

# **Advanced Database System Lab**

## **Assignment 6**

### **Parallel Databases**

#### **Introduction to Parallel Databases**

Companies need to handle huge amounts of data with high data transfer rates. The client server and centralized system is not very efficient. The need to improve the efficiency gave birth to the concept of Parallel Databases.

Parallel database system improves performance of data processing using multiple resources in parallel, like multiple CPU and disks are used parallelly.

It also performs many parallelization operations like, data loading and query processing.

#### **Goals of Parallel Databases**

The concept of Parallel Database was built with a goal to:

Improve performance:

The performance of the system can be improved by connecting multiple CPU and disks in parallel. Many small processors can also be connected in parallel.

Improve availability of data:

Data can be copied to multiple locations to improve the availability of data.

For example: if a module contains a relation (table in database) which is unavailable then it is important to make it available from another module.

Improve reliability:

Reliability of the system is improved with completeness, accuracy and availability of data.

Provide distributed access of data:

Companies having many branches in multiple cities can access data with the help of a parallel database system.

## Steps: 1. Created the Tables

### ParallelSort():

1. Students
2. Output\_students

### ParallelJoin():

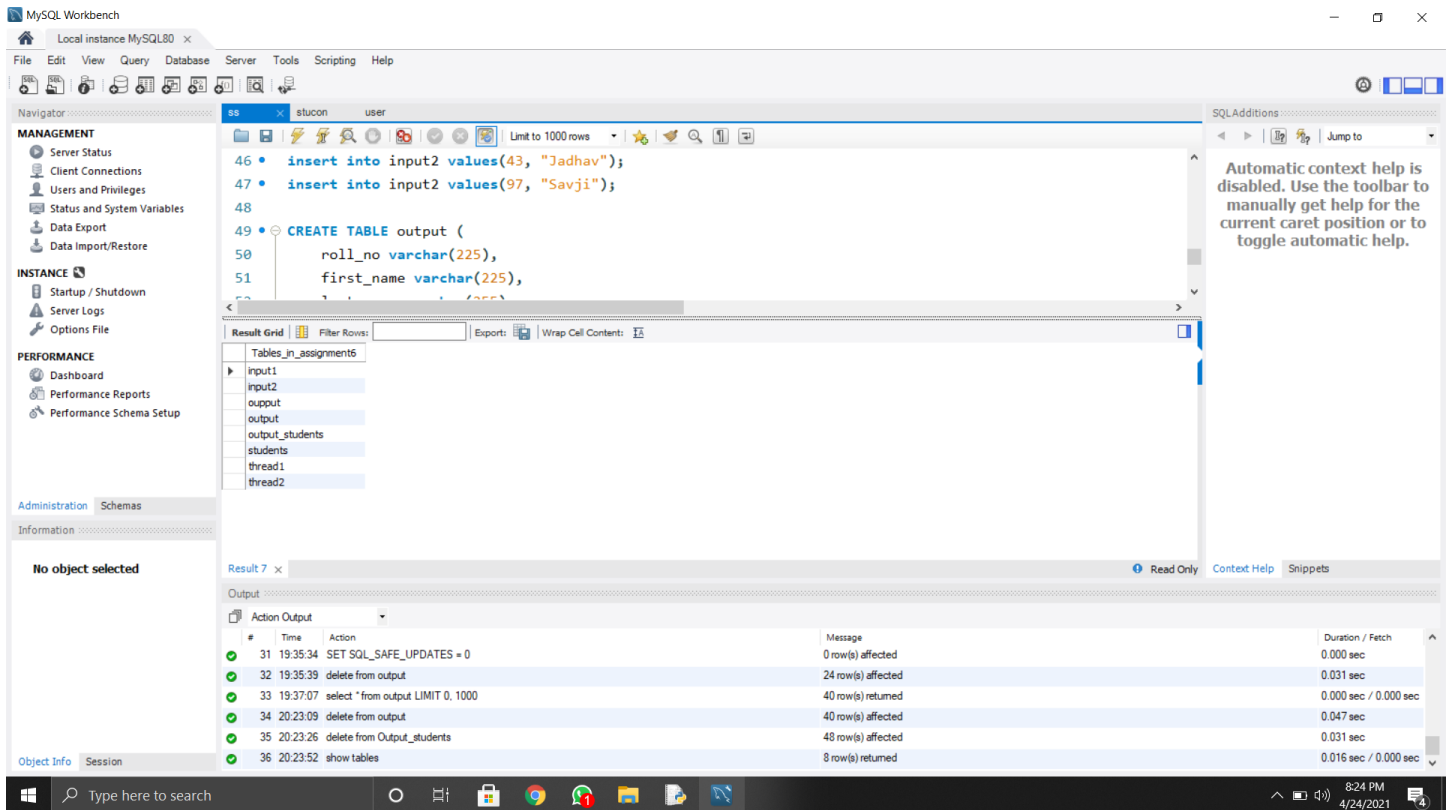
1. Input1
2. Input2
3. Output

The screenshot displays the MySQL Workbench interface. The left sidebar contains the 'MANAGEMENT' and 'PERFORMANCE' sections. The main editor window shows a SQL script with the following content:

```
35 roll_no int,  
36 last_name varchar(255)  
37 );  
38  
39 insert into input1 values(99, "Ganesh");  
40 insert into input1 values(89, "Sudhanshu");  
41 insert into input1 values(43, "Shivani");  
42 insert into input1 values(97, "Suyash");  
43  
44 insert into input2 values(99, "Kasan");  
45 insert into input2 values(89, "Pusadkar");  
46 insert into input2 values(43, "Jadhav");  
47 insert into input2 values(97, "Savji");  
48  
49 CREATE TABLE output (  
50 roll_no varchar(225),  
51 first_name varchar(225),  
52 last_name varchar(255)  
53 );  
54  
55 use Assignment6;
```

The bottom panel shows the 'Output' tab with the following execution results:

#	Time	Action	Message	Duration / Fetch
30	19:35:18	delete from output	Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that us...	0.000 sec
31	19:35:34	SET SQL_SAFE_UPDATES = 0	0 row(s) affected	0.000 sec
32	19:35:39	delete from output	24 row(s) affected	0.031 sec
33	19:37:07	select * from output LIMIT 0, 1000	40 row(s) returned	0.000 sec / 0.000 sec
34	20:23:09	delete from output	40 row(s) affected	0.047 sec
35	20:23:26	delete from Output_students	48 row(s) affected	0.031 sec



## Step 2: Created Python Program

1. Created Connection to MySQL using MySQL connector
2. Sorted the Students table using threads
3. Joined the input1 & input2 stored in new table output



## 2. After Sorting

The screenshot shows the MySQL Workbench interface with the following components:

- Navigator:** Displays the database structure, including 'MANAGEMENT' (Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore) and 'INSTANCE' (Startup / Shutdown, Server Logs, Options File).
- SQL Editor:** Contains the following SQL queries:

```
58  
59 • select * from output;  
60  
61 • delete from output;  
62  
63 • select * from Output_students;
```
- Result Grid:** Displays the results of the last query, showing a table with columns: roll\_no, first\_name, last\_name, city. The data is as follows:

roll_no	first_name	last_name	city
43	Shivani	Jadhav	Satara
89	Sudhanshu	Pusadkar	Nagpur
97	Suyash	Savji	Beed
99	Ganesh	Kasar	Jalgaon
- Action Output:** Displays the execution log of the queries, showing the time taken and the number of rows affected or returned.

#	Time	Action	Message	Duration / Fetch
36	20:23:52	show tables	8 row(s) returned	0.016 sec / 0.000 sec
37	20:33:56	delete from Output_students	12 row(s) affected	0.094 sec
38	20:34:08	delete from output	24 row(s) affected	0.093 sec
39	20:35:29	select * from Students LIMIT 0, 1000	4 row(s) returned	0.015 sec / 0.000 sec
40	20:35:50	select * from Output_students LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
41	20:36:09	select * from Output_students LIMIT 0, 1000	4 row(s) returned	0.016 sec / 0.000 sec
- SQLAdditions:** A sidebar on the right with a message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

## 1. Input Tables

The screenshot shows the MySQL Workbench interface with the following components:

- Navigator:** Displays the database structure, including 'MANAGEMENT' (Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore) and 'INSTANCE' (Startup / Shutdown, Server Logs, Options File).
- SQL Editor:** Contains the following SQL queries:

```
59 • select * from output;  
60  
61 • delete from output;  
62  
63 • select * from input1;  
64 • select * from input2;
```
- Result Grid:** Displays the results of the last query, showing a table with columns: roll\_no, first\_name. The data is as follows:

roll_no	first_name
99	Ganesh
89	Sudhanshu
43	Shivani
97	Suyash
- Action Output:** Displays the execution log of the queries, showing the time taken and the number of rows affected or returned.

#	Time	Action	Message	Duration / Fetch
39	20:35:29	select * from Students LIMIT 0, 1000	4 row(s) returned	0.015 sec / 0.000 sec
40	20:35:50	select * from Output_students LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
41	20:36:09	select * from Output_students LIMIT 0, 1000	4 row(s) returned	0.016 sec / 0.000 sec
42	20:36:38	select * from input1 LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
43	20:36:38	select * from input2 LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
44	20:36:47	select * from input1 LIMIT 0, 1000	4 row(s) returned	0.016 sec / 0.000 sec
- SQLAdditions:** A sidebar on the right with a message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

## 2. Output tables

The screenshot displays the MySQL Workbench interface. The left sidebar contains the 'Navigator' pane with sections for 'MANAGEMENT' (Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore), 'INSTANCE' (Startup / Shutdown, Server Logs, Options File), and 'PERFORMANCE' (Dashboard, Performance Reports, Performance Schema Setup). Below these are tabs for 'Administration' and 'Schemas', with 'Information' selected. The main editor shows a SQL script with the following queries:

```
60
61 • delete from output;
62
63 • select * from input1;
64 • select * from input2;
65
```

The 'Result Grid' below the script shows a table with columns 'roll\_no', 'first\_name', and 'last\_name'. The data is as follows:

roll_no	first_name	last_name
99	Ganesh	Kasar
89	Sudhanshu	Pusadkar
43	Shivani	Jadhav
97	Suyash	Savji

The bottom pane shows the 'Output' tab with a table titled 'Action Output' containing execution details:

#	Time	Action	Message	Duration / Fetch
41	20:36:09	select * from Output_students LIMIT 0, 1000	4 row(s) returned	0.016 sec / 0.000 sec
42	20:36:38	select * from input1 LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
43	20:36:38	select * from input2 LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
44	20:36:47	select * from input1 LIMIT 0, 1000	4 row(s) returned	0.016 sec / 0.000 sec
45	20:36:56	select * from input2 LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
46	20:37:08	select * from output LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

The right sidebar contains the 'SQLAdditions' pane with a message: 'Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.'