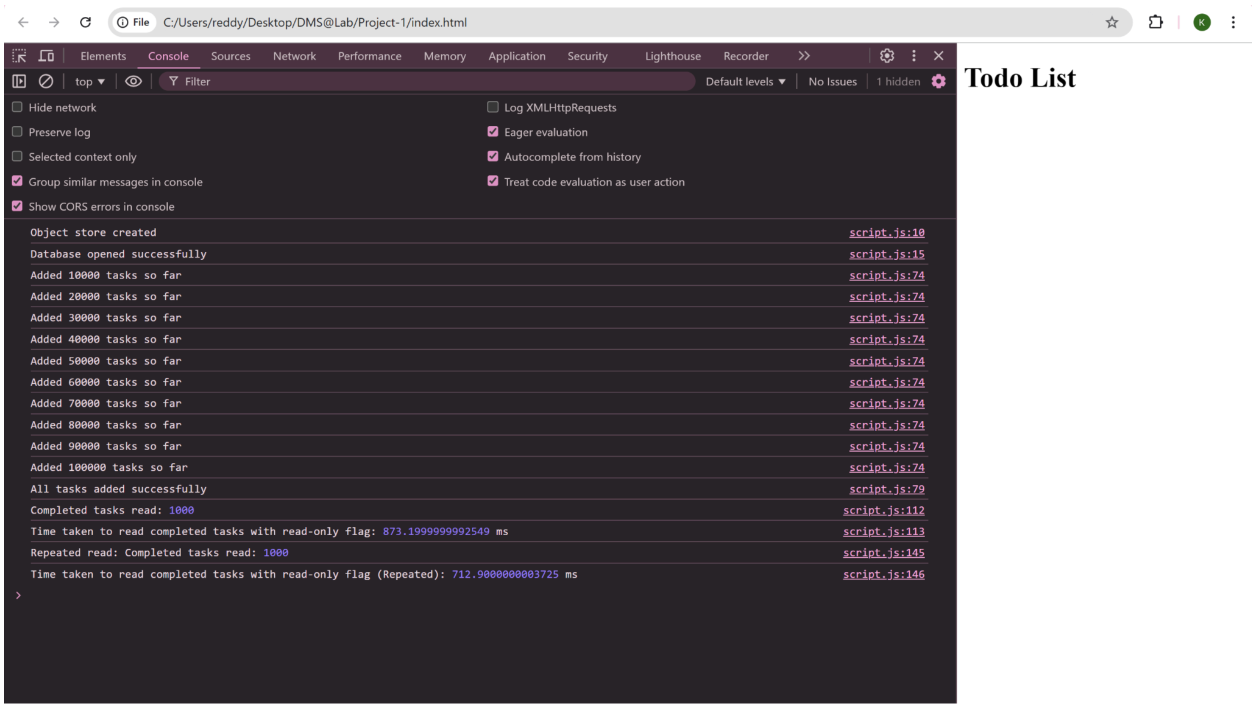
Measure and display the time (in milliseconds) required to read all objects with `status` set to "completed" on the console or the browser



```
Object store and index created                                              script.js:10
Database opened successfully                                                script.js:15
Added 10000 tasks so far                                                    script.js:74
Added 20000 tasks so far                                                    script.js:74
Added 30000 tasks so far                                                    script.js:74
Added 40000 tasks so far                                                    script.js:74
Added 50000 tasks so far                                                    script.js:74
Added 60000 tasks so far                                                    script.js:74
Added 70000 tasks so far                                                    script.js:74
Added 80000 tasks so far                                                    script.js:74
Added 90000 tasks so far                                                    script.js:74
Added 100000 tasks so far                                                   script.js:74
All tasks added successfully                                                script.js:79
Completed tasks read: 1000                                                 script.js:111
Time taken to read completed tasks with read-only flag: 19.299999998882413 ms  script.js:112
```

Task-4

Apply a **read-only flag** to the object store and measure and display the time to read all completed tasks again on the console or the browser. Please lookup the documentation:

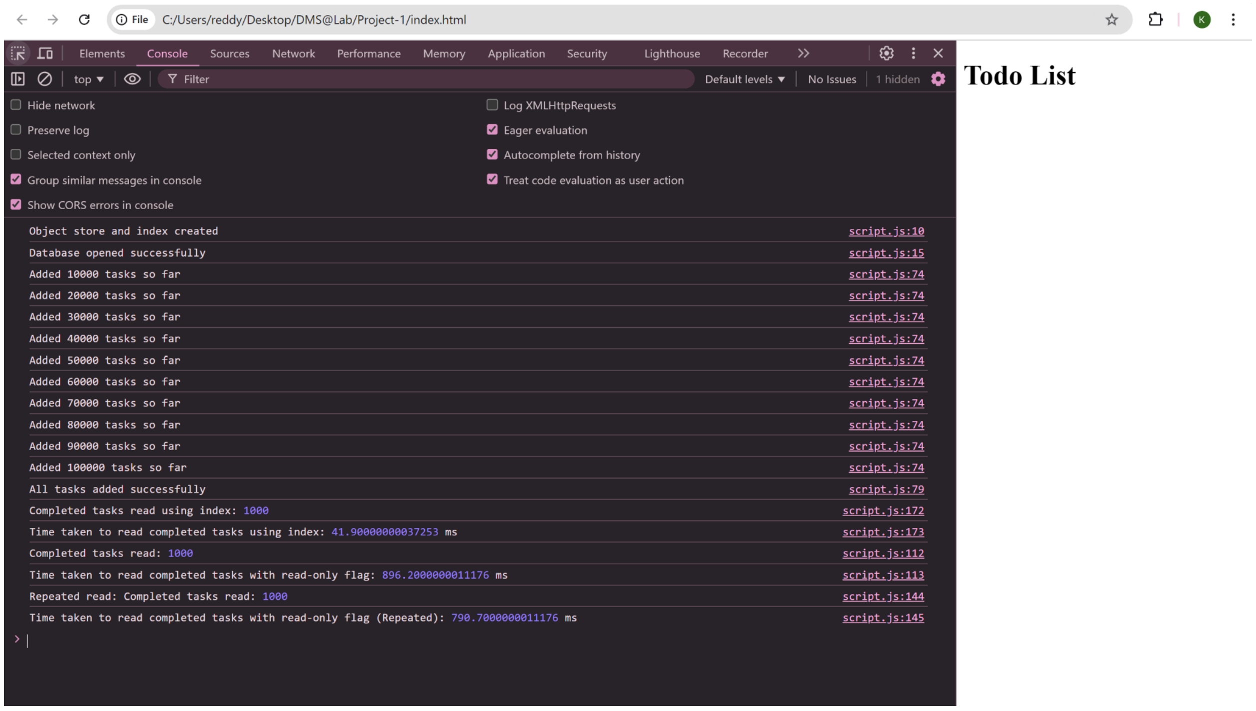


Hide network
Preserve log
Selected context only
Group similar messages in console ✓
Show CORS errors in console ✓
Log XMLHttpRequests
Eager evaluation ✓
Autocomplete from history ✓
Treat code evaluation as user action ✓

```
Object store created                                                              script.js:10
Database opened successfully                                                      script.js:15
Added 10000 tasks so far                                                          script.js:74
Added 20000 tasks so far                                                          script.js:74
Added 30000 tasks so far                                                          script.js:74
Added 40000 tasks so far                                                          script.js:74
Added 50000 tasks so far                                                          script.js:74
Added 60000 tasks so far                                                          script.js:74
Added 70000 tasks so far                                                          script.js:74
Added 80000 tasks so far                                                          script.js:74
Added 90000 tasks so far                                                          script.js:74
Added 100000 tasks so far                                                         script.js:74
All tasks added successfully                                                      script.js:79
Completed tasks read: 1000                                                       script.js:112
Time taken to read completed tasks with read-only flag: 873.1999999992549 ms     script.js:113
Repeated read: Completed tasks read: 1000                                        script.js:145
Time taken to read completed tasks with read-only flag (Repeated): 712.9000000003725 ms   script.js:146
>
```

Task-5

Create an index on the `status` field, then measure and display the time to read all completed tasks on the console or the browser. Please lookup the documentation:

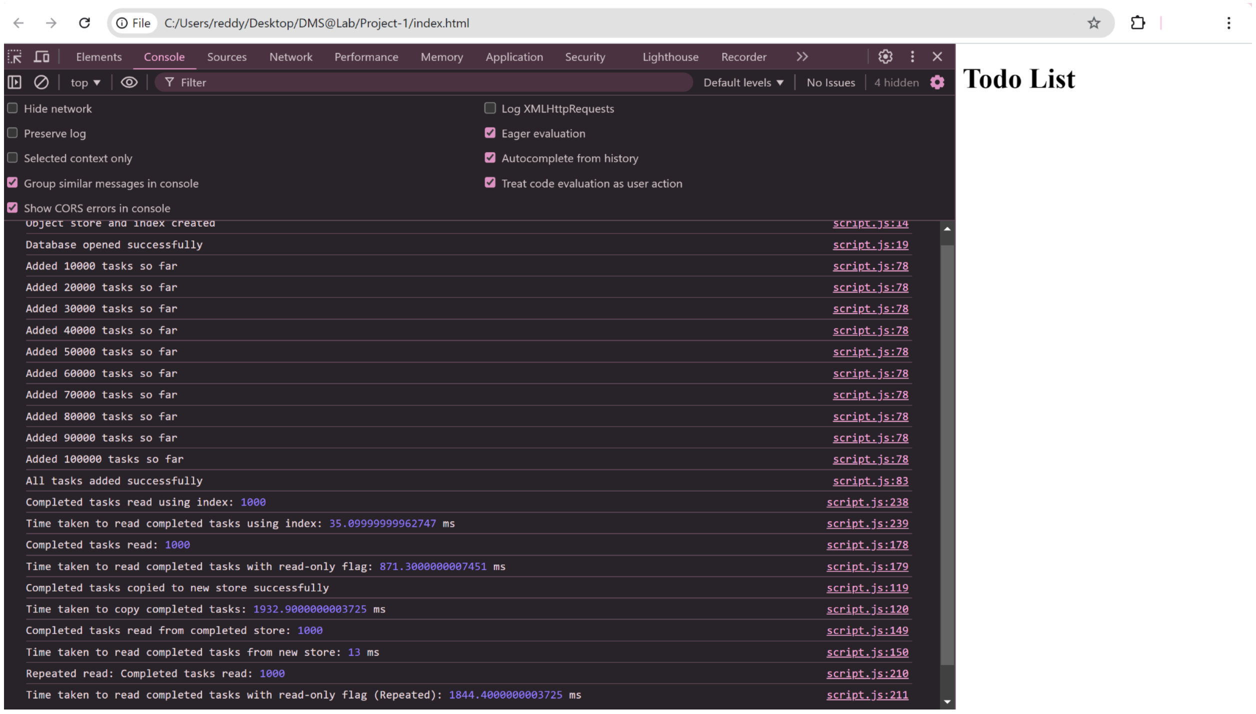


Hide network
Preserve log
Selected context only
Group similar messages in console ✓
Show CORS errors in console ✓
Log XMLHttpRequests
Eager evaluation ✓
Autocomplete from history ✓
Treat code evaluation as user action ✓

```
Object store and index created                                                   script.js:10
Database opened successfully                                                     script.js:15
Added 10000 tasks so far                                                         script.js:74
Added 20000 tasks so far                                                         script.js:74
Added 30000 tasks so far                                                         script.js:74
Added 40000 tasks so far                                                         script.js:74
Added 50000 tasks so far                                                         script.js:74
Added 60000 tasks so far                                                         script.js:74
Added 70000 tasks so far                                                         script.js:74
Added 80000 tasks so far                                                         script.js:74
Added 90000 tasks so far                                                         script.js:74
Added 100000 tasks so far                                                        script.js:74
All tasks added successfully                                                     script.js:79
Completed tasks read using index: 1000                                          script.js:172
Time taken to read completed tasks using index: 41.90000000037253 ms            script.js:173
Completed tasks read: 1000                                                      script.js:112
Time taken to read completed tasks with read-only flag: 896.2000000011176 ms    script.js:113
Repeated read: Completed tasks read: 1000                                       script.js:144
Time taken to read completed tasks with read-only flag (Repeated): 790.7000000011176 ms  script.js:145
>
```

Task-6

Define a new object store called "TodoListCompleted", copy all completed tasks from "TodoList" to this new store, and measure and display the time required to read all completed tasks from "TodoListCompleted" on the console or the browser.



☐ Hide network
☐ Preserve log
☐ Selected context only
☑ Group similar messages in console
☑ Show CORS errors in console
☐ Log XMLHttpRequests
☑ Eager evaluation
☑ Autocomplete from history
☑ Treat code evaluation as user action

```
Object store and index created                                                      script.js:14
Database opened successfully                                                        script.js:19
Added 10000 tasks so far                                                            script.js:78
Added 20000 tasks so far                                                            script.js:78
Added 30000 tasks so far                                                            script.js:78
Added 40000 tasks so far                                                            script.js:78
Added 50000 tasks so far                                                            script.js:78
Added 60000 tasks so far                                                            script.js:78
Added 70000 tasks so far                                                            script.js:78
Added 80000 tasks so far                                                            script.js:78
Added 90000 tasks so far                                                            script.js:78
Added 100000 tasks so far                                                           script.js:78
All tasks added successfully                                                        script.js:83
Completed tasks read using index: 1000                                              script.js:238
Time taken to read completed tasks using index: 35.09999999962747 ms                script.js:239
Completed tasks read: 1000                                                          script.js:178
Time taken to read completed tasks with read-only flag: 871.3000000007451 ms        script.js:179
Completed tasks copied to new store successfully                                    script.js:119
Time taken to copy completed tasks: 1932.9000000003725 ms                           script.js:120
Completed tasks read from completed store: 1000                                     script.js:149
Time taken to read completed tasks from new store: 13 ms                            script.js:150
Repeated read: Completed tasks read: 1000                                           script.js:210
Time taken to read completed tasks with read-only flag (Repeated): 1844.4000000003725 ms   script.js:211
```

Lessons Learned

Task 1: Create and Populate IndexedDB

- Insight: We learned how to create an IndexedDB object store named "TodoList" and populate it with 100,000 randomly generated objects, each containing properties like task, status, and dueDate.
- Key Takeaway: This task highlighted the importance of the asynchronous nature of IndexedDB. We discovered that if two functions are executed sequentially, the second function may not work properly if the first one has not completed, underscoring the need for effective management of asynchronous operations.

Tasks 2, 3, and 4: Measure Read Times

- Insight: In these tasks, we learned how changing the method of accessing the database affects the time required to retrieve data. Specifically, we measured the time taken to read all objects with the status set to "completed" using different access methods, including read-only transactions and indexed queries.
- Key Takeaway: This experience taught us the significance of optimizing data access patterns. We became familiar with using cursors and how they can enhance performance when retrieving data from large object stores.

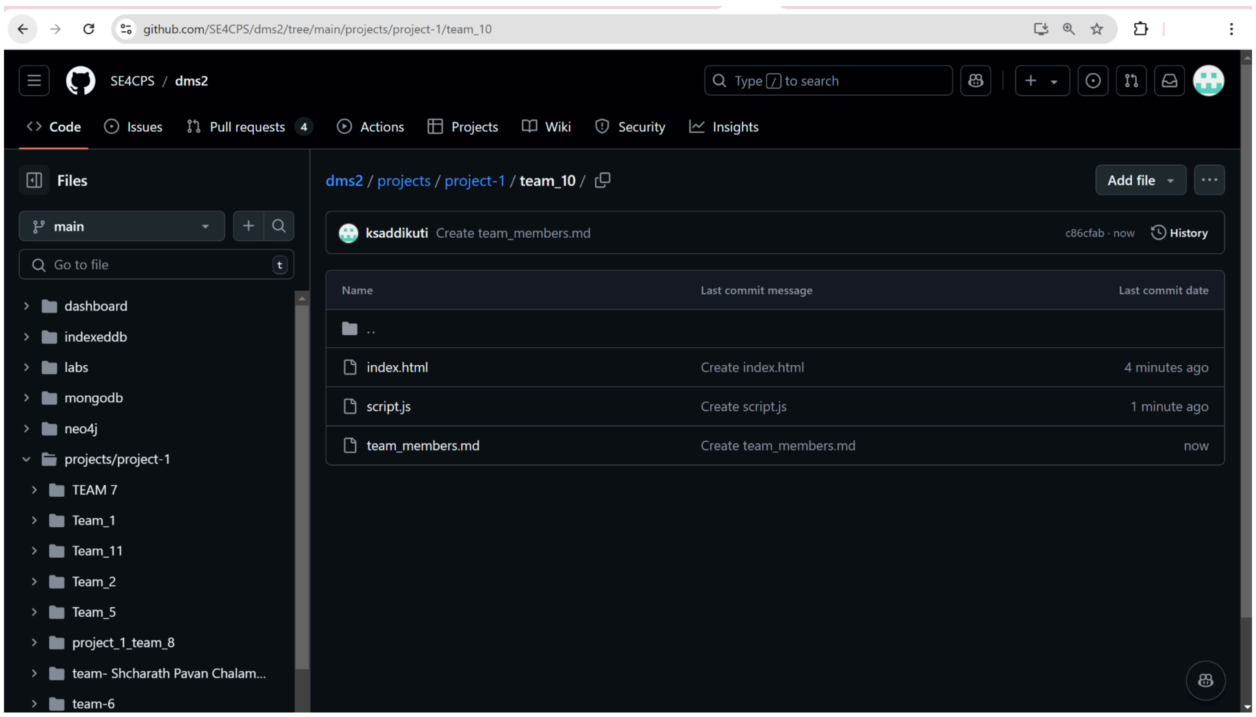
Task 5,6: Create and Populate "TodoListCompleted" Object Store

- Insight: In this task, we learned how to filter data from one object store and copy it to another, specifically creating a new store called "TodoListCompleted" for completed tasks. We also learned how to efficiently add multiple records to a new database.
- Key Takeaway: We observed that the time taken to read data from a new object store is significantly reduced when tasks are organized in separate tables. This reinforces the importance of data organization in improving database performance.

Challenges Faced

- We encountered challenges, particularly in understanding and working with the asynchronous nature of IndexedDB. Managing the flow of operations to ensure that one task completes before another starts was crucial for the success of our program. This experience has enhanced our ability to navigate asynchronous programming patterns, which is essential for web development.

Github Screenshot:



github.com/SE4CPS/dms2/tree/main/projects/project-1/team_10

SE4CPS / dms2

dms2 / projects / project-1 / team_10 /

ksaddikuti Create team_members.md c86cfab · now History

Name	Last commit message	Last commit date
..		
index.html	Create index.html	4 minutes ago
script.js	Create script.js	1 minute ago
team_members.md	Create team_members.md	now